Solaris™ Security Toolkit 4.1 관리 설명서

Sun Microsystems, Inc.
www.sun.com

부품 번호 817-7654-10
2004년 10월, 개정판 A

다음 사이트로 이 설명서에 대한 귀하의 의견을 보내주십시오: http://www.sun.com/hwdocs/feedback

재활용
가능

Adobe PostScript

목차

그림

표

코드 예제

머리말

이 설명서에는 Solaris Security Toolkit 소프트웨어의 이해 및 사용에 대한 참고 정보가 나옵니다. 이 설명서는 주로 Solaris Security Toolkit 소프트웨어를 사용하여 Solaris™ 운영 체제(OS) 버전 8과 9에서 보안하려는 관리자, 컨설턴트 및 새로운 Sun 시스템을 전개하거나 전개된 시스템을 보안하려는 사용자를 위한 것입니다. 지침은 JumpStart™ 모드 또는 독립형 모드에서 소프트웨어를 사용하는데 적용됩니다.

이 문서를 읽기 전에

사용자는 Solaris™용 Sun 인증 시스템 관리자 또는 Solaris™ 운영 체제에 대한 Sun 인증 네트워크 관리자여야 합니다. 또한 독립형 네트워크 프로토콜 및 토폴로지를 이해해야 합니다.

이 문서는 다양한 보안 경험이나 지식이 있는 사람에게 유용하도록 고안되었기 때문에 사용자의 경험과 지식이 이 문서를 사용하는 방법을 결정합니다.

이 설명서의 구성

이 설명서는 사용자 안내서로 사용됩니다. 각 장에는 시스템 보안을 위해 소프트웨어를 사용하기 위한 정보, 지시사항 및 지침이 들어있습니다. 이 문서는 다음과 같이 구성됩니다.

1 장은 Solaris Security Toolkit 소프트웨어의 디자인 및 목적에 대해 설명합니다. 핵심 구성요소, 기능, 이점 및 지원 플랫폼을 다룹니다.

2 장은 시스템 보안 방법론에 대해 설명합니다. Solaris Security Toolkit 소프트웨어를 사용하여 시스템을 보안하기 전에 적용할 수 있는 프로세스를 제공합니다.

3 장은 Solaris Security Toolkit 소프트웨어 및 기타 보안 관련 소프트웨어 다운로드, 설치 및 실행에 대한 지시사항을 제공합니다.

4 장은 강화 작업 중에 Solaris Security Toolkit 소프트웨어를 통해 수행된 변경사항의 역전(실행 취소)에 대한 정보 및 절차를 제공합니다.

5 장은 Solaris Security Toolkit 소프트웨어를 사용하기 위한 JumpStart 서버 구성 및 관리 정보를 제공합니다.

6 장은 Solaris Security Toolkit 소프트웨어를 사용하여 시스템의 보안을 감사(검증)하는 방법에 대해 설명합니다. 강화 후에 설정된 보안 프로파일 유지보수에 대해서는 이 장의 정보와 절차를 사용하십시오.

7 장은 새로운 시스템을 설치 및 보안하기 위해 이전 장에서 제공되는 정보와 전문 지식을 실현 가능한 시나리오에 적용하는 방법에 대해 설명합니다.

UNIX® 명령어 사용

이 문서는 시스템 종료, 시스템 시동 및 장치 구성과 같은 기본 UNIX® 명령어 및 절차를 포함하지 않을 수도 있습니다. 이 정보에 대해서는 다음을 참조하십시오.

- 시스템과 함께 제공된 소프트웨어 설명서
- 다음 웹 사이트에서 나오는 Solaris 운영 체제 설명서

  `http://docs.sun.com`

쉘 프롬프트

쉘	프롬프트
C 쉘	*machine-name*%
C 쉘 수퍼유저	*machine-name*#
Bourne 쉘 및 Korn 쉘	$
Bourne 쉘 및 Korn 쉘 수퍼유저	#

활자체 규약

활자체*	의미	보기
AaBbCc123	명령어, 파일 및 디렉토리 이름; 화면 출력	.login 파일을 편집하십시오. 모든 파일을 나열하려면 ls -a를 사용하십시오. % You have mail.
AaBbCc123	컴퓨터 화면 출력에서 사용자가 직접 입력하는 내용	% **su** Password:
AaBbCc123	문서 제목, 새 단어 및 용어, 강조하는 단어. 명령줄 변수를 실제 이름이나 값으로 바꾸십시오.	사용 설명서의 제 6장을 읽어 보십시오. 이들을 *class* 옵션이라고 합니다. 이 작업을 수행하려면 수퍼유저여야 합니다. 파일을 삭제하려면 rm 파일이름을 입력하십시오.

* 사용자 브라우저의 설정이 이 설정과 다를 수 있습니다.

Sun 설명서 액세스

다음 웹 사이트에서 번역 버전을 포함한 광범위한 Sun 문서를 열람, 인쇄 또는 구입할 수 있습니다.

`http://www.sun.com/documentation`

타사 웹 사이트

Sun은 이 설명서에 언급된 타사 웹 사이트의 이용 여부에 대해 책임지지 않습니다. Sun은 해당 사이트 또는 자원을 통해 사용 가능한 내용, 광고, 제품 또는 기타 자료에 대해 보증하거나 책임지지 않습니다. Sun은 그러한 사이트 또는 자원을 통해 사용 가능한 내용, 상품 또는 서비스의 사용이나 신뢰에 의해 야기되는 실질적 또는 주장된 손해나 손실에 대해 책임지지 않습니다.

관련 자원

관련 설명서 및 사이트가 이 절에 나열되어 있습니다.

참고 서적

- Andert, Donna, Wakefield, Robin 및 Weise, Joel. "Trust Modeling for Security Architecture Development,"Sun BluePrints™ OnLine, 2002년 12월, `http://www.sun.com/blueprints/1202/817-0775.pdf`
- Dasan, Vasanthan, Noordergraaf, Alex 및 Ordica, Lou. "The Solaris Fingerprint Database - A Security Tool for Solaris Software and Files," Sun BluePrints OnLine, 2001년 5월, `http://www.sun.com/blueprints/0501/Fingerprint.pdf`.
- Englund, Martin, ìSecuring Systems with Host-Based Firewalls - Implemented With SunScreen Lite 3.1 Software,î Sun BluePrints OnLine, 2001년 9월, `http://sun.com/blueprints/0901/sunscreenlite.pdf`.
- Garfinkel, Simon 및 Spafford, Gene. *Practical UNIX and Internet Security*, 2판, O'Reilly & Associates, 1996년 4월.

- Howard, John S. 및 Noordergraaf, Alex. *JumpStart Technology: Effective Use in the Solaris Operating Environment*, The Official Sun Microsystems Resource Series, Prentice Hall, 2001년 8월.
- Moffat, Darren J., FOCUS on SUN: *Solaris BSM Auditing*, `http://www.securityfocus.com/infocus/1362`
- Noordergraaf, Alex. "Solaris™ Operating Environment Minimization for Security: A Simple, Reproducible and Secure Application Installation Methodology Updated for Solaris 8 Operating Environment," Sun BluePrints OnLine, 2000년 11월, `http://sun.com/blueprints/1100/minimize-updt1.pdf.`
- Noordergraaf, Alex. "Minimizing the Solaris Operating Environment for Security: Updated for Solaris 9 Operating Environment," Sun BluePrints OnLine, 2002년 11월,`http://sun.com/blueprints/1102/816-5241.pdf.`
- Noordergraaf, Alex. "Securing the Sun Cluster 3.x Software,"Sun BluePrints OnLine 기사, 2003년 2월, `http://www.sun.com/solutions/blueprints/0203/817-1079.pdf`
- Noordergraaf, Alex, "Securing the Sun Enterprise 10000 System Service Processors,"Sun BluePrints OnLine 기사, 2002년 3월, `http://www.sun.com/blueprints/0302/securingenter.pdf`
- Noordergraaf, Alex 등, *Enterprise Security: Solaris Operating Environment Security Journal, Solaris Operating Environment Versions 2.5.1, 2.6, 7, and 8*, Sun Microsystems™, Prentice Hall Press, ISBN 0-13-100092-6, 2002년 6월.
- Noordergraaf, Alex 및 Nimeh, Dina. "Securing the Sun Fire 12K and 15K Domains,"Sun BluePrints OnLine 기사, 2003년 2월, `http://www.sun.com/blueprints/0203/817-1357.pdf`
- Noordergraaf, Alex 및 Nimeh, Dina. "Securing the Sun Fire 12K and 15K System Controllers,"Sun BluePrints OnLine 기사, 2003년 2월, `http://www.sun.com/blueprints/0203/817-1358.pdf`
- Noordergraaf, Alex 및 Watson, Keith. "Solaris Operating Environment Security: Updated for the Solaris 9 Operating Environment," Sun BluePrints OnLine, 2002년 12월, `http://www.sun.com/blueprints/1202/816-5242.pdf.`
- O'Donnell, Nicholas 및 Noordergraaf, Alex. "Minimizing Domains for Sun Fire V1280, 6800, 12K, and 15K Systems," Sun BluePrints OnLine articles, 2003년 9월, `http://www.sun.com/blueprints/0903/817-3340.pdf` [Part I] 및 `http://www.sun.com/blueprints/0903/817-3628.pdf` [Part II]
- Osser, William 및 Noordergraaf, Alex. "Auditing in the Solaris 8 Operating Environment," Sun BluePrints OnLine, 2001년 2월 `http://www.sun.com/blueprints/0201/audit_config.pdf`.
- Reid, Jason M. 및 Watson, Keith. "Building and Deploying OpenSSH in the Solaris Operating Environment," Sun BluePrints OnLine, 2001년 7월, `http://sun.com/blueprints/0701/openSSH.pdf`

- Reid, Jason M. "Configuring OpenSSH for the Solaris Operating Environment,"Sun BluePrints OnLine 기사, 2002년 1월, `http://www.sun.com/blueprints/0102/configssh.pdf`
- Reid, Jason. *Secure Shell in the Enterprise*, The Official Sun Microsystems Resource Series, Prentice Hall, 2003년 6월
- *Solaris Advanced Installation Guide*, Sun Microsystems, `http://docs.sun.com`.
- *SunSHIELD Basic Security Module Guide*, Sun Microsystems, Inc., `http://docs.sun.com`.
- Watson, Keith 및 Noordergraaf, Alex. "Solaris Operating Environment Network Settings for Security: Updated for Solaris 9 Operating Environment,"Sun BluePrints OnLine, 2003년 6월, `http://www.sun.com/solutions/blueprints/0603/816-5240.pdf`
- Weise, Joel, and Martin, Charles R. "Developing a Security Policy,"Sun BluePrints OnLine 기사, 2001년 12월, `http://www.sun.com/solutions/blueprints/1201/secpolicy.pdf`

웹 사이트

- AUSCERT, *UNIX Security Checklist*, http://www.auscert.org.au/render.html?it=1935&cid=1920
- CERT/CC(`http://www.cert.org`)는 연방 정부의 지원을 받아서 컴퓨터 보안 문제에 대해 연구하는 연구 개발 센터입니다.
- Chkrootkit, `http://www.chkrootkit.org`
- Galvin, Peter Baer, *Solaris Security FAQ*, `http://www.itworld.com/Comp/2377/security-faq/`
- HoneyNet Project, "Know Your Enemy: Motives" `http://project.honeynet.org/papers/motives/`
- List open files software, `ftp://vic.cc.purdue.edu/pub/tools/unix/lsof/`
- Nmap Port Scanner, `http://www.insecure.org`
- OpenSSH tool, `http://www.openssh.com/`
- Pomeranz, Hal, *Solaris Security Step by Step*, `http://www.sans.org/`
- Rhoads, Jason, *Solaris Security Guide*, `http://www.sabernet.net/papers/Solaris.html`
- Security Focus(`http://www.securityfocus.org`)는 관련 보안 주제에 대해 논의하는 전용 웹 사이트입니다.
- Sendmail Consortium, `sendmail` 구성 정보, `http://www.sendmail.org/`

- Spitzner, Lance, *Armoring Solaris*, `http://secinf.net/unix_security/Armoring_Solaris.html`
- SSH Communications Security, Secure Shell (SSH) tool, `http://www.ssh.com/`
- Sun BluePrints OnLine, `http://sun.com/blueprints`
- FixModes 소프트웨어용 Sun BluePrints OnLine Tools 및 MD5 스크립트, `http://jsecom15k.sun.com/ECom/EComActionServlet?StoreId=8&PartDetailId=817-0074-10&TransactionId=try&LMLoadBalanced=`
- Sun Enterprise Authentication Mechanism™ 정보, `http://www.sun.com/software/solaris/ds/ds-seam`
- SunSolve℠ - `http://sunsolve.sun.com`

Sun 기술 지원 문의

이 문서에서 해답을 찾을 수 없는 제품에 대한 기술 관련 질문은 다음 웹 사이트를 방문하십시오.

`http://www.sun.com/service/contacting`

Sun은 여러분의 의견을 환영합니다

Sun은 본 문서의 내용 향상에 관심이 있으며 사용자 의견을 환영합니다. 다음 웹 사이트에서 사용자 의견을 제출할 수 있습니다.

`http://www.sun.com/hwdocs/feedback`

의견에 문서의 제목과 부품 번호를 적어 주십시오.

Solaris Security Toolkit 4.1 관리 설명서, 부품 번호 817-7654-10

1장

서론

이 장은 Solaris Security Toolkit 소프트웨어의 디자인 및 용도에 대해 설명합니다. 핵심 구성요소, 기능, 이점 및 지원 플랫폼을 다룹니다. 이 장은 수정 및 전개의 버전 제어 유지를 위한 지침을 제공하며, Solaris Security Toolkit 소프트웨어의 사용자 정의를 위한 중요 지침을 설명합니다.

이 장에서는 다음 주제를 다룹니다.

Solaris Security Toolkit 소프트웨어를 사용한 시스템 보안

Solaris Security Toolkit 소프트웨어는 약식으로는 JASS(JumpStart Architecture and Security Scripts) 도구 세트로 알려져 있으며 안전한 Solaris OS 시스템을 구축하고 유지보수하기 위한 자동화되고 확장 가능하며 계량 가능한 체계를 제공합니다. Solaris Security Toolkit 소프트웨어를 사용하면 시스템의 보안을 강화하고, 절차를 최소화하며, 감사를 수행할 수 있습니다.

- 강화 - 시스템 보안을 개선하기 위한 Solaris OS 구성의 수정
- 최소화 - 특정 시스템에서 필요 없는 Solaris OS 패키지의 제거 (각 시스템의 요구사항이 다양하므로, 불필요한 패키지 또한 다양하며 모두 평가되어야 합니다.) 이 제거는 패치하고 보안해야 하는 구성요소의 수를 줄이므로 침입자가 사용할 수 있는 진입점의 수를 줄입니다.

- 감사 - 시스템의 구성이 사전 정의된 보안 프로파일을 준수하는지 결정하는 프로세스
- 점수 - 점수는 감사 실행 중에 적발된 실패의 수와 연관된 값입니다. 실패가 발견되지 않을 경우, 결과 점수는 0입니다. Solaris Security Toolkit은 실패가 감지될 때마다 점수(취약성 값이라고도 함)가 1점씩 증가합니다.

시스템 설치 및 구성은 가능한 한(이상적으로는 100%) 자동화되어야 합니다. 이 지침에는 OS 설치 및 구성, 네트워크 구성, 사용자 계정, 응용 프로그램 및 보안 수정이 포함되어 있습니다. 보안 수정은 시스템의 목적에 따라서 강화 및/또는 최소화를 포함할 수 있습니다. Solaris OS 설치를 자동화하기 위해 사용 가능한 한 가지 기술이 JumpStart 소프트웨어입니다. JumpStart 소프트웨어는 사람의 간섭이 거의 또는 전혀 필요 없이 네트워크를 통해 시스템을 설치하는 체계를 제공합니다. Solaris Security Toolkit 소프트웨어는 JumpStart 소프트웨어 기반 설치에서 Solaris OS 시스템 강화 및 최소화와 연관된 대부분의 작업을 구현하고 자동화하는 프레임워크 및 스크립트를 제공합니다.

또한 Solaris Security Toolkit 소프트웨어에는 독립형 모드가 있습니다. 이 모드는 JumpStart 모드에서와 동일한 모든 강화 기능을 수행하는 기능을 제공하지만 전개된 시스템에서만 작동합니다. 두 가지 모드 모두에서 보안 수정 사항은 시스템에 대한 보안 요구사항과 일치하도록 사용자 정의할 수 있으며 또한 그렇게 되어야 합니다.

시스템의 설치 방법과 관계 없이 처음에 Solaris Security Toolkit 소프트웨어를 사용하여 시스템을 강화하고 최소화할 수 있습니다. 그런 다음 주기적으로 Solaris Security Toolkit 소프트웨어를 사용하여 보안된 시스템의 보안 프로파일이 우발적이나 악의적으로 수정되지 않았는지 감사하십시오.

주 – 감사라는 용어는 보안 상태를 사전 정의된 보안 프로파일과 비교하여 보안 상태를 검증하는 Solaris Security Toolkit 소프트웨어의 자동화된 프로세스를 설명합니다. 이 용어의 사용으로 인해 감사 옵션을 사용한 후 시스템이 완전히 안전함을 보장하는 것은 않습니다.

소프트웨어 구성요소 이해

이 절은 Solaris Security Toolkit 소프트웨어 구성요소 구조의 개요를 제공합니다. Solaris Security Toolkit 소프트웨어는 파일 및 디렉토리의 집합입니다. 그림 1-1이 해당 구조를 보여줍니다.

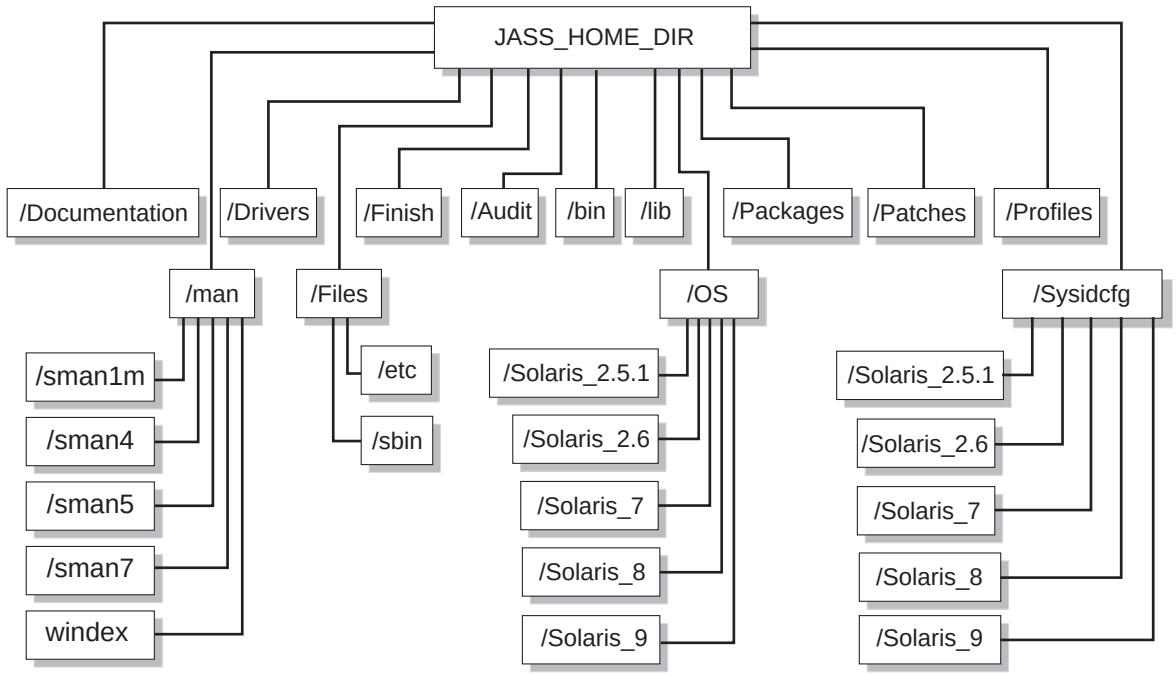

그림 **1-1** 소프트웨어 구성요소 구조

이러한 디렉토리 및 하위 디렉토리 이외에 다음 파일이 Solaris Security Toolkit 소프트웨어 구조의 최상위 레벨인 `/bin`에 있습니다.

- `add-client` - JumpStart 환경으로 클라이언트를 추가하기 위한 JumpStart 보조 프로그램
- `rm-client` - JumpStart 환경으로부터 클라이언트를 제거하기 위한 JumpStart 보조 프로그램
- `make-jass-pkg` - 사용자 정의된 Solaris Security Toolkit 구성의 내부 배포를 단순화하기 위해 Solaris Security Toolkit 디렉토리의 내용에서 Solaris OS 패키지를 작성하는 기능을 제공하는 명령
- `jass-check-sum` - 각 Solaris Security Toolkit 실행 중에 작성된 체크섬을 기초로 Solaris Security Toolkit 소프트웨어에 의해 수정된 파일이 변경되었는지 판별하는 기능을 제공하는 명령
- `jass-execute` - Solaris Security Toolkit 응용 프로그램을 구성하는 명령

디렉토리

Solaris Security Toolkit 구조의 구성요소는 다음 디렉토리로 구성됩니다.

- `/Audit`
- `/bin`
- `/Documentation`
- `/man`
- `/Drivers`
- `/Files`
- `/Finish`
- `/lib`
- `/OS`
- `/Packages`
- `/Patches`
- `/Profiles`
- `/Sysidcfg`

이 절은 각 디렉토리에 대해 설명합니다. 적절한 위치에 각 스크립트, 구성 파일 또는 하위 디렉토리가 나열되며, 자세한 정보를 보기 위한 다른 장으로의 참조가 제공됩니다.

Solaris Security Toolkit 디렉토리 구조는 다음 Sun BluePrints 설명서에 기초합니다.
JumpStart Technology: Effective Use in the Solaris Operating Environment

`Audit` 디렉토리

이 디렉토리에는 시스템이 정의된 보안 프로파일 또는 감사 스크립트 세트를 준수하는지 평가하는 감사 스크립트가 들어있습니다. 이 디렉토리의 스트립트는 다음 범주로 구성됩니다.

- 사용 불가
- 사용 가능
- 설치
- 최소화
- 인쇄
- 제거
- 설정
- 업데이트

이러한 각 범주에 있는 스크립트의 상세한 목록과 각 스크립트의 설명에 대해서는 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.

`Documentation` 디렉토리

이 디렉토리는 `README` 파일과 같은 사용자를 위한 정보 텍스트 파일을 포함합니다.

man 디렉토리

이 디렉토리에는 명령, 기능 및 드라이버에 대한 man 페이지의 절의 하위 디렉토리가 들어있습니다. 이 디렉토리는 또한 특별히 제공된 명령 색인인 `windex` 파일을 포함하고 있습니다.

이런 man 페이지에 대한 상세한 정보는 man 페이지 자체 또는 *Solaris Security Toolkit 4.1 Man Page Guide*를 참조하십시오.

Drivers 디렉토리

이 디렉토리에는 Solaris Security Toolkit 소프트웨어를 실행할 때 실행 및 설치되는 파일을 지정하는 구성 정보 파일이 들어있습니다. 이 디렉토리에는 드라이버, 스크립트 및 구성 파일이 들어있습니다.

다음은 `Drivers` 디렉토리에 있는 드라이버 및 스크립트의 예입니다.

- `common_{log|misc}.funcs`
- `config.driver`
- `desktop-{config|hardening|secure}.driver`
- `driver.{funcs|init|run}`
- `hardening.driver`
- `finish.init`
- `install-Sun_ONE-WS.driver`
- `jumpstart-{config|hardening|secure}.driver`
- `secure.driver`
- `starfire-{config|hardening|secure}.driver`
- `suncluster3x-{config|hardening|secure}.driver`
- `sunfire_15k_domain-{config|hardening|secure}.driver`
- `sunfire_15k_sc-{config|hardening|secure}.driver`
- `sunfire_mf_msp-{config|hardening|secure}.driver`
- `undo.{funcs|init|run}`
- `hardening.driver`
- `user.init.SAMPLE`
- `user.run.SAMPLE`
- `audit_{private|public}.funcs`

모든 제품 관련 드라이버 및 일부 다른 드라이버는 각 드라이버에 대해 다음 세 가지 파일을 포함합니다.

- *`name`*`-secure.driver`
- *`name`*`-config.driver`
- *`name`*`-hardening.driver`

이 세 파일은 이전 목록에서 대괄호 안에 표시됩니다(예: `sunfire_15k_sc-{config|hardening|secure}.driver`). 이러한 파일은 완전성을 위해 나열됩니다. 드라이버를 실행하려는 경우에는 *`name`*`-secure.driver`만 사용하십시오. 해당 드라이버가 자동으로 관련 드라이버를 호출합니다.

Solaris Security Toolkit 구조는 실제 스크립트 자체를 수정하지 않지만 드라이버, 종료 및 감사 스크립트가 여러 환경에서 사용될 수 있도록 하는 구성 정보를 포함하고 있습니다. 종료 및 감사 스크립트에서 사용된 모든 변수는 구성 파일 세트에 유지됩니다. 이러한 구성 파일은 드라이버가 가져오며, 드라이버가 호출시 종료 및 감사 스크립트에서 변수를 사용할 수 있습니다.

Solaris Security Toolkit 소프트웨어에는 아래 세 개의 기본 구성 파일이 있는데, 모두가 `Drivers` 디렉토리에 저장됩니다.

- `driver.init`
- `finish.init`
- `user.init`

드라이버가 호출하는 종료 스크립트는 Finish 디렉토리에 있습니다. 드라이버가 호출하는 감사 스크립트는 Audit 디렉토리에 있습니다. 드라이버가 설치하는 파일은 Files 디렉토리에서 읽습니다. 종료 및 감사 스크립트에 대한 자세한 정보는 이 설명서의 해당 장을 참조하십시오.

그림 1-2는 드라이버 제어 흐름의 흐름도를 보여줍니다.

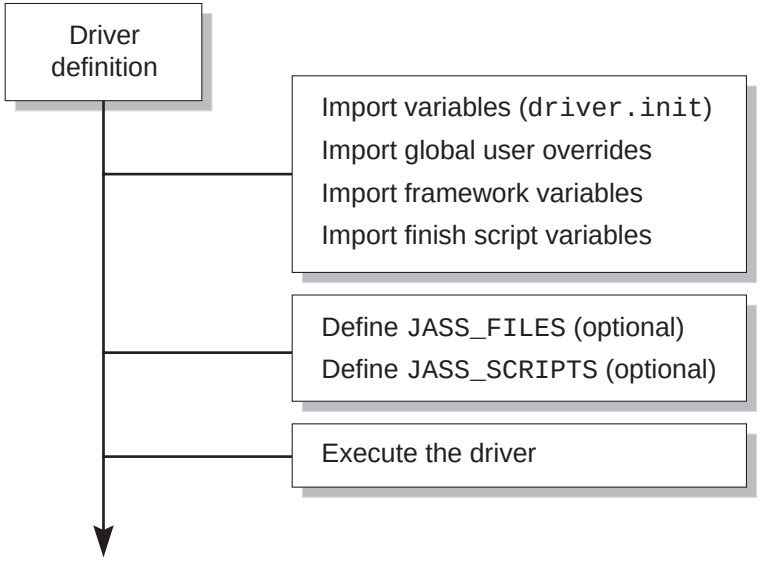

그림 **1-2** 드라이버 제어 흐름

첫 번째로 다양한 `.init` 파일에서 모든 환경 변수를 가져옵니다. 이것이 완료된 후 드라이버가 2부로 이동하는데, 여기에서는 `JASS_FILES` 및 `JASS_SCRIPTS`를 정의합니다. 이들의 정의는 선택적입니다. 즉, 하나의 환경을 정의하거나 둘 다 정의하거나 둘 다 정의하지 않을 수 있습니다. 드라이버의 3부는 `driver.run`을 호출하여 `JASS_FILE` 및 `JASS_SCRIPTS` 환경 변수에 의해 정의되는 작업을 수행합니다.

코드 예 1-1은 드라이버의 제어 흐름을 나타냅니다.

코드 예 **1-1** 드라이버 제어 흐름

```
DIR="'/bin/dirname $0'"

export DIR
. ${DIR}/driver.init

JASS_FILES="
                          /etc/cron.d/cron.allow
                          /etc/default/ftpd
                          /etc/default/telnetd
"

JASS_SCRIPTS="
                          install-at-allow.fin
                          remove-unneeded-accounts.fin
"
. ${DIR}/driver.run
```

이 코드 예제는 DIR 환경 변수를 설정하고 내보내서 드라이버가 시작 디렉토리를 인식하게 합니다. 다음, JASS_FILES 환경 변수가 클라이언트의 JASS_HOME_DIR/Files 디렉토리에서 복사되는 파일을 포함하도록 정의됩니다. 그런 다음 JASS_SCRIPTS 환경 변수가 Solaris Security Toolkit 소프트웨어에 의해 실행되는 종료 스크립트로 정의됩니다. 마지막으로 driver.run 드라이버를 호출하여 강화 작업의 실행이 시작됩니다. 일단 호출되면, driver.run은 JASS_FILES에 의해 지정되는 파일을 복사하고, JASS_SCRIPTS로 지정되는 스크립트를 실행합니다.

Files 디렉토리

이 디렉토리는 JASS_FILES 환경 변수와 driver.run 스크립트에 의해 사용됩니다. 이 디렉토리는 JumpStart 클라이언트로 복사되는 파일을 저장합니다.

다음 파일이 이 디렉토리에 있습니다.

- /.cshrc
- /.profile
- /etc/default/sendmail
- /etc/dt/config/Xaccess
- /etc/hosts.{allow|deny}
- /etc/init.d/nddconfig
- /etc/init.d/set-tmp-permissions
- /etc/init.d/sms_arpconfig
- /etc/init.d/swapadd
- /etc/issue
- /etc/motd

- /etc/notrouter
- /etc/rc2.d/S00set-tmp-permissions
- /etc/rc2.d/S07set-tmp-permissions
- /etc/rc2.d/S70nddconfig
- /etc/rc2.d/S73sms_arpconfig
- /etc/rc2.d/S73swapadd
- /etc/security/audit_class
- /etc/security/audit_control
- /etc/security/audit_event
- /etc/sms_domain_arp
- /etc/sms_sc_arp
- /etc/syslog.conf

Finish 디렉토리

이 디렉토리에는 설치 중에 시스템 수정 및 업데이트를 수행하는 종료 스크립트가 들어 있습니다. 이 디렉토리의 스트립트는 다음 범주로 구성됩니다.

- 사용 불가
- 사용 가능
- 설치
- 최소화
- 인쇄
- 제거
- 설정
- 업데이트

이러한 각 범주에 있는 스크립트의 상세한 목록과 각 스크립트의 설명에 대해서는 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.

OS 디렉토리

이 디렉토리에는 Solaris OS 이미지만 들어있습니다. 이들은 클라이언트 설치의 소스로서 JumpStart 소프트웨어 설치 프로세스에 의해 사용되며 `add_install_client` 및 `rm_install_client` 스크립트를 제공합니다. `add_client` 스크립트가 이들 추가 디렉토리 이름을 받아들입니다.

Solaris OS 이미지 로드 및 수정에 대한 자세한 정보는 Sun BluePrints 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

표준 설치 이름 지정 규칙을 따릅니다.

Solaris OS

Solaris OS의 경우 다음 이름 지정 표준 규칙을 사용하십시오.

Solaris_*os* 버전_*CD* 릴리스의 *4*자리 숫자 년도_*2*자리 숫자 월

예를 들어 2001년 4월자 Solaris 8 운영 환경 CD는 Solaris_8_2001-04의 디렉토리 이름을 갖습니다. Solaris OS의 업데이트와 릴리스를 구분하여 테스트 및 전개 목적으로 매우 정교한 제어를 유지할 수 있습니다.

Trusted Solaris OS

Trusted Solaris의 경우 다음 디렉토리 이름 지정 표준을 사용하십시오.

Trusted_Solaris_*os* 버전_*CD* 릴리스의 *4*자리 숫자 년도_*2*자리 숫자 월

예를 들어 Trusted Solaris 소프트웨어 릴리스가 2000년 2월자인 경우 디렉토리 이름은 Trusted_Solaris_8_2000-02입니다.

Solaris OS Intel 플랫폼 개정판

Solaris OS Intel 플랫폼 개정판의 경우 아래 디렉토리 이름 지정을 사용하십시오.

Solaris_*os* 버전_*CD* 릴리스의 *4*자리 숫자 년도_*2*자리 숫자 월_ia

예를 들어 Solaris OS Intel 플랫폼 개정판 릴리스가 2001년 4월자인 경우 디렉토리 이름은 Solaris_8_2001-04_ia입니다.

Packages 디렉토리

이 디렉토리에는 종료 스크립트로 설치할 수 있는 소프트웨어 패키지가 들어있습니다. 예를 들어, Sun Java™ System Web Server(이전의 Sun™ ONE Web Server 및 iPlanet ™ Web Server), 소프트웨어 패키지를 Packages 디렉토리에 저장할 수 있으므로 적절한 종료 스크립트가 해당 소프트웨어를 필요시 설치합니다.

Solaris Security Toolkit 소프트웨어에 포함되어 있는 여러 종료 스크립트가 소프트웨어 설치 및 기본 구성 기능을 수행합니다. Packages 디렉토리에서 소프트웨어를 설치하는 스크립트에는 다음이 포함됩니다.

- install-fix-modes.fin
- install-Sun_ONE-WS.fin
- install-jass.fin
- install-md5.fin
- install-openssh.fin

Patches 디렉토리

이 디렉토리는 Solaris OS용 Recommended and Security Patch Clusters 저장을 위한 것입니다. 필수 패치를 이 디렉토리에 다운로드하고 추출하십시오.

이 디렉토리에 패치를 저장하고 추출함으로써 설치 능률을 올릴 수 있습니다. 패치가 이 디렉토리로 추출될 때 Solaris Security Toolkit 소프트웨어의 패치 설치 스크립트가 설치를 자동화하므로 각 시스템 설치를 위해 패치 클러스터를 수동으로 추출할 필요가 없게 됩니다.

사용되는 각 Solaris OS 버전에 대한 하위 디렉토리를 작성하십시오. 예를 들어 `Patches` 디렉토리 안에 `2.5.1_Recommended` 및 `2.6_Recommended` 디렉토리를 가질 수 있습니다.

Solaris Security Toolkit 소프트웨어는 Solaris OS 인텔 플랫폼 개정판 패치 클러스터를 지원합니다. 이러한 패치 클러스터에 대한 지원되는 이름 지정 규칙은 SunSolve OnLine℠ 서비스를 통해 사용 가능한 규칙과 동일합니다.

형식은 `Solaris_<release>_x86_Recommended`입니다. Solaris 8 OS용 Solaris OS Intel 플랫폼 개정판 패치 클러스터는 `Solaris_8_x86_Recommended`라는 이름의 디렉토리에 있습니다.

`Profiles` 디렉토리

이 디렉토리에는 모든 JumpStart 프로파일이 들어있습니다. 이들 프로파일에는 설치할 Solaris OS 클러스터(예: 핵심, 일반 사용자, 개발자 또는 전체 배포), 디스크 배치 및 수행할 설치 유형(예: 독립형)을 판별하기 위해 JumpStart 소프트웨어가 사용하는 구성 정보가 들어있습니다.

JumpStart 프로파일은 특정 시스템 또는 시스템 그룹이 구축되는 방법을 정의하기 위해 `rules` 파일에 나열되고 사용됩니다.

`Sysidcfg` 디렉토리

`Profiles` 디렉토리와 비슷하게 `Sysidcfg` 디렉토리에는 JumpStart 모드 설치 중에만 사용되는 파일이 들어있습니다. 이들 파일은 필수 설치 정보를 제공하여 Solaris OS 설치를 자동화합니다. 별도의 디렉토리 트리가 OS에 특정한 정보를 저장합니다.

각 Solaris OS는 개별 디렉토리를 갖습니다. 각 릴리스에 대해 `Solaris_`*OS* 버전으로 이름이 지정되는 디렉토리가 있습니다. Solaris Security Toolkit 소프트웨어에 Solaris OS 버전 2.5.1이상 9이하를 위한 예제 `sysidcfg` 파일이 포함되어 있습니다.

예제 `sysidcfg` 파일을 네트워크, 호스트 등과 같은 다른 유형으로 확장할 수 있습니다. Solaris Security Toolkit 소프트웨어는 임의의 `sysidcfg` 파일을 지원합니다.

`sysidcfg` 파일에 대한 추가 예제는 Sun BluePrints 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

데이터 저장소

데이터 저장소는 Solaris Security Toolkit 실행 취소 작업을 지원하고 각 작업이 실행되는 방법에 대한 데이터를 저장하고 소프트웨어가 수정한 파일의 목록을 유지하고 실행 로그 데이터를 저장하는 `JASS_REPOSITORY` 디렉토리의 환경 변수입니다. 실행 취소 기능은 데이터 저장소에 저장된 정보에 의존합니다.

버전 제어 유지

Solaris Security Toolkit 소프트웨어가 사용하는 모든 파일 및 스크립트에 대한 버전 제어 유지는 두 가지 이유 때문에 중요합니다. 첫째, 이 환경의 목표중 하나는 시스템 설치를 다시 작성할 수 있다는 것입니다. 이 목표는 설치시 사용된 모든 파일 버전의 스냅샷이 없으면 불가능합니다. 둘째, 이러한 스크립트는 다수의 조직에 중요한 프로세스인 보안 기능을 수행 중이므로 적절하고 테스트된 변경만 적용되도록 최대한 주의를 기울여야 합니다.

원시 코드 제어 시스템(SCCS) 버전 제어 패키지가 Solaris OS `SUNWsprot` 패키지에서 제공됩니다. 프리웨어 및 상업용 제품으로 사용할 수 있는 기타 버전 제어 소프트웨어를 사용하여 버전 정보를 관리할 수 있습니다. 버전 제어 제품의 이름에 상관없이 업데이트를 관리하고 향후 시스템 재작성을 위해 버전 정보를 기록해두십시오.

버전 제어 이외에 무결성 관리 솔루션을 사용하여 파일의 내용이 수정되었는지 확인하십시오. 시스템 권한이 있는 사용자는 버전 제어 시스템을 우회할 수 있지만 원격 시스템에 무결성 데이터베이스를 유지하는 무결성 관리 시스템을 쉽게 우회할 수는 없습니다. 지역적으로 저장되는 데이터베이스는 악의적으로 수정될 수 있기 때문에 무결성 관리 솔루션은 중앙 집중될 때 최고의 성능을 갖습니다.

지원되는 Solaris OS 버전 실행

Solaris Security Toolkit 소프트웨어에 대한 Sun 지원은 Solaris 8 및 Solaris 9 운영 체제에서 사용할 때만 가능합니다. 해당 소프트웨어가 Solaris 2.5.1, Solaris 2.6 및 Solaris 7 운영 체제에서 사용될 경우, Sun 지원은 사용할 수 없습니다.

Solaris Security Toolkit 소프트웨어는 자동으로 설치된 Solaris 운영 체제 소프트웨어의 버전을 감지한 후 운영 체제 버전에 적합한 작업을 실행합니다.

지원되는 Solaris SMS 버전 실행

시스템 제어(SC)를 관리하기 위해 System Management Services(SMS)를 사용할 경우, SMS 버전 1.3에서 버전 1.4.1까지 사용시 Solaris Security Toolkit 4.1 소프트웨어에 대해 Sun 지원을 사용할 수 있습니다.

Solaris Security Toolkit 소프트웨어 구성 및 사용자 정의

Solaris Security Toolkit 소프트웨어에는 Sun BluePrints 설명서 *Enterprise Security: Solaris Operating Environment Security Journal, Solaris Operating Environment Versions 2.5.1, 2.6, 7, and 8* 및 보안 관련 Sun BluePrints OnLine 기사의 모든 보안 지침을 구현하는 스크립트, 프레임워크 기능 및 변수에 대한 기본값이 들어있습니다. 이러한 설정값이 모든 시스템에 적합한 것은 아니므로, Solaris Security Toolkit 소프트웨어를 사용자 정의하여 사용 시스템의 보안 요구사항에 맞도록 합니다.

Solaris Security Toolkit 소프트웨어의 가장 중요한 특성 중 하나는 사용자의 환경, 시스템 및 요구사항에 맞도록 쉽게 사용자 정의할 수 있다는 점입니다. Solaris Security Toolkit 소프트웨어를 사용자 정의하려면 드라이버, 종료 스크립트, 감사 스크립트, 프레임워크 기능, 환경 변수 및 파일 템플리트를 통해 소프트웨어의 활동을 조정하십시오.

대부분의 사용자는 Solaris Security Toolkit 코드를 수정할 필요가 없습니다. 임의로 변경하면 지원 가능성 및 업그레이드에 부정적인 영향을 줄 수도 있습니다. 사용자 환경에서 Solaris Security Toolkit 소프트웨어를 사용하기 위해 코드 수정이 절대적으로 필요한 경우 코드를 고유한 파일 또는 기능 이름으로 복사하여 13페이지의 "지침"에서와 같이 변경사항을 쉽게 추적할 수 있게 하십시오.

이 안내서에서 Solaris Security Toolkit 소프트웨어 사용자 정의에 대한 안내 및 지침은 각 장에서 해당될 경우 제공됩니다. 드라이버 사용자 정의에 대해 유용한 정보를 찾으려면 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오. 사용자 정의에는 파일 또는 변수 수정 및 작성이 포함됩니다.

다음 장에서 Solaris Security Toolkit 소프트웨어 사용자 정의 예제를 제공합니다. 예제에서 Solaris Security Toolkit 소프트웨어를 사용자 정의할 수 있는 몇 가지 방법을 알려주지만, 그 이외에도 많은 방법이 있습니다.

다음 절은 Solaris Security Toolkit 소프트웨어의 사용자 정의 전에 명확히 이해해야 할 정보를 나타냅니다. 이 정보는 많은 전개에서 수집되는 공유된 경험을 바탕으로 하므로 일반적인 함정을 피할 수 있습니다.

방침 및 요구사항

Solaris Security Toolkit 소프트웨어를 사용자 정의 및 전개 시, 적절한 계획을 세우면 결과 플랫폼 구성이 올바르고 조직의 기대를 부응할 수 있습니다.

계획 수립 단계에서 보안 방침 및 표준사항, 산업 규정 및 지침, 그리고 공급업체가 제공하는 바람직한 실례를 포함한 다양한 소스로부터 정보를 얻으십시오.

이 정보 이외에 구성의 결과가 플랫폼의 비즈니스 기능 제공 능력에 영향을 주지 않도록 적용 및 작업 요구사항을 고려하는 것이 필수적입니다.

지침

Solaris Security Toolkit 소프트웨어를 사용자 정의할 때 다음 지침을 고려하십시오. 이들 지침을 이해하고 준수하는 것이 전개를 유지하는 프로세스를 훨씬 더 쉽고 보다 효율적으로 만드는 데 도움이 됩니다.

- 일반적인 규칙으로서 Solaris Security Toolkit 소프트웨어와 함께 제공되는 어떤 원본 파일(드라이버, 스크립트, 파일 등)을 절대 수정하지 마십시오. 원본 파일을 변경하면 원본 파일에 대한 모든 변경이 파일의 새 버전에 의해 겹쳐 써질 수 있기 때문에 Solaris Security Toolkit 소프트웨어의 최신 버전으로 업그레이드하는 조직의 기능을 억제하고 제한합니다. (모든 사용자 정의 변경사항이 손실되고 시스템의 구성이 원하지 않는 방식으로 변경될 수 있습니다.) 파일을 사용자 정의하려면, 사본을 만들어서 원본은 그대로 두고 사본을 수정하십시오. 이 지침에 대한 단 세 가지 예외는 `sysidcfg` 파일, Files 디렉토리의 템플리트 및 Sun BluePrints OnLine 기사에 의해 지시되는 경우입니다.
- 드라이버 또는 스크립트의 사본에 적당한 이름을 지정하여 원본과 구별할 수 있게 하십시오. 스크립트의 목적을 표시하는 접두어나 키워드를 사용하십시오. 예를 들어 회사의 이름이나 증권 기호, 부서 식별자 심지어 플랫폼이나 응용 프로그램 유형을 포함하는 접두어가 모두 탁월한 이름 지정 표준입니다. 표 1-1은 몇 가지 이름 지정 표준의 예입니다.

표 **1-1** 사용자 정의 파일에 대한 이름 지정 표준

사용자 정의 파일	이름 지정 표준
`abccorp-secure.driver`	회사 접두어
`abcc-nj-secure.driver`	회사 증권 기호, 위치
`abbcorp-nj-webserver.driver`	회사, 위치, 응용 프로그램 유형
`abc-nj-trading-webserver.driver`	회사, 위치, 조직, 응용 프로그램 유형

■ 다음 Solaris Security Toolkit 파일이 사용자 시스템에 적합한지 검토하십시오. 이들 파일을 사용자 정의하려면 사본의 이름을 `user.init` 및 `user.run`으로 바꾼 후 사본의 내용을 수정 또는 추가하십시오.

`Drivers/user.init.SAMPLE`	전역 매개변수 사용자 정의에 사용됩니다.
`Drivers/user.run.SAMPLE`	전역 기능 사용자 정의에 사용됩니다.

■ 필요한 경우 다음 원본 파일을 수정하십시오. 이러한 파일은 사용자가 직접 수정해야 하는 유일한 원본 Solaris Security Toolkit 파일입니다.

`Sysidcfg/*/sysidcfg`	JumpStart 자동 구성에 사용됩니다.
`Files/*`	파일 템플리트로 사용되고 시스템에 복사됩니다.

주 – `SUNWjass`가 `pkgrm` 명령을 사용하여 제거되는 경우 `user.init` 및 `user.run` 파일은 작성된 경우에도 제거되지 않음을 명심하십시오. 이 작동은 Solaris Security Toolkit 디렉토리 구조에 추가되고 분배에는 포함되지 않는 모든 고객 파일의 경우에 발생합니다. Solaris Security Toolkit 분배에 포함되는 Files 디렉토리의 파일과 `sysidcfg` 파일이 존재하며 따라서 제거해야 합니다.

2장

시스템 보안: 방법론 적용

이 장은 시스템 보안 방법론을 제공합니다. Solaris Security Toolkit 소프트웨어를 사용하여 시스템을 보안하기 전에 적용할 수 있는 프로세스를 제공합니다.

이 장은 다음 주제를 다룹니다.

계획 및 준비

적절한 계획 수립은 Solaris Security Toolkit 소프트웨어를 사용하여 시스템을 성공적으로 보안하기 위한 핵심입니다. 계획 단계는 시스템의 응용 프로그램 및 작동 요구사항뿐 아니라 조직의 보안 방침 및 표준을 기초로 하여 시스템을 위한 Solaris Security Toolkit 프로파일을 구성합니다. 이 단계는 다음 작업으로 구분됩니다.

이 설명서에서는 다루지 않지만 이 단계에 대한 기타 고려사항에는 위험 및 노출 이해, 기반구조 및 그의 보안 요구사항 이해 및 책임, 로깅 및 사용 감사가 포함될 수 있습니다.

위험 및 수익 고려

이 절은 시스템을 보안하기 전에 명확하게 이해해야 하는 고려사항을 제공합니다. 위험과 수익을 주의깊게 검토하여 어떤 활동이 사용자 조직에 적합한지 판별하십시오.

시스템을 강화할 때 Solaris Security Toolkit 소프트웨어가 구현된 후 시스템이 기능하도록 보장하기 위해 특별한 예방 조치를 취해야 합니다. 게다가 모든 정지 시간을 가능한 최소화하기 위해 프로세스를 최적화하는 것이 중요합니다.

주 – 전개된 시스템을 보안할 때는 일부 경우에 조직이 시스템을 재구축하고 설치 시에 시스템을 강화한 후 작동에 필요한 모든 소프트웨어를 재로드하는 것이 훨씬 효과적일 수 있습니다.

---

1. 시스템에 대한 서비스 및 응용 프로그램의 요구사항을 이해하십시오.

   Solaris Security Toolkit 소프트웨어를 실행하기 전에 시스템에서 실행 중인 서비스와 응용 프로그램을 식별해야 합니다. 서비스 및 응용 프로그램과 연관된 모든 종속성을 열거하여 Solaris Security Toolkit 소프트웨어의 구성을 충분히 조정할 수 있게 해야 합니다. 그렇게 하지 않으면 필요한 서비스를 사용할 수 없거나 시작하지 못할 수 있습니다. Solaris Security Toolkit 소프트웨어가 수행한 변경 사항이 대부분의 경우에 실행 취소될 수 있지만, 설치 전에 올바른 프로파일을 개발하면 Solaris Security Toolkit 소프트웨어 구현과 연관된 잠재적 정지 시간을 제한하게 됩니다.

2. 시스템을 오프라인한 후 재부팅해야 합니다.

   Solaris Security Toolkit 소프트웨어 변경이 효력을 발생하려면 시스템을 재부트해야 합니다. 시스템의 건전성, 시스템이 제공하는 서비스 및 유지보수 창의 가용성에 따라서 조직은 소프트웨어 구현에 어려움을 겪을 수 있습니다. 보안을 강화하지 않았을 때의 위험에 대해 정지 시간의 비용을 신중히 평가하여 결정해야 합니다.

3. 기능성을 검증하기 위해 여러 번의 시스템 재부트가 필요할 수 있습니다.

   임무 결정적 설정으로 시스템을 구현하기 전에 비생산 시스템에서 모든 변경을 수행하십시오. 이것이 항상 가능하지는 않습니다. 예를 들어 대상 환경을 효과적으로 미러링하는 충분한 하드웨어 또는 소프트웨어의 부족으로 인해 불가능할 수 있습니다. Solaris Security Toolkit 소프트웨어의 설치 전후에 테스트를 수행해야 합니다. 시스템이 강화된 후에 문제점 해결이 필요한 식별되지 않은 종속성이 여전이 존재할 수 있습니다. 대부분의 경우에 이들 문제는 이 장에서 설명하는 기법을 사용하여 상당히 빨리 해결할 수 있습니다. Solaris Security Toolkit 소프트웨어 설치 후에 기능성 문제점이 발견되는 경우 Solaris Security Toolkit 소프트웨어의 효과를 실행 취소하거나 누락된 기능성을 지원하고 활성화하는 시스템의 보안 구성을 더 변경하기 위해 추가로 플랫폼을 재부트해야 할 수도 있습니다.

4. 플랫폼 보안은 단순한 강화 및 최소화보다 많은 것을 요구합니다.

   보안 상태를 향상시키기 위해 시스템 구성을 업데이트할 때, 플랫폼 강화와 최소화는 시스템, 서비스 및 데이터를 보호하기 위해 수행할 수 있고 수행해야 미소한 부분이라는 것을 알아야 합니다. 추가 대책 및 제어 처리는 이 문서의 범위를 벗어나지만, 계정 관리, 권한 관리, 파일 시스템과 데이터 무결성, 호스트 기반 액세스 제어, 침입 보호, 취약점 검색 및 분석 그리고 응용 프로그램 보안과 관련된 문제를 고려하는 것이 좋습니다.

5. 시스템이 이미 부당하게 이용되었거나 이용 가능한 취약점을 가질 수 있습니다.

강화될 플랫폼이 이미 공격자에 의해 부당하게 이용되었을 수 있습니다. Solaris Security Toolkit 소프트웨어가 너무 늦게 구현되어 부당하게 이용된 취약점에 대한 보호를 제공하지 못할 수 있습니다. 이 경우 시스템을 재설치한 후 Solaris Security Toolkit 소프트웨어를 사용하여 보안을 개선하십시오.

보안 방침, 표준 및 관련 문서 검토

시스템 보안에서의 첫 번째 작업은 조직의 관련 보안 방침, 표준 및 플랫폼 보안에 관한 지침을 이해하는 것입니다. 이들 문서가 조직의 모든 시스템에 대해 따라야 하는 요구 사항 및 관례를 전달하기 때문에 이들 문서를 사용하여 Solaris Security Toolkit 프로파일의 토대를 준비하십시오. 조직에 이런 문서가 없는 경우 해당 문서를 개발하는 것이 Solaris Security Toolkit 소프트웨어를 사용자 정의하는 능력을 향상시킵니다.

주 – 이러한 문서를 찾을 때 일부 자료는 최상의 용례나 다른 문서에 나열될 수 있음을 명심하십시오.

보안 방침에 대한 상세한 정보는 Sun BluePrints OnLine 기사 "Developing a Security Policy"를 참조하십시오. 이 문서를 사용하여 보안 방침이 조직의 보안 계획에서 수행하는 역할에 대해 잘 이해할 수 있습니다.

다음 두 예제는 방침 선언이 Solaris Security Toolkit 프로파일이 구성되는 방법에 직접적으로 영향을 미칠 수 있는 방법을 보여줍니다.

예 1

- 방침 - 조직은 사용자의 강력한 인증 및 전송된 데이터의 암호화를 지원하는 관리 프로토콜을 사용해야 합니다.
- 프로파일 영향 - 평문 프로토콜인 Telnet, FTP, SNMPv1 및 기타 프로토콜은 사용하면 안됩니다. 기본적으로 Solaris Security Toolkit은 그런 서비스를 비활성화하므로 추가 구성은 필요하지 않습니다.

주 – 텔넷 및 FTP 서비스는 둘 다 Kerberos 같은 확장을 사용하여 더욱 강력한 인증 및 암호화를 지원하도록 구성할 수 있습니다. 이들 서비스가 예제로서 나열되지만 기본 구성은 이러한 추가 보안 레벨을 지원하지 않습니다.

예 2

방침 - 모든 사용자는 30일마다 암호를 변경해야 합니다.

프로파일 영향 -암호 사용 기간을 사용할 수 있도록 Solaris Security Toolkit 소프트웨어를 구성할 수 있습니다. 기본적으로 Solaris Security Toolkit 소프트웨어는 최대 암호 사용 기간을 8주(56일)로 설정합니다. 이 방침을 준수하기 위해서 Solaris Security Toolkit 소프트웨어의 프로파일을 변경해야 합니다. *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.

Solaris Security Toolkit 소프트웨어가 시스템에서 실행될 때 기본적으로 암호 사용 기간을 사용할 수 있지만 이 변경이 기존 사용자에게는 영향을 주지 않습니다. 기존 사용자에 대해 암호 사용 기간을 사용하려면 각 사용자 계정에서 `passwd`(1) 명령을 호출하십시오.

응용 프로그램 및 서비스 요구사항 판별

이 작업은 시스템이 강화된 후에도 서비스를 사용할 수 있도록 보장합니다. 이 작업은 다음 단계로 구성됩니다.

- 18페이지의 "응용 프로그램 및 작동 서비스 목록 명세 식별"
- 18페이지의 "서비스 요구사항 판별"

응용 프로그램 및 작동 서비스 목록 명세 식별

응용 프로그램, 서비스 및 작동 또는 관리 기능의 목록을 작성하십시오. 이 목록은 시스템에서 실제로 사용되고 있는 소프트웨어를 판별하기 위해 필요합니다. 많은 경우 시스템은 실제 사용하는 것보다 많은 소프트웨어와 비즈니스 기능을 지원하지 않는 소프트웨어로 구성되어 있습니다.

시스템은 가능하면 항상 최소한으로 구성되어야 합니다. 즉 비즈니스 기능을 지원하기 위해 필요하지 않은 소프트웨어는 설치하지 않아야 합니다. 시스템에 있는 불필요한 소프트웨어 응용 프로그램은 공격자가 시스템을 부당하게 이용할 수 있는 기회를 높입니다. 또한 시스템에 소프트웨어가 많을수록 대개 적용해야 하는 패치도 많아집니다. Solaris OS 최소화에 대해서는 Sun BluePrints OnLine 기사 "Minimizing the Solaris Operating Environment for Security"를 참조하십시오.

소프트웨어 목록을 구축할 때 반드시 시스템에 상주하는 응용 프로그램 외에 관리, 모니터링 및 백업 소프트웨어 같은 기반구조 구성요소를 포함시키십시오.

서비스 요구사항 판별

응용 프로그램 및 서비스 목록 작성을 완료한 후 임의의 구성요소가 강화 프로세스에 의해 영향을 받을 수 있는 종속성을 갖는지 판별하십시오. 많은 타사 응용 프로그램은 Solaris OS가 제공하는 서비스를 직접 사용하지 않습니다. 해당 응용 프로그램에 대해 다음 절에서 도움이 되는 정보를 제공합니다.

- 19페이지의 "공유 라이브러리"
- 22페이지의 "구성 파일"

■ 22페이지의 "서비스 프레임워크"

공유 라이브러리

응용 프로그램을 지원하기 위해 필요한 라이브러리를 이해하는 것이 중요합니다. 이 지식은 디버깅 상황에서 가장 유용하지만 강화할 시스템을 준비할 때도 유용합니다. 시스템의 상태를 알 수 없을 때 소프트웨어 종속성 같은 문제를 명확히 이해하도록 가능한 많은 정보를 수집하십시오.

이러한 방법을 사용하여 설치하는 Solaris OS에 따라서 응용 프로그램이 사용하는 라이브러리를 판별할 수 있습니다.

- 첫 번째는 파일 시스템 개체(예: 응용 프로그램 이진)에 대해 사용됩니다.
- 두 번째는 실행 중인 응용 프로그램을 분석할 때 사용됩니다.
- 세 번째는 프로그램이 시작될 때 프로그램을 추적하는 데 사용됩니다.

예제: DNS 서버 소프트웨어를 지원하는 데 필요한 라이브러리를 판별하십시오.

파일 시스템 개체에 대한 정보를 수집하려면 /usr/bin/ldd 명령을 사용하십시오.

코드 예 **2-1** 파일 시스템 개체에 대한 정보 획득

```
# ldd /usr/sbin/in.named
libresolv.so.2 => /usr/lib/libresolv.so.2
libsocket.so.1 => /usr/lib/libsocket.so.1
libnsl.so.1 =>    /usr/lib/libnsl.so.1
libc.so.1 =>      /usr/lib/libc.so.1
libdl.so.1 =>    /usr/lib/libdl.so.1
libmp.so.2 =>    /usr/lib/libmp.so.2
/usr/platform/SUNW,Ultra-5_10/lib/libc_psr.so.1
```

실행 중인 프로세스로부터 정보를 수집하려면 /usr/proc/bin/pldd 명령(Solaris OS 버전 8과 9에서 사용 가능)을 사용하십시오.

코드 예 **2-2** 실행 중 프로세스로부터 정보 수집

```
# pldd 20307
20307:  /usr/sbin/in.named
/usr/lib/libresolv.so.2
/usr/lib/libsocket.so.1
/usr/lib/libnsl.so.1
/usr/lib/libc.so.1
/usr/lib/libdl.so.1
/usr/lib/libmp.so.2
/usr/platform/sun4u/lib/libc_psr.so.1
/usr/lib/dns/dnssafe.so.1
/usr/lib/dns/cylink.so.1
```

pldd 명령은 응용 프로그램이 링크되는 라이브러리 외에 응용 프로그램이 동적으로 로드하는 공유 라이브러리를 표시합니다. 이 정보는 다음 truss 명령을 사용하여 수집할 수도 있습니다.

주 – 다음 출력은 간단하게 표시하기 위해 일부가 생략되었습니다.

코드 예 **2-3** 동적으로 로드된 응용 프로그램 식별

```
# truss -f -topen,open64 /usr/sbin/in.named
20357:  open("/usr/lib/libresolv.so.2", O_RDONLY)       = 3
20357:  open("/usr/lib/libsocket.so.1", O_RDONLY)       = 3
20357:  open("/usr/lib/libnsl.so.1", O_RDONLY)          = 3
20357:  open("/usr/lib/libc.so.1", O_RDONLY)            = 3
20357:  open("/usr/lib/libdl.so.1", O_RDONLY)           = 3
20357:  open("/usr/lib/libmp.so.2", O_RDONLY)           = 3
20357:  open("/usr/lib/nss_files.so.1", O_RDONLY)       = 4
20357:  open("/usr/lib/nss_files.so.1", O_RDONLY)       = 4
20357:  open("/usr/lib/dns/dnssafe.so.1", O_RDONLY)     = 4
20357:  open("/usr/lib/dns/cylink.so.1", O_RDONLY)      = 4
20357:  open("/usr/lib/dns/sparcv9/cylink.so.1", O_RDONLY) = 4
```

이 출력 버전에는 시스템 호출의 반환값 뿐만 아니라 프로세스 ID, 시스템 호출(이 경우에 open) 및 그의 인수가 들어있습니다. 반환값을 사용하면 시스템 호출이 공유 라이브러리를 찾아서 여는 데 성공한 시기가 명백합니다.

공유 라이브러리 목록이 잘 알려지면 다음 명령을 사용하여 해당 라이브러리가 속하는 Solaris OS 패키지를 판별하십시오.

```
# grep '/usr/lib/dns/cylink.so.1' /var/sadm/install/contents
/usr/lib/dns/cylink.so.1 f none 0755 root bin 63532 24346 \
  1018126408 SUNWcsl
```

결과 출력은 이 공유 라이브러리가 SUNWcsl(코어, 공유 라이브러리) 패키지에 속함을 표시합니다. 이 프로세스는 응용 프로그램이나 서비스를 지원하기 위해 필요한 패키지를 식별하는 데 도움이 되기 때문에 특히 플랫폼 최소화를 수행할 때 유용합니다.

구성 파일

요구사항을 수집하는 다른 방법은 구성 파일을 통하는 것입니다. 이 프로세스는 구성 파일을 제거하거나 이름을 바꾸어서 서비스를 비활성화 할 수 있기 때문에 시스템이 강화되는 방법에 더욱 직접적인 영향을 미칩니다. 자세한 정보는 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.

구성 파일을 사용 중인지 판별하려면 `truss` 명령을 사용하십시오.

주 – 다음 출력은 간단하게 표시하기 위해 일부가 생략되었습니다.

코드 예 **2-4** 구성 파일을 사용 중인지 판별

```
# truss -f -topen,open64 /usr/sbin/in.named 2>&1 | \
grep -v "/usr/lib/.*.so.*"
20384:  open("/etc/resolv.conf", O_RDONLY)              = 3
20384:  open("/dev/conslog", O_WRONLY)                  = 3
20384:  open("/usr/share/lib/zoneinfo/US/Eastern", O_RDONLY) = 4
20384:  open("/var/run/syslog_door", O_RDONLY)          = 4
20384:  open("/etc/nsswitch.conf", O_RDONLY)            = 4
20384:  open("/etc/services", O_RDONLY)                 = 4
20384:  open("/etc/protocols", O_RDONLY)                = 4
20384:  open("/etc/named.conf", O_RDONLY)               = 4
20384:  open("named.ca", O_RDONLY)                      = 5
20384:  open("named.local", O_RDONLY)                   = 5
20384:  open("db.192.168.1", O_RDONLY)                  = 5
20384:  open("db.internal.net", O_RDONLY)               = 5
```

이 예제에서 DNS 서비스가 `/etc/named.conf` 같은 구성 파일을 사용합니다. 이전 예제에서와 같이 서비스의 반환값이 오류를 표시하는 경우 문제점이 있을 수 있습니다. 강화 전후의 결과를 주의하여 문서화하는 것이 전체 검증 프로세스를 가속화하는 데 도움이 됩니다.

서비스 프레임워크

이 범주는 더 크고 더 복잡한 응용 프로그램이 구축되는 프레임워크 또는 메타서비스를 포함합니다. 이 범주에서 일반적으로 발견되는 프레임워크의 유형은 이름 지정 서비스(예: NIS, NIS+ 및 LDAP), 인증 서비스(예: Kerberos 및 LDAP) 및 RPC 기능이 사용하는 포트 매핑 프로그램 같은 서비스입니다.

응용 프로그램이 이런 유형의 서비스에 종속할 때가 항상 명확하지는 않습니다. Kerberos 영역에 추가하는 작업과 같이 응용 프로그램을 구성하기 위해 특별한 조정이 필요한 경우 종속성이 알려집니다. 일부 경우에는 응용 프로그램 종속성이 어떤 추가 작업도 필요하지 않으며 공급 업체가 실제 종속성을 문서화하지 않을 수 있습니다.

그런 예 중의 하나가 RPC 포트 매핑 프로그램입니다. 기본적으로 Solaris Security Toolkit 소프트웨어는 RPC 포트 매핑 프로그램을 비활성화합니다. 이 조치가 이 서비스에 의존하는 다른 서비스에서 예기치 않은 작동을 유발할 수 있습니다. 과거 경험을 토대로, 응용 프로그램 코드가 예외 사례를 얼마나 잘 다루도록 작성되는지에 따라서 서비스가 취소, 중단 또는 실패합니다. 응용 프로그램이 RPC 포트 매핑 프로그램을 사용 중인지 확인하려면 `rpcinfo` 명령을 사용하십시오. 예를 들면,

코드 예 **2-5** RPC를 사용하는 응용 프로그램 판별

```
# rpcinfo -p
100000    3   tcp    111  rpcbind
100000    4   udp    111  rpcbind
100000    2   udp    111  rpcbind
100024    1   udp  32777  status
100024    1   tcp  32772  status
100133    1   udp  32777
100133    1   tcp  32772
100021    1   udp   4045  nlockmgr
100021    2   udp   4045  nlockmgr
100021    3   udp   4045  nlockmgr
100021    4   udp   4045  nlockmgr
100021    1   tcp   4045  nlockmgr
```

서비스 열은 `/etc/rpc` 파일의 정보 및/또는 LDAP 같이 구성된 이름 지정 서비스로 채워집니다.

타사 제품에 대한 경우에서 종종 그런 것처럼 이 파일에 서비스에 대한 항목이 없는 경우 서비스 필드가 비어 있을 수도 있습니다. 이로 인해 다른 응용 프로그램으로 등록한 응용 프로그램을 식별하기가 어렵습니다.

예를 들어 `rusers` 명령을 고려하십시오. 이 명령은 RPC 포트 매핑 서비스에 의존합니다. RPC 포트 매핑 프로그램이 실행하지 않는 경우 `rusers` 명령은 중지한 것으로 나타납니다. 프로그램은 결국 다음 오류 메시지와 함께 시간 초과됩니다.

```
# rusers -a localhost
localhost: RPC: Rpcbind failure
```

이 문제점은 프로그램이 서비스와 통신할 수 없기 때문에 발생합니다. 그러나 /etc/init.d/rpc로부터 RPC 포트 매핑 서비스를 시작한 후 프로그램이 즉시 결과를 산출합니다.

또 다른 예로서, RPC 포트 매핑 서비스가 실행 중이고 rusers 서비스가 실행하도록 구성되지 않은 경우를 고려하십시오. 이 경우에 완전히 다른 응답이 생성되며 상대적으로 검증하기 쉽습니다.

코드 예 **2-6** rusers 서비스 검증

```
# rusers -a localhost
localhost: RPC: Program not registered
# grep rusers /etc/rpc
rusersd         100002  rusers
# rpcinfo -p | grep rusers
<No output generated>
```

rpcinfo 명령이 rusers 서비스에 대한 레지스트리를 갖지 않는 경우 서비스가 실행하도록 구성되지 않았다고 가정하는 것이 안전합니다. 이 가정은 /etc/inet/inetd.conf에서 서비스 항목을 조사하여 검증됩니다.

```
# grep rusers /etc/inet/inetd.conf
# rusersd/2-3   tli     rpc/datagram_v,circuit_v     wait root
/usr/lib/netsvc/rusers/rpc.rusersd     rpc.rusersd
```

서비스 행의 시작부에 있는 주석 표시(#)는 rusers 서비스를 사용할 수 없음을 표시합니다. 서비스를 활성화 하려면 행의 주석 표시를 제거하고 다음과 같이 SIGHUP 신호를 /usr/sbin/inetd 프로세스로 보내십시오.

```
# pkill -HUP inetd
```

주 – pkill 명령은 Solaris OS 버전 7에서 9까지에서만 사용 가능합니다. 다른 버전의 경우, ps 및 kill 명령을 각각 사용하여 프로세스를 찾고 신호하십시오.

응용 프로그램이 RPC 기능을 사용하는지 판별하는 또 다른 방법은 앞에서 설명한 `ldd` 명령을 사용하는 것입니다.

코드 예 **2-7** RPC를 사용하는 응용 프로그램을 판별하기 위한 대체 방법

```
# ldd /usr/lib/netsvc/rusers/rpc.rusersd
libnsl.so.1 =>   /usr/lib/libnsl.so.1
librpcsvc.so.1 =>        /usr/lib/librpcsvc.so.1
libc.so.1 =>     /usr/lib/libc.so.1
libdl.so.1 =>    /usr/lib/libdl.so.1
libmp.so.2 =>    /usr/lib/libmp.so.2
/usr/platform/SUNW,Ultra-250/lib/libc_psr.so.1
```

`librpcsvc.so.1`에 대한 항목은 파일 이름과 함께 이 서비스가 RPC 포트 매핑 서비스에 의존함을 나타냅니다.

RPC 포트 매핑 프로그램 외에, 응용 프로그램이 FTP, SNMP 또는 NFS 같은 다른 일반 OS 서비스에 의존할 수 있습니다. 비슷한 방법을 사용하여 이들 서비스를 디버그하고 해당 서비스가 비즈니스 기능을 지원하기 위해 실제로 필요한지 판별할 수 있습니다. 한 가지 방법은 다음과 같이 `netstat` 명령을 사용하는 것입니다.

```
# netstat -a | egrep "ESTABLISHED|TIME_WAIT"
```

이 명령은 최근에 사용 중이거나 사용 중인 서비스의 목록을 반환합니다. 예를 들어,

표 **2-1** 최근에 사용 중인 서비스 목록

```
localhost.32827      localhost.32828      49152      0 49152      0
ESTABLISHED
localhost.35044      localhost.32784      49152      0 49152      0
ESTABLISHED
localhost.32784      localhost.35044      49152      0 49152      0
ESTABLISHED
localhost.35047      localhost.35046      49152      0 49152      0
ESTABLISHED
localhost.35046      localhost.35047      49152      0 49152      0
ESTABLISHED
filefly.ssh 192.168.0.3.2969    17615     1 50320     0 ESTABLISHED
```

이 예제에서 많은 서비스를 사용 중이지만 어떤 서비스나 응용 프로그램에 어떤 포트가 있는지 명확하지 않습니다. 이 정보는 `pfiles` 명령(Solaris OS 버전 8과 9에서 사용 가능)을 사용하는 프로세스를 조사하여 수집할 수 있습니다.

코드 예 **2-8** 서비스 또는 응용 프로그램에 있는 포트 판별

```
# for pid in `ps -aeo pid | grep -v PID`; do
> pfiles ${pid} | egrep "^${pid}:|sockname:"
> done
```

이러한 종속성을 판별하는 보다 효과적이고 효율적인 방법은 `lsof`(list open files) 명령을 사용하는 것입니다. 이 명령은 어떤 프로세스가 어떤 파일 및 포트를 사용 중인지 판별합니다. 예를 들어 이전 예제에서 어떤 프로세스가 포트 35047을 사용 중인지 판별하려면 다음 명령을 사용하십시오.

코드 예 **2-9** 파일 및 포트를 사용 중인 프로세스 판별

```
# ./lsof -i | grep 35047
ttsession   600 root 9u  IPv4 0x3000b4d47e8     0t1  TCP
localhost:35047->localhost:35046 (ESTABLISHED)
dtexec     5614 root 9u  IPv4 0x3000b4d59e8     0t0  TCP
localhost:35046->localhost:35047 (ESTABLISHED)
```

`lsof`의 출력은 포트 35047이 `dtexec`와 `ttsession` 프로세스 사이의 통신에 사용 중임을 나타냅니다.

`lsof` 프로그램을 사용할 때 파일 시스템 또는 네트워크 사용이 필요한 시스템간 또는 응용 프로그램간 종속성을 훨씬 더 빨리 판별할 수 있습니다. `lsof` 프로그램의 다양한 옵션을 사용하여 이 절에서 다루는 거의 모든 것을 포착할 수 있습니다.

`lsof` 프로그램을 확보하려면 다음 위치에서 다운로드 하십시오.

`ftp://vic.cc.purdue.edu/pub/tools/unix/lsof/`

주 – 종속성 판별을 설명하는 방법이 드물게 사용되는 항목은 찾지 못할 수 있습니다. 이 방법을 사용하는 것 외에 문서 및 공급 업체 문서를 검토하십시오.

Solaris Security Toolkit 프로파일 개발 및 구현

계획 및 준비 단계를 완료한 후 보안 프로파일을 개발 및 구현하십시오. 보안 프로파일은 강화 드라이버(예: *name*-hardening.driver) 및 사이트 특정 보안 방침을 구현하기 위한 모든 관련 드라이버, 스크립트 및 파일로 구성됩니다.

Solaris Security Toolkit 소프트웨어와 함께 제공되는 보안 프로파일 중 하나를 사용자 정의하거나 고유한 프로파일을 개발하십시오. 각 조직의 방침, 표준 및 응용 프로그램 요구사항은 아주 조금일지라도 서로 다릅니다.

보안 프로파일을 사용자 정의하려면 종료 스크립트, 감사 스크립트, 환경 변수, 프레임워크 기능 및 파일 템플리트를 통해 프로파일의 작동을 조정하십시오.

자세한 정보는 다음 장을 참조하십시오.

- 소프트웨어 사용자 정의에 대한 중요한 지침은 2 장, 12페이지의 "Solaris Security Toolkit 소프트웨어 구성 및 사용자 정의"를 참조하십시오.
- 보안 프로파일이 작성되는 예제 시나리오에 대해서는 2 장, 91페이지의 "보안 프로파일 작성"을 참조하십시오.
- 사용자 정의 드라이버에 대한 상세한 정보는 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.

  필요시 스크립트, 프레임워크 기능, 환경 변수 및 파일에 대한 정보는 *Solaris Security Toolkit 4.1 Reference Manual*의 적용 가능한 장을 참조하십시오. 사용자 정의할 수 있는 두 개의 핵심 환경 변수는 JASS_FILES와 JASS_SCRIPTS입니다.

플랫폼 고유의 차이를 고려하면서 대다수의 플랫폼에서 표준을 집행하려면 내포 또는 계층 구조 보안 프로파일이라고 부르는 방법을 사용하십시오. 자세한 정보는 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오. 결과 보안 프로파일을 조직의 방침, 표준 및 요구사항과 비교하여 우발적으로 또는 오류가 있는 변경이 작성되지 않도록 하십시오.

소프트웨어 설치

Solaris Security Toolkit 소프트웨어의 설치는 전개된 시스템 및 설치되고 있는 새 시스템 모두에 대해 동일합니다. 자세한 지시는 2 장을 참조하십시오.

전개된 시스템의 경우 소수의 특별한 경우가 이 프로세스를 더 간단하고 빠르게 만들 수 있습니다. 이러한 경우는 강화 프로세스에서 집중되지 않지만 설치 전 및 설치 후 작업에 집중됩니다.

설치 전 작업 수행

전개된 시스템을 강화하기 전에 백업 및 검증의 두 가지 중요한 작업을 고려하고 계획하십시오. 이들 작업은 전개된 시스템의 상태를 판별하고 시스템이 강화되기 전에 모든 잠재적 구성 문제점을 해결하는 데 도움이 됩니다.

데이터 백업

이 작업은 긴급 대책 계획에 집중합니다. 문제점이 있는 경우 시스템의 구성 및 데이터가 어떤 양식으로 보존되도록 보장하는 것이 필요합니다. 시스템을 백업하고 백업 매체를 읽을 수 있는지 확인하고 그의 내용을 복원할 수 있는지 검증해야 합니다. 시스템의 구성에 어떤 중대한 변경을 수행하기 전에 이 단계를 취하십시오.

시스템 안정성 확인

검증 작업은 백업 작업 만큼이나 중요합니다. 검증은 강화 프로세스에 의한 변경과 같이 모든 구성 변경의 구현 전에 시스템이 안정하고 작동 중인 상태에 있음을 보장합니다. 이 검증 프로세스는 응용 프로그램 또는 서비스의 성공적인 테스트 뒤에 오는 재부트를 포함합니다. 잘 정의된 테스트 및 승인 계획이 우선적이지만 문서화가 항상 가능하지는 않습니다. 그런 경우에는 시스템의 사용 방법에 기초하여 합리적인 방법으로 시스템을 테스트하십시오. 이 노력의 목표는 실행 중인 구성이 사실상 저장된 구성과 일치하도록 보장하는 것입니다.

시스템이 시동하거나 응용 프로그램이 시작할 때 표시되는 모든 오류 메시지나 경고를 조사하십시오. 오류를 정정할 수 없는 경우 강화 프로세스 중에 문제점의 잠재적인 원인으로 포함되지 않도록 오류를 로그하십시오. 로그 파일을 조사할 때 다음과 같은 시스템, 서비스 및 응용 프로그램 로그를 포함하십시오.

- `/var/adm/messages`
- `/var/adm/sulog`
- `/var/log/syslog`
- `/var/cron/log`

이 작업은 오류 또는 경고 메시지가 발생하지 않거나 알 수 없는 오류 또는 경고(알려진 모든 오류나 경고는 문서화되어 있음)가 발생하지 않고 시스템을 다시 시작할 수 있을 때 완료됩니다. 시스템은 알려지고 안정한 상태로 다시 시작해야 합니다. 검증 과정 중에 시스템의 실행 중인 구성 및 저장된 구성이 다르다는 것을 발견하는 경우 조직의 변경 제어 방침 및 프로세스를 다시 평가하여 해당 상태를 유발하는 차이를 식별하십시오.

설치 후 작업 수행

설치 후 작업은 설치 전 작업의 확장입니다. 작업의 목적은 강화 프로세스가 시스템이나 응용 프로그램에 어떤 새 결함도 초래하지 않도록 확인하는 것입니다. 이 작업은 주로 시스템 및 응용 프로그램 로그 파일을 검토하여 수행됩니다. 강화 및 후속 재부트 후에 작성되는 로그 파일은 시스템이 강화되기 전에 수집된 로그 파일과 비슷해야 합니다. 어떤 경우에는 더 적은 서비스가 시작되기 때문에 더 적은 메시지가 있을 수 있습니다. 가장 중요한 것은 새로운 오류 또는 경고 메시지가 없어야 합니다.

로그 파일 검토 외에, 일부 응용 프로그램은 로그 항목을 생성하지 않고 실패할 수 있기 때문에 기능성을 테스트하십시오. 자세한 검증 정보에 대해서는 다음 절을 참조하십시오.

응용 프로그램 및 서비스 기능성 확인

시스템 보안 프로세스의 마지막 작업은 시스템이 제공하는 응용 프로그램 및 서비스가 올바르게 기능하고 있는지 확인하는 것입니다. 이 작업은 또한 보안 프로파일이 보안 방침의 요구사항을 성공적으로 구현했음을 확인합니다. 모든 비정상 또는 문제점을 감지하고 바로 정정하려면 강화된 플랫폼의 재부트 후에 이 작업을 즉각 철저하게 수행하십시오. 이 작업은 보안 프로파일 설치 확인 및 응용 프로그램과 서비스 기능성 확인이라는 두 하위 작업으로 구분됩니다.

보안 프로파일 설치 확인

Solaris Security Toolkit 소프트웨어가 보안 프로파일을 오류 없이 올바르게 설치했는지 확인하기 위해 설치 로그 파일을 검토하십시오. 이 파일은 `JASS_REPOSITORY/jass-install-log.txt`에 설치됩니다.

주 – Solaris Security Toolkit 소프트웨어가 시스템에 수행한 작업을 이해하기 위해 이 로그 파일을 참조하십시오. 시스템의 각 실행에 대해 실행 시작 시간을 기초로 디렉토리에 저장되는 새로운 로그 파일이 있습니다.

프로파일의 설치를 확인하는 것 외에, 시스템의 보안 구성을 평가하십시오. 수동 검사를 수행하거나 도구를 사용하여 프로세스를 자동화하십시오.

응용 프로그램 및 서비스 기능성 확인

프로세스 응용 프로그램 및 서비스를 검증하기 위해 잘 정의된 테스트 및 승인 계획을 실행하십시오. 이 계획은 시스템 또는 응용 프로그램의 다양한 구성 요소를 사용하여 해당 구성 요소가 사용 가능하며 작업 순서대로 있는지 판별합니다. 이 계획을 사용할 수 없는 경우는 시스템의 사용 방법에 기초하여 합리적인 방법으로 시스템을 검사하십시오. 이 작업의 목적은 강화 프로세스가 자체 기능을 수행하는 응용 프로그램 또는 서비스 기능에 전혀 영향을 미치지 않음을 확인하는 것입니다.

시스템이 강화된 후 응용 프로그램 또는 서비스가 제대로 작동하지 않는 경우 응용 프로그램 로그 파일을 검토하여 문제점을 판별하십시오. 많은 경우에 `truss` 명령을 사용하여 응용 프로그램의 문제점이 있는 위치를 판별하십시오. 이 위치가 알려진 후에 해당 문제점을 대상으로 하여 Solaris Security Toolkit 소프트웨어가 변경한 내용을 다시 추적할 수 있습니다.

시스템 보안 유지보수

많은 조직의 공통적인 실수는 설치 중에만 보안에 신경 쓰고 그 후에는 거의 또는 전혀 다시 확인하지 않는 것입니다. 보안 유지보수는 지속적인 프로세스입니다. 주기적으로 시스템 보안을 검토하고 확인해야 합니다.

임의의 시스템에 대한 기본 보안 구성이 시간이 지남에 따라 점차 허술해지기 때문에 보안 시스템 유지보수는 경계가 필요합니다. 예를 들어 시스템 취약점이 알려집니다. 다음의 기본 지침에서 개요를 제공합니다.

- Solaris OS 패치는 설치의 일부로서 추가 소프트웨어 패키지를 설치할 수 있으며 시스템 구성 파일을 겹쳐 쓸 수 있습니다. 패치를 설치하기 전과 후에 시스템의 보안 상태를 검토하십시오. 또한 항상 시스템을 최신 패치로 업데이트하는 것이 중요합니다.

  Solaris Security Toolkit 소프트웨어가 한 시스템에서의 반복적인 실행을 지원하기 때문에 패치를 설치한 후 시스템을 보안할 수 있도록 사용자의 패치 적용을 도와줄 수 있습니다. 모든 패치 설치 후에 적용 가능한 드라이버와 함께 소프트웨어를 실행하여 구성이 정의된 보안 방침과 일관성을 유지하는지 확인하십시오. 또한 사용되는 Solaris Security Toolkit 소프트웨어의 버전이 설치된 패치에 의해 추가된 새로운 기능을 지원하지 않을 수 있기 때문에 시스템을 수동으로 검토하십시오.
- 지속적으로 시스템을 모니터하여 허가되지 않은 작동이 발생하지 않도록 하십시오. 시스템 계정, 암호 및 액세스 패턴을 검토하십시오. 시스템에서 발생하고 있는 사항에 대한 많은 정보를 제공할 수 있습니다.
- 중앙 집중된 `syslog` 저장소를 전개하고 유지 보수하여 `syslog` 메시지를 수집하고 분석하십시오. 이들 로그를 수집하고 검토하여 가치있는 정보를 얻을 수 있습니다.

- 포괄적인 취약점 및 감사 전략을 설정하여 시스템 구성을 모니터하고 유지 보수하십시오. 이 요구사항은 특히 시간에 따라 시스템을 안전한 구성으로 유지 보수한다는 점에서 중요합니다.
- Solaris Security Toolkit 소프트웨어의 최신 버전으로 시스템을 주기적으로 업데이트하십시오.

Solaris Security Toolkit 소프트웨어에는 시작점으로 사용할 수 있는 기본 보안 프로파일이 포함되어 있습니다.

3장

보안 소프트웨어 설치 및 실행

이 장은 Solaris Security Toolkit 소프트웨어 및 기타 보안 관련 소프트웨어 다운로드, 설치 및 실행에 대한 지침을 제공합니다. 독립형 또는 JumpStart 모드에 맞게 환경을 구성하고 지원을 얻기 위한 지시가 포함됩니다.

소프트웨어를 설치, 구성 및 실행하려면 이 절에 제공되는 지침 및 프로세스를 따르십시오. 이들 지시에는 추가 보안 소프트웨어, 도움이 되는 예제 및 안내서 다운로드가 포함됩니다.

Solaris Security Toolkit 소프트웨어가 독립형 제품이지만 다운로드할 수 있도록 제공되는 추가 보안 소프트웨어와 함께 사용될 때 가장 효과적입니다. 이 소프트웨어에는 SunSolve OnLine의 Recommended and Security Patch Cluster, 포함되지 않은 Solaris OS용 Secure Shell 소프트웨어, Solaris OS 및 타사 소프트웨어 사용 권한을 강화하기 위한 사용 권한 및 소유권 수정 소프트웨어 및 Sun 파일 및 실행 파일의 무결성을 검증하기 위한 무결성 검증 이진이 포함됩니다.

이 절에는 다음 작업이 포함됩니다.

계획 및 설치 전 작업 수행

적절한 계획 수립은 Solaris Security Toolkit 소프트웨어를 사용하여 시스템을 성공적으로 보안하기 위한 핵심입니다. 소프트웨어를 설치하기 전의 계획 수립에 대한 자세한 정보는 2 장을 참조하십시오.

전개된 시스템에 소프트웨어를 설치하는 경우, 전개된 시스템에 소프트웨어를 설치하기 전에 설치 전 작업 수행에 대해 28페이지의 "설치 전 작업 수행"을 참조하십시오.

종속성

Solaris Security Toolkit 4.1 소프트웨어는 종속성이 거의 없습니다.

하드웨어 종속성

Solaris 운영 체제의 지원 버전에 대한 자세한 정보는 11페이지의 "지원되는 Solaris OS 버전 실행"을 참조하십시오.

소프트웨어 종속성

Solaris Security Toolkit 4.1 소프트웨어는 SUNWloc 패키지에 의존합니다. 이 패키지가 없을 경우 Solaris Security Toolkit은 작동하지 않습니다.

시스템 관리 서비스(SMS) 소프트웨어의 지원 버전에 대한 자세한 정보는 12페이지의 "지원되는 Solaris SMS 버전 실행"을 참조하십시오.

사용할 모드 판별

시스템이 보안되지 않은 상태에 있는 동안 공격에 노출될 수 있는 기간을 제한하기 위해 설치 중 또는 직후에 시스템을 강화하십시오. Solaris Security Toolkit 소프트웨어를 사용하여 시스템을 보안하기 전에 사용자 환경에서 제대로 실행하도록 Solaris Security Toolkit 소프트웨어를 구성하십시오.

Solaris Security Toolkit 소프트웨어는 모듈형 프레임워크를 갖고 있습니다. JumpStart 제품을 사용하지 않을 경우 Solaris Security Toolkit 소프트웨어 프레임워크의 유연성으로 나중에 JumpStart를 사용하도록 효율적으로 준비할 수 있습니다. JumpStart를 사용하는 경우, 기존 JumpStart 기반구조에 통합하는 Solaris Security Toolkit 소프트웨어의 기능이 유용합니다.

다음 절은 독립형 및 JumpStart 모드에 대해 설명합니다.

독립형 모드

Solaris Security Toolkit 소프트웨어는 독립형 모드에서 Solaris OS 쉘 프롬프트에서 직접 실행합니다. 이 모드로 보안 수정이나 업데이트가 필요하지만 아직 스크래치로부터 OS를 다시 설치하기 위해 서비스를 받을 수 없는 시스템에서 Solaris Security Toolkit 소프트웨어를 사용할 수 있습니다. 그러나 가능하면 시스템을 보안하기 위해 스크래치로부터 시스템을 다시 설치해야 합니다.

독립형 모드는 특히 패치를 설치한 후 시스템을 강화할 때 유용합니다. 오류 없이 한 시스템에 Solaris Security Toolkit 소프트웨어를 여러 번 실행할 수 있습니다. 패치가 Solaris Security Toolkit 소프트웨어가 수정한 파일을 겹쳐 쓰거나 수정할 수 있습니다. Solaris Security Toolkit 소프트웨어를 다시 실행하여 패치 설치에 의해 무효가 된 모든 보안 수정을 다시 구현할 수 있습니다.

주 – 생산 환경에서는 활동 중인 환경에 패치를 설치하기 전에 테스트 및 개발 환경에서 패치를 계획하십시오.

독립형 모드는 전개된 시스템을 가능한 빨리 강화하는 최상의 옵션 중 하나입니다. 36 페이지의 "보안 소프트웨어 다운로드"에 제공된 다운로드 및 설치 지시의 단계 외에 Solaris Security Toolkit 소프트웨어를 JumpStart가 없는 구조에 통합하기 위해 특별한 단계가 필요하지는 않습니다.

JumpStart 모드

Sun의 네트워크 기반 Solaris OS 설치 체계인 JumpStart 기술은 설치 프로세스 중에 Solaris Security Toolkit 스크립트를 실행할 수 있습니다. 이 책에서는 사용자가 JumpStart 기술에 익숙하고 기존 JumpStart 환경을 사용할 수 있다고 가정합니다. JumpStart 기술에 대한 자세한 정보는 Sun BluePrints 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

JumpStart 환경에서 사용하기 위해서는 `JASS_HOME_DIR`(`tar` 다운로드의 경우) 또는 `/opt/SUNWjass`(`pkg` 다운로드의 경우) 중 하나의 Solaris Security Toolkit 소스를 JumpStart 서버의 기본 디렉토리에 복사해야 합니다. 기본값은 JumpStart 서버의 `/jumpstart`입니다. `JASS_HOME_DIR`이 JumpStart 서버의 기본 디렉토리가 됩니다.

Solaris Security Toolkit 소프트웨어를 JumpStart 구조에 통합하기 위해서는 몇 개의 단계만 필요합니다. JumpStart 서버 구성 방법에 대한 지시는 5 장을 참조하십시오.

보안 소프트웨어 다운로드

시스템 강화의 첫 번째 단계에서는 보안하려는 시스템으로 추가 소프트웨어 보안 패키지를 다운로드해야 합니다. 이 절에서는 다음 작업을 다룹니다.

- 36페이지의 "Solaris Security Toolkit 소프트웨어 다운로드"
- 38페이지의 "Recommended Patch Cluster 소프트웨어 다운로드"
- 39페이지의 "FixModes 소프트웨어 다운로드"
- 40페이지의 "OpenSSH 소프트웨어 다운로드"
- 41페이지의 "MD5 소프트웨어 다운로드"

주 – 이 절에서 설명하는 소프트웨어 중에서 Solaris Security Toolkit 소프트웨어, Recommended and Security Patch Cluster, FixModes 및 MD5 소프트웨어는 필수입니다. OpenSSH 대신 다양한 공급 업체에서 구할 수 있는 Secure Shell의 상업용 버전으로 대체할 수 있습니다. 모든 시스템에 Secure Shell 제품을 설치하고 사용하십시오. Solaris 9 OS의 경우는 함께 제공된 Secure Shell 버전을 사용하십시오.

Solaris Security Toolkit 소프트웨어 다운로드

먼저 Solaris Security Toolkit 소프트웨어를 다운로드한 후 독립형 모드에서 Solaris Security Toolkit 소프트웨어를 사용하려는 서버나 JumpStart 모드의 경우 JumpStart 서버에 설치하십시오.

주 – 다음 지시는 버전 번호를 참조하지 않는 파일 이름을 사용합니다. 항상 웹 사이트에서 최신 버전을 다운로드하십시오.

이 안내서 전체에서 `JASS_HOME_DIR` 환경 변수는 Solaris Security Toolkit 소프트웨어의 루트 디렉토리를 의미합니다. Solaris Security Toolkit 소프트웨어가 `tar` 아카이브로부터 설치될 때, `JASS_HOME_DIR`은 `jass-`*n.n*까지(해당 경로 포함)의 경로인 것으로 정의됩니다. `/opt` 디렉토리에 `tar` 버전 배포를 설치하는 경우 `JASS_HOME_DIR` 환경 변수는 `/opt/jass-`*n.n*으로 정의됩니다.

Solaris Security Toolkit 소프트웨어는 전통적인 압축 `tar` 아카이브 외에 Solaris OS 패키지 형식으로 배포됩니다. 두 아카이브 모두에 동일한 소프트웨어가 들어있습니다.

상황에 따라 가장 적합한 형식을 선택하십시오. `pkg` 형식이 클라이언트에게 가장 좋고, `tar`는 JumpStart 시스템 및 사용자 정의 패키지 개발에 가장 좋습니다.

이들 두 개의 아카이브 유형을 다운로드 및 설치하는 절차는 다음 절에서 설명합니다.

▼ tar 버전 다운로드

1. 소프트웨어 배포 파일(jass-*n.n*.tar.Z)을 다운로드합니다.

   소스 파일은 다음 웹 사이트에 있습니다.

   http://www.sun.com/security/jass

2. 표시된 것처럼 zcat 및 tar 명령을 사용하여 서버의 디렉토리에 소프트웨어 배포 파일을 추출합니다.

```
# zcat jass-n.n.tar.Z | tar xvf -
```

   여기서 *n.n*은 다운로드하려는 최신 버전입니다.

   이 명령을 실행하면 현재 작업 디렉토리에 jass-*n.n* 하위 디렉토리가 작성됩니다. 이 하위 디렉토리에는 모든 Solaris Security Toolkit 디렉토리 및 연관된 파일이 들어있습니다.

▼ pkg 버전 다운로드

1. 소프트웨어 배포 파일(SUNWjass-*n.n*.pkg.Z)을 다운로드합니다.

   소스 파일은 다음 위치에 있습니다.

   http://www.sun.com/security/jass

주 – 소프트웨어 다운로드 중에 어려움이 있는 경우 브라우저의 통합된 다른 이름으로 저장 옵션을 사용하십시오.

---

2. uncompress 명령을 사용하여 소프트웨어 배포 파일을 서버의 디렉토리에 추출합니다.

```
# uncompress SUNWjass-n.n.pkg.Z
```

3. 아래에 표시된 것처럼 pkgadd 명령을 사용하여 서버의 디렉토리에 소프트웨어 배포 파일을 설치합니다.

```
# pkgadd -d SUNWjass-n.n.pkg SUNWjass
```

   여기서 *n.n*은 다운로드하려는 최신 버전입니다.

   이 명령을 실행하면 /opt에 SUNWjass 디렉토리가 작성됩니다. 이 하위 디렉토리에는 모든 Solaris Security Toolkit 디렉토리 및 연관된 파일이 들어있습니다.

Recommended Patch Cluster 소프트웨어 다운로드

패치는 Sun에서 성능, 안정성, 기능성 및 보안을 위한 Solaris OS 수정사항을 제공하기 위해 릴리스됩니다. 가장 최신의 패치 클러스터가 설치되는 것이 시스템 보안에 아주 중요합니다. 최신 Solaris OS Recommended and Security Patch Cluster가 시스템에 설치되도록 보장하기 위해 이 절에서 최신 패치 클러스터를 다운로드하는 방법에 대해 설명합니다.

주 – 패치를 설치하기 전에 비생산 시스템에서 또는 예정된 유지보수 기간 중에 패치를 평가하고 테스트하십시오.

▼ Recommended Patch Cluster 소프트웨어 다운로드

패치 클러스터를 설치하기 전에 개별 패치 README 파일 및 기타 제공되는 정보를 검토하십시오. 이 정보에는 종종 패치 클러스터를 설치하기 전에 알아야 하는 제안과 정보가 들어있습니다.

1. 다음 주소의 **SunSolve OnLine** 웹 사이트에서 최신 패치 클러스터를 다운로드합니다.

   `http://sunsolve.sun.com`

2. 왼쪽 탐색 막대의 맨 위에 있는 **Patches** 링크를 누릅니다.

3. **Recommended Patch Clusters** 링크를 누릅니다.

   라이센스 계약 조건이 표시됩니다.

4. **Recommended Solaris Patch Clusters** 상자에서 적합한 **Solaris OS** 버전을 선택합니다.

   이 예제에서는 Solaris 8 OS를 선택하겠습니다.

5. 연결된 라디오 버튼으로 적합한 다운로드 옵션**(HTTP** 또는 **FTP)**을 선택하고 **Go**를 누릅니다.

   브라우저 창에 다른 이름으로 저장 대화 상자가 표시됩니다.

6. 파일을 로컬에 저장합니다.

7. 강화될 시스템으로 해당 파일을 안전하게 이동합니다.

   `scp` (`scp(1)`– 보안 복사 (원격 복사 프로그램)) 명령 또는 보안 파일 전송을 제공하는 다른 방법을 사용합니다.

   다음과 같이 `scp` 명령을 사용하십시오.

```
# scp 8_Recommended.zip target01:
```

8. 파일을 /opt/SUNWjass/Patches 디렉토리로 이동하고 압축을 풉니다.

예를 들면,

코드 예 3-1 패치 파일을 /opt/SUNWjass/Patches 디렉토리로 이동

```
# cd /opt/SUNWjass/Patches
# mv /directory in which file was saved/8_Recommended.zip .
# unzip 8_Recommended.zip
Archive:     8_Recommended.zip
   creating: 8_Recommended/
  inflating: 8_Recommended/CLUSTER_README
  inflating: 8_Recommended/copyright
  inflating: 8_Recommended/install_cluster
[. . .]
```

다른 보안 패키지를 다운로드하고 Solaris Security Toolkit 소프트웨어를 실행하면 패치 클러스터 소프트웨어가 자동으로 설치됩니다.

주 – Recommended and Security Patch Cluster 소프트웨어를 /opt/SUNWjass/Patches 디렉토리에 저장하지 않은 경우, Solaris Security Toolkit 소프트웨어를 실행할 때 경고 메시지가 표시됩니다. OS의 새 릴리스에서 가끔 발생하는 것처럼 적용된 패치 클러스터가 없는 경우 이 메시지를 무시해도 안전합니다.

FixModes 소프트웨어 다운로드

FixModes는 기본 Solaris OS 디렉토리 및 파일 사용 권한을 강화하는 소프트웨어 패키지입니다. 이러한 사용 권한을 강화하면 전체 보안을 크게 개선할 수 있습니다. 사용 권한을 제한할수록 불순한 사용자는 시스템에 대한 권한을 얻기가 더 어려워집습니다.

주 – Solaris 9 OS 릴리스에서는 이전에 FixModes 소프트웨어에 의해 변경된 개체의 기본 사용 권한을 개선하기 위해 변경했습니다. 그러나 타사 소프트웨어 및 번들로 제공되지 않는 소프트웨어에서 파일 및 디렉토리 사용 권한의 강화가 필요하기 때문에 FixModes 소프트웨어가 여전히 필요합니다.

▼ FixModes 소프트웨어 다운로드

1. 다음 위치에서 사전에 이진 코드로 컴파일된 **FixModes** 소프트웨어를 다운로드합니다.

   `http://www.sun.com/security/jass`

   FixModes 소프트웨어는 Solaris OS 시스템용으로 형식화된 사전 컴파일되고 압축된 패키지 버전 파일로서 배포됩니다. 파일 이름은 `SUNBEfixm.pkg.Z`입니다.

2. `scp` 명령이나 안전한 파일 전송을 제공하는 다른 방법을 사용하여 강화될 시스템으로 해당 파일을 안전하게 이동합니다.

   다음과 같이 `scp` 명령을 사용하십시오.

```
# scp SUNBEfixm.pkg.Z target01:
```

3. 다음 명령으로 `/opt/SUNWjass/Packages`의 **Solaris Security Toolkit** `Packages` 디렉토리에 `SUNBEfixm.pkg.Z` 파일을 압축을 풀고 저장합니다.

```
# uncompress SUNBEfixm.pkg.Z
# mv SUNBEfixm.pkg /opt/SUNWjass/Packages/
```

나중에 다른 모든 보안 패키지를 다운로드하고 Solaris Security Toolkit 소프트웨어를 실행하면 FixModes 소프트웨어가 자동으로 설치됩니다.

OpenSSH 소프트웨어 다운로드

보안된 환경에서 사용자 대화식 세션을 보호하려면 철저하게 인증되는 암호를 사용해야 합니다. 최소한 네트워크 액세스를 암호화해야 합니다.

암호화를 구현하기 위해 가장 일반적으로 사용되는 도구는 Secure Shell 소프트웨어로서 Solaris OS와 함께 번들로 제공되는 버전이나 타사의 상업용 버전 또는 프리웨어 버전을 사용합니다. Solaris Security Toolkit 소프트웨어가 수행하는 모든 보안 수정 사항을 적용하려면, Secure Shell 소프트웨어 제품이 있어야 합니다.

주 – Solaris 9 OS를 사용하는 경우, 소프트웨어와 함께 제공되는 Secure Shell 버전을 사용하십시오. 이 버전의 Secure Shell은 Sun의 지원 조직에 의한 지원뿐 아니라 BSM(Basic Security Module) 같은 다른 Solaris OS 보안 기능과 통합합니다.

Secure Shell의 상업용 버전을 얻을 수 있는 위치에 대한 정보는 xx페이지의 "관련 자원"에 제공되어 있습니다.

Solaris Security Toolkit 소프트웨어는 특히 시스템의 모든 암호화되지 않은 사용자 대화식 서비스 및 데몬(특히 `in.telnetd`, `in.ftpd`, `in.rshd` 및 `in.rlogind` 같은 데몬)을 사용 불가능하게 합니다.

Secure Shell을 사용하면 텔넷과 FTP를 사용하는 경우와 마찬가지로 시스템에 액세스할 수 있습니다.

▼ OpenSSH 소프트웨어 다운로드

주 – 서버가 Solaris 9 OS를 실행 중인 경우 번들로 제공되는 Secure Shell 소프트웨어를 사용하고 이 절의 OpenSSH 설치 단계를 생략할 수 있습니다.

- 다음 **Sun BluePrints OnLine** 기사를 읽고, 기사에 나오는 지침을 사용하여 소프트웨어를 다운로드하십시오.

  OpenSSH 컴파일 및 전개 방법에 관한 Sun BluePrints OnLine 기사인 "Building and Deploying OpenSSH on the Solaris Operating Environment"는 다음 위치에서 찾을 수 있습니다.

  `http://www.sun.com/blueprints`

  또는 서점에서 구할 수 있는 Sun BluePrints 참고 서적인 *Secure Shell in the Enterprise*를 읽어보십시오.

  다른 모든 보안 패키지를 다운로드하고 Solaris Security Toolkit 소프트웨어를 실행한 후 OpenSSH 소프트웨어가 자동으로 설치됩니다.

주의 – 강화될 시스템에서 OpenSSH를 컴파일하지 말고 강화될 시스템에 컴파일러를 설치하지 마십시오. 동일한 Solaris OS 버전, 구조 및 모드를 실행하는 별도의 Solaris OS 시스템(예: Solaris 8 OS, Sun4U(`sun4u`) 및 64비트)을 사용하여 OpenSSH를 컴파일하십시오. 상업용 SSH 버전을 구현하는 경우 컴파일이 필요 없습니다. 잠재적인 침입자에 대한 컴파일러 기능을 제한하는 것이 목적입니다. 그러나 시스템에 지역적으로 컴파일러를 설치하지 않는다 해도 여전히 사전에 컴파일된 도구를 설치할 수 있기 때문에 의도적인 공격자에 대해서까지 보호할 수는 없습니다.

MD5 소프트웨어 다운로드

MD5 소프트웨어는 강화되는 시스템에 MD5 디지털 지문을 생성합니다. 디지털 지문을 생성한 후, 이를 Sun이 올바르다고 공개한 것과 비교하여 허가받지 않은 사용자에 의해 변경되거나 악성 코드가 삽입된 (안전해 보이는 것 안에 숨겨진) 시스템 이진을 감지하십시오. 공격자는 시스템 이진을 수정하여 시스템에 대한 백도어 액세스를 확보합니다. 즉, 자신의 존재를 숨기고 시스템을 불안정하게 작동시킬 수 있습니다.

▼ MD5 소프트웨어 다운로드

1. 다음 웹 사이트에서 **MD5** 이진을 다운로드합니다.

   `http://www.sun.com/security/jass`

   MD5 프로그램은 압축 패키지 버전 파일로서 배포됩니다.

2. `scp` 명령이나 안전한 파일 전송을 제공하는 다른 방법을 사용하여 `SUNBEmd5.pkg.Z` 파일을 안전하게 강화될 시스템으로 이동합니다.

   다음과 같이 `scp` 명령을 사용하십시오.

   ```
   # scp SUNBEmd5.pkg.Z target01:
   ```

3. 다음과 비슷한 명령을 사용하여 파일을 `/opt/SUNWjass/Packages`의 **Solaris Security Toolkit** `Packages` 디렉토리에 압축을 풀고 이동합니다.

   ```
   # uncompress SUNBEmd5.pkg.Z
   # mv SUNBEmd5.pkg /opt/SUNWjass/Packages/
   ```

   MD5 소프트웨어가 `/opt/SUNWjass/Packages` 디렉토리에 저장된 후 Solaris Security Toolkit 소프트웨어를 실행하면 해당 소프트웨어가 설치됩니다.

   MD5 이진이 설치된 후 이를 사용하여 Solaris 지문 데이터베이스를 통해 시스템에 있는 실행 파일의 무결성을 검증할 수 있습니다. Solaris 지문 데이터베이스에 대한 자세한 정보는 Sun BluePrints OnLine 기사 "The Solaris Fingerprint Database - A Security Tool for Solaris Software and Files"에 나옵니다.

4. **(**선택적**)** 다음 위치의 **Sun BluePrint** 웹 사이트에서 **Solaris Fingerprint Database Companion** 및 **Solaris Fingerprint Database Sidekick** 소프트웨어를 다운로드하고 설치합니다.

   `http://www.sun.com/blueprints/tools`

   이러한 선택 도구를 MD5 소프트웨어와 함께 설치하고 사용하십시오. 이들 도구는 MD5 체크섬의 데이터베이스에 대해 시스템 이진을 검증하는 프로세스를 간단하게 만듭니다. 이러한 도구를 자주 사용하여 보안된 시스템의 Solaris OS 이진 및 파일의 무결성을 검증하십시오.

   이러한 도구 및 다운로드 지침은 Sun BluePrints OnLine 기사 "The Solaris Fingerprint Database - A Security Tool for Solaris Software and Files"에 나옵니다.

   다운로드한 보안 도구의 무결성을 확인해야 합니다. Solaris Security Toolkit 소프트웨어 및 추가 보안 소프트웨어를 설치하고 실행하기 전에 MD5 체크섬을 사용하여 무결성을 검증하십시오. Solaris Security Toolkit의 다운로드 페이지에서 MD5 체크섬을 구할 수 있습니다.

보안 프로파일 사용자 정의

다양한 보안 프로파일 템플리트가 Solaris Security Toolkit 소프트웨어 배포에 드라이버로서 포함되어 있습니다. 이전 장에서 언급한 것처럼 기본 보안 프로파일 및 이들 드라이버로 작성된 변경이 사용자 시스템에 적합하지 않을 수 있습니다. 이러한 드라이버에 의해 구현되는 보안 프로파일은 필요하지 않은 서비스를 비활성화하고 기본적으로 사용 불가능한 선택적 보안 기능을 활성화합니다.

Solaris Security Toolkit 소프트웨어를 실행하기 전에 기본 보안 프로파일을 검토하고 사용자 환경에 맞게 사용자 정의하거나 새 프로파일을 개발하십시오. 보안 프로파일 사용자 정의에 대한 방법과 지침은 *Solaris Security Toolkit 4.1 Reference Manual*에 있습니다.

소프트웨어 설치 및 실행

Solaris Security Toolkit 소프트웨어를 실행하기 전에 다음 예비 작업을 완료하는 것이 중요합니다. 대부분의 강화는 Solaris Security Toolkit 소프트웨어를 실행할 때 자동으로 완료됩니다.

- 강화하려는 시스템이나 JumpStart 서버에 추가 보안 소프트웨어와 Solaris Security Toolkit 소프트웨어를 다운로드하십시오. 36페이지의 "보안 소프트웨어 다운로드"을 참조하십시오.
- 시스템을 독립형 또는 JumpStart 모드로 구성하십시오. 34페이지의 "사용할 모드 판별"을 참조하십시오.
- 적용 가능한 경우 사용자 환경에 맞게 Solaris Security Toolkit 소프트웨어를 사용자 정의하십시오.
- Solaris Security Toolkit 소프트웨어 및 추가 보안 소프트웨어를 설치하고 실행하기 전에 MD5 체크섬을 사용하여 무결성을 검증하십시오.

명령줄에서 직접 또는 JumpStart 서버에서 Solaris Security Toolkit 소프트웨어를 실행할 수 있습니다.

명령줄 옵션 및 소프트웨어 실행에 대한 기타 정보는 다음 중 하나를 참조하십시오.

- 44페이지의 "독립형 모드에서 소프트웨어 실행"
- 52페이지의 "JumpStart 모드에서 소프트웨어 실행"

독립형 모드에서 소프트웨어 실행

코드 예 3-2는 독립형 모드에서 명령줄 사용법의 예를 보여줍니다.

코드 예 **3-2** 독립형 모드의 명령줄 사용법 예제

```
# ./jass-execute -h

usage:

To apply this Toolkit to a system, using the syntax:
   jass-execute [-r root_directory -p os_version ]
      [ -q | -o output_file ] [ -m e-mail_address ]
      [ -V [3|4] ] [ -d ] driver

To undo a previous application of the Toolkit from a system:
   jass-execute -u [ -b | -f | -k ] [ -q | -o output_file ]
      [ -m e-mail_address ] [ -V [3|4] ]

To audit a system against a pre-defined profile:
   jass-execute -a driver [ -V [0-4] ] [ -q | -o output_file ]
      [ -m e-mail_address ]

To display the history of Toolkit applications on a system:
   jass-execute -H

To display the last application of the Toolkit on a system:
   jass-execute -l

To display this help message:
   jass-execute -h
   jass-execute -?

To display version information for this program:
   jass-execute -v

#
```

표 3-1은 사용 가능한 명령줄 옵션을 나열하고 각 옵션에 대해 설명합니다.

표 **3-1** `jass-execute`와 함께 사용하는 명령줄 옵션

옵션	설명
`-a`	시스템이 보안 프로파일을 준수하는지 판별합니다.
`-b`	-u 옵션과 함께 사용됩니다. 최종 강화 작업 이후에 수동으로 변경된 모든 파일을 백업한 후 시스템을 원래 상태로 복원합니다.
`-d`	독립형 모드에서 실행될 드라이버를 지정합니다.

표 3-1 jass-execute와 함께 사용하는 명령줄 옵션 (계속)

옵션	설명
-f	-u 옵션과 함께 사용됩니다. 파일이 강화 작업 후에 수동으로 변경된 경우에도 사용자에게 예외에 대해 알리지 않고 강화 작업 중에 작성된 변경을 취소합니다.
-h	사용 가능한 옵션의 개요를 제공하는 jass-execute 도움말 메시지를 표시합니다.
-H	시스템에서 Solaris Security Toolkit 소프트웨어의 내역을 표시합니다.
-k	-u 옵션과 함께 사용됩니다. 최종 강화 작업 이후에 수동으로 변경한 내용을 보존합니다.
-l	시스템에서 Solaris Security Toolkit의 최종 작업을 표시합니다.
-m	출력을 지정된 전자 우편 주소로 보냅니다.
-o	출력을 지정된 파일로 보냅니다.
-p	-r *root_directory* 옵션과 함께 사용됩니다. Solaris 운영 체제의 OS 버전을 지정합니다. 형식은 uname -r과 동일합니다.
-q	화면에 대한 출력 표시를 금지합니다. 또한 정숙 옵션이라고도 합니다.
-r	-p *os_version*과 함께 사용되어야 합니다. jass-execute 실행 중에 사용되는 루트 디렉토리를 지정합니다. 기본적으로 루트 파일 시스템은 /입니다. 이 루트 디렉토리는 Solaris Security Toolkit (JASS) 환경 변수 JASS_ROOT_DIR에 의해 정의됩니다. 보안되는 Solaris OS는 /를 통해 사용 가능합니다. 예를 들어 /mnt에 임시로 마운트되는 별도의 OS 디렉토리를 보안하려는 경우 -r 옵션을 사용하여 /mnt를 지정합니다.
-u	예외가 발생했을 때 사용자가 취하고자 하는 활동을 묻는 대화식 프롬프트와 함께 실행 취소 옵션을 실행합니다. -d, -a, -h, -l 또는 -H 옵션과 함께 사용할 수 없습니다.
-v	이 프로그램의 버전 정보를 표시합니다.
-V	메시지 출력의 세부사항 레벨을 지정합니다.
-?	사용 가능한 옵션의 개요를 제공하는 jass-execute 도움말 메시지를 표시합니다.

독립형 모드에서 jass-execute 명령과 함께 사용 가능한 옵션에 대한 자세한 정보는 다음 절을 참조하십시오.

- 47페이지의 "감사 옵션"
- 47페이지의 "도움말 표시 옵션"
- 48페이지의 "드라이버 옵션"
- 49페이지의 "전자 우편 통지 옵션"
- 49페이지의 "실행 내역 옵션"
- 50페이지의 "가장 최근 실행 옵션"
- 50페이지의 "출력 파일 옵션"
- 51페이지의 "정숙 출력 옵션"
- 51페이지의 "루트 디렉토리 옵션"
- 52페이지의 "실행 취소 옵션"

사용 가능한 드라이버의 전체 목록은 Drivers 디렉토리를 참조하십시오. 소프트웨어의 최신 버전에 추가 드라이버가 있을 수 있습니다.

▼ 독립형 모드에서 소프트웨어 실행

1. 다음과 같이 `secure.driver`(또는 `sunfire_15k_sc-secure.driver` 같은 제품 특정 스크립트)를 실행합니다.

코드 예 **3-3** 독립형 모드에서 소프트웨어 실행

```
# cd /opt/SUNWjass
# ./jass-execute -d secure.driver

[NOTE] The following prompt can be disabled by setting
JASS_NOVICE_USER to 0.
[WARN] Depending on how the Solaris Security Toolkit is configured,
it is both possible and likely that by default all remote shell
and file transfer access to this system will be disabled upon
reboot effectively locking out any user without console access to
the system.

Are you sure that you want to continue? (YES/NO) [NO]
y

[NOTE] Executing driver, secure.driver

================================================================
secure.driver: Driver started.
================================================================

================================================================
Solaris Security Toolkit Version: 4.1.0
Node name:                        ufudu
Host ID:                          8085816e
Host address:                     10.8.31.115
MAC address:                      8:0:20:85:81:6e
OS version:                       5.9
Date:                             Tue May 4 16:28:24 EST 2004
================================================================
[...]
```

사용 가능한 드라이버의 전체 목록은 Drivers 디렉토리를 참조하십시오. 소프트웨어의 최신 버전에 추가 드라이버가 있을 수 있습니다.

2. 시스템에서 **Solaris Security Toolkit** 소프트웨어를 실행한 후, 시스템을 재부팅하면 변경사항이 적용됩니다.

   강화 작업 중에 클라이언트의 구성이 다양하게 수정됩니다. 이러한 수정에는 서비스용 시작 스크립트 사용 불가, 서비스용 옵션 사용 불가 및 패치를 통한 새 이진 또는 라이브러리 설치가 포함될 수 있습니다. 클라이언트가 다시 시작될 때까지 이들 수정은 효력을 갖지 않습니다.

3. 시스템을 재부트한 후 수정의 정확성과 완전성을 검증합니다.

   53페이지의 "시스템 수정 검증"을 참조하십시오.

4. 오류가 발생하는 경우 오류를 수정하고 독립형 모드에서 **Solaris Security Toolkit** 소프트웨어를 다시 실행합니다.

감사 옵션

Solaris Security Toolkit 소프트웨어는 `-a` 옵션을 통해 감사 실행을 수행하여 시스템이 보안 프로파일을 준수하는지 판별할 수 있습니다. 이 실행은 작성된 시스템 파일 수정이 여전히 활동하는지 여부뿐 아니라 이전에 사용 불가능하게 된 프로세스가 실행 중인지 아니면 제거된 소프트웨어 패키지가 다시 설치되었는지를 검증합니다. 이 기능에 대한 자세한 정보는 6 장을 참조하십시오.

보안 프로파일에 대해 시스템을 감사하는 사용 예제:

```
# jass-execute -a driver [ -V [0-4] ] [ -q | -o output-file ]
[ -m email-address ]
```

도움말 표시 옵션

`-h` 옵션은 사용 가능한 옵션의 개요를 제공하는 `jass-execute` 도움말 메시지를 표시합니다.

`-h` 옵션은 다음과 비슷한 출력을 작성합니다.

코드 예 **3-4** 예제 `-h` 옵션 출력

```
# ./jass-execute -h
To apply this Toolkit to a system, using the syntax:
   jass-execute [-r root_directory -p os_version ]
      [ -q | -o output_file ] [ -m e-mail_address ]
      [ -V [3|4] ] [ -d ] driver

To undo a previous application of the Toolkit from a system:
   jass-execute -u [ -b | -f | -k ] [ -q | -o output_file ]
      [ -m e-mail_address ] [ -V [3|4] ]

To audit a system against a pre-defined profile:
   jass-execute -a driver [ -V [0-4] ] [ -q | -o output_file ]
      [ -m e-mail_address ]

To display the history of Toolkit applications on a system:
   jass-execute -H
```

코드 예 **3-4** 예제 -h 옵션 출력 (계속)

```
To display the last application of the Toolkit on a system:
   jass-execute -l

To display this help message:
   jass-execute -h
   jass-execute -?

To display version information for this program:
   jass-execute -v

Note that just the driver name should be specified when using the
'-d' or '-a' options. A path need not be specified as the script
is assumed to exist in the Drivers directory.

The '-u' undo option is mutually exclusive with the '-d' and '-a'
options. The default undo behavior is to ask the user what to do if
a file to be restored has been modified as the checksum is
incorrect.

The -u option can be combined with the '-k', '-b', or '-f' to
override the default interactive behavior. The use of one of these
options is required when run in quiet mode ('-q').

The '-k' option can be used to always keep the current file and
backup if checksum is incorrect. The 'b' can be used to backup the
current file and restore original if the checksum is incorrect.
The 'f' option will always overwrite the original if the checksum
is incorrect, without saving the modified original.
```

드라이버 옵션

-d 드라이버 옵션은 독립형 모드에서 실행될 드라이버를 지정합니다.

-d 옵션과 함께 드라이버를 지정해야 합니다. Solaris Security Toolkit 소프트웨어는 추가된 스크립트의 이름 앞에 Drivers/를 붙입니다. 명령줄에는 스크립트 이름만 입력하면 됩니다.

주 – -u, -H, -h 또는 -a 옵션과 함께 -d 옵션을 사용할 수 없습니다.

-d 드라이버 옵션을 사용한 jass-execute 강화 작업은 다음과 비슷한 출력을 생성합니다.

코드 예 **3-5** 예제 -d 드라이버 옵션 출력

```
# ./jass-execute -d secure.driver
[...]
[NOTE] Executing driver, secure.driver

================================================================
secure.driver: Driver started.
================================================================

================================================================
Solaris Security Toolkit Version: 4.1.0
Node name:                        ufudu
Host ID:                          8085816e
Host address:                     10.8.31.115
MAC address:                      8:0:20:85:81:6e
OS version:                       5.9
Date:                             Tue Oct 4 16:28:24 EST 2004
================================================================
[...]
```

전자 우편 통지 옵션

-m 전자 우편 주소 옵션은 실행이 완료될 때 독립형 강화 및 실행 취소 출력이 Solaris Security Toolkit 소프트웨어에 의해 자동으로 전자 우편을 통해 보내질 수 있는 체계를 제공합니다. 전자 우편 보고서는 다른 옵션을 사용하여 시스템에 생성되는 모든 로그에 추가로 작성됩니다.

전자 우편 옵션을 사용하여 sunfire_15k_sc-config.driver를 호출하는 Solaris Security Toolkit 실행은 다음과 비슷합니다.

```
# ./jass-execute -m root -d sunfire_15k_sc-config.driver
[...]
```

실행 내역 옵션

-H 옵션은 Solaris Security Toolkit 소프트웨어가 한 시스템에서 실행된 횟수를 판별하는 간단한 체계를 제공합니다. 실행 취소되었는지 여부와 상관없이 모든 실행이 나열됩니다.

-H 옵션은 다음과 비슷한 출력을 작성합니다.

코드 예 **3-6** 예제 -H 옵션 출력

```
# ./jass-execute -H
Note: This information is only applicable for applications of
      the Solaris Security Toolkit starting with version 4.1.0.

The following is a listing of the applications of the Solaris
Security Toolkit on this system.  This list is provided in
reverse chronological order:

1.   June 31, 2004 at 12:20:19 (20040631122019) (UNDONE)
2.   June 31, 2004 at 12:10:29 (20040631121029)
3.   June 31, 2004 at 12:04:15 (20040631120415)
```

이 출력은 Solaris Security Toolkit 소프트웨어가 이 시스템에서 세 번 실행되었고 마지막 실행이 취소되었음을 나타냅니다.

가장 최근 실행 옵션

-l 옵션은 가장 최근 실행을 판별하는 체계를 제공합니다. 이것은 또한 항상 -H 옵션에 의해 나열되는 마지막 실행입니다.

-l 옵션은 다음과 비슷한 출력을 제공합니다.

코드 예 **3-7** 예제 -l 옵션 출력

```
# ./jass-execute -l

Note: This information is only applicable for applications of
      the Solaris Security Toolkit starting with version 4.1.0.

The last application of the Solaris Security Toolkit was:

1.   June 31, 2004 at 12:20:19 (20040631122019) (UNDONE)
```

출력 파일 옵션

-o 출력 파일 옵션은 jass-execute 실행의 콘솔 출력을 별개 파일인 출력 파일로 경로 재지정합니다.

이 옵션은 JASS_REPOSITORY 디렉토리에 보존되는 로그에는 아무 영향도 주지 않습니다. 이 옵션은 Solaris Security Toolkit 실행에 의해 생성되는 상당한 양의 출력이 있기 때문에 특히 느린 터미널 연결을 통해 수행될 때 유용합니다.

이 옵션은 -d, -u 또는 -a 옵션 중 하나와 함께 사용할 수 있습니다.

-o 옵션은 다음과 비슷한 출력을 표시합니다.

코드 예 **3-8** 예제 -o 옵션 출력

```
# ./jass-execute -o jass-output.txt -d secure.driver
[NOTE] Executing driver, secure.driver
[NOTE] Recording output to jass-output.txt
```

정숙 출력 옵션

-q 옵션은 강화 작업 중에 표준 입출력(stdio) 스트림에 대한 Solaris Security Toolkit 출력을 비활성화 합니다.

이 옵션은 JASS_REPOSITORY 디렉토리에 보존되는 로그에는 아무 영향도 주지 않습니다. -o 옵션과 비슷하게, 이 옵션은 특히 cron 작업을 통하거나 느린 네트워크 연결을 통해 Solaris Security Toolkit 소프트웨어를 실행할 때 도움이 됩니다.

이 옵션은 -d, -u 또는 -a 옵션 중 하나와 함께 사용할 수 있습니다.

-q 옵션은 다음과 비슷한 출력을 작성합니다.

코드 예 **3-9** 예제 -q 옵션 출력

```
# ./jass-execute -q -d secure.driver
[NOTE] Executing driver, secure.driver
```

루트 디렉토리 옵션

-r 루트 디렉토리 옵션은 jass-execute 실행 중에 사용되는 루트 디렉토리를 지정하기 위한 것입니다. -r 옵션 사용은 또한 -p 옵션을 사용하여 플랫폼(OS) 버전을 지정해야 합니다. -p 옵션의 형식은 uname -r에 의해 작성되는 형식과 동등합니다.

기본적으로 루트 파일시스템 디렉토리는 /입니다. 이 루트 디렉토리는 Solaris Security Toolkit 환경 변수 JASS_ROOT_DIR에 의해 정의됩니다. 보안되는 Solaris OS는 /를 통해 사용 가능합니다. 예를 들어 /mnt에 임시로 마운트되는 별도의 OS 디렉토리를 보안하려는 경우 -r 옵션을 사용하여 /mnt를 지정합니다. 모든 스크립트가 해당 OS 이미지에 적용됩니다.

실행 취소 옵션

`-u` 옵션을 통해 Solaris Security Toolkit 소프트웨어가 강화 중에 수행되는 시스템 수정 사항을 실행 취소할 수 있습니다. 각 종료 스크립트는 `-u` 옵션으로 실행 취소할 수 있습니다. 또한 Solaris Security Toolkit의 실행 취소 능력은 각 실행 중에 생성되는 체크섬과 밀접하게 통합됩니다. 이 기능에 대한 자세한 정보는 4 장을 참조하십시오.

실행 취소 명령의 명령줄 사용 예제:

```
# jass-execute -u [ -b | -f | -k ] [ -q | -o output_file ]
        [ -m e-mail_address ] [ -V [3|4] ]
```

JumpStart 모드에서 소프트웨어 실행

JumpStart 모드는 JumpStart 서버의 `rules` 파일에 삽입된 Solaris Security Toolkit 드라이버에 의해 제어됩니다.

JumpStart 모드를 사용하도록 환경을 구성하지 않은 경우 5 장을 참조하십시오.

JumpStart 기술에 대한 자세한 정보는 Sun BluePrints 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

▼ JumpStart 모드에서 소프트웨어 실행

JumpStart 모드에서 Solaris Security Toolkit 소프트웨어를 실행하려면 JumpStart 환경에 통합되고 JumpStart 설치와 연관된 종료 스크립트의 일부로서 호출되어야 합니다. Solaris Security Toolkit 소프트웨어를 사용자 환경에 통합하는 방법에 대해서는 5 장을 참조하십시오.

1. 드라이버에 모든 필수 수정 사항을 적용한 후, **JumpStart** 기반 구조를 사용하여 클라이언트를 설치합니다.

   이 작업은 클라이언트의 `ok` 프롬프트에서 다음 명령을 사용하여 수행됩니다.

   ```
   ok> boot net - install
   ```

   설치가 완료된 후 시스템이 JumpStart 소프트웨어에 의해 재부트됩니다.

   시스템이 올바른 구성 상태에 있어야 합니다. 강화 작업 중에 클라이언트의 구성이 다양하게 수정됩니다. 이러한 수정에는 서비스용 시작 스크립트 사용 불가, 서비스용 옵션 사용 불가 및 패치를 통한 새 이진 또는 라이브러리 설치가 포함될 수 있습니다. 클라이언트가 다시 시작될 때까지 이들 수정은 효력을 갖지 않습니다.

2. 시스템이 재부트된 후 수정 사항이 제대로 올바르게 적용되었는지 검증합니다.
   53페이지의 "시스템 수정 검증"을 참조하십시오.
3. 오류가 발생할 경우, 오류를 수정하고 클라이언트 **OS**를 재설치합니다.

시스템 수정 검증

시스템을 재부트한 후, 다음 절에서 설명하는 것처럼 수정사항의 정확성과 완전성을 검증하십시오.

서비스에 대한 QA 점검 수행

시스템 보안에 관련되는 중대한 도전 중 하나는 시스템이 제대로 기능하기 위해 계속 사용 가능해야 하는 OS 서비스를 판별하는 것입니다. Solaris OS 서비스는 시스템에 로그온하기 위해 직접 사용되기 때문에(예: Secure Shell) 필요할 수 있습니다. 또는 타사 소프트웨어 관리 도구의 그래픽 사용자 인터페이스용으로 RPC(Remote Procedure Call) 데몬을 사용하는 것처럼 간접적으로 사용할 수 있습니다.

이들 요구사항의 대부분은 Solaris Security Toolkit 소프트웨어를 실행하기 전에 판별되어야 합니다. (18페이지의 "응용 프로그램 및 서비스 요구사항 판별"을 참조하십시오.) 그러나 유일하게 결정적인 체계는 시스템을 설치 및 보안한 후 품질 보장(QA) 테스트를 통한 필수 기능성의 철저한 테스트를 수행하는 것입니다. 시스템이 강화된 후 전개되는 모든 새로운 시스템에 대해 QA 계획이 적절하게 실행되어야 합니다. 비슷하게 강화될 전개된 시스템의 경우 모든 필수 및 예상 기능이 존재하는지 확인하기 위해 철저한 테스트를 수행해야 합니다.

QA 프로세스가 어떤 모순도 적발하지 않는 경우 다음을 수행하십시오.

1. 2 장의 지침을 기초로 문제 영역을 판별합니다.
2. 응용프로그램이 수정된 구성에서 실행하는지 검증합니다.
3. Solaris Security Toolkit의 실행을 취소합니다.
4. 문제점 해결에 근거하여 보안 프로파일(드라이버)을 수정합니다.
5. Solaris Security Toolkit 소프트웨어를 다시 실행합니다.

종료 결과가 어떤 필수 기능에도 부정적으로 영향을 주지 않고 시스템에서 실행될 수 있는 보안 프로파일이어야 합니다.

구성의 보안 평가 수행

시스템이 모든 필수 기능을 수행하는지 검증하는 동안 보안 구성을 평가하여 시스템이 원하는 레벨로 보안되는지 판별하십시오. 시스템에 수행된 강화 또는 최소화에 따라서 이것이 다른 측면과 관련될 수 있습니다.

최소한 다음 방법으로 시스템의 구성을 검토해야 합니다.

- 적합한 모든 Security and Recommended Patches가 설치되었는지 확인합니다.
- 필수 및 적합한 프로세스만이 실행 중이고 이들이 적당한 인수와 함께 실행 중인지 확인합니다.
- 필수 데몬만이 실행 중이고 이들이 적당한 인수와 함께 실행 중인지 확인합니다.
- 지역적으로(예를 들어 `netstat -a`) 및 네트워크 인터페이스에서 사용 가능한 포트를 판별할 수 있는 Nmap 같은 포트 스캐너를 사용하여 원격으로 점검하여 시스템에 필수 포트만이 열려있는지 확인합니다.
- 시스템이 최소화된 경우 필수 Solaris OS 패키지만 설치되었는지 확인합니다.

이 검토가 새로 구축되고 보안된 시스템에 대한 최소로 고려되어야 합니다. 레거시 시스템을 강화할 때 허가되지 않은 수정이 있었는지 판별하기 위해 기초 OS를 검증해야 합니다. 이런 종류의 무결성 점검은 시스템의 파일 시스템을 읽기 전용 모드로 마운트하고 알려진 OS 인스턴스로부터 무결성 점검 소프트웨어를 실행하여 가장 잘 수행됩니다. Sun BluePrints OnLine 기사 "The Solaris Fingerprint Database-A Security Tool for Solaris Software and Files"에 설명되어 있는 도구가 이러한 작업에 유용하게 사용될 수 있습니다.

보안 프로파일 검증

시스템이 보안되고 사용자가 시스템의 필수 서비스 및 능력을 검증한 후 감사 기능을 사용하여 보안 프로파일이 제대로 완벽하게 적용되었는지 확인하십시오. 이 작업은 두 가지 이유 때문에 중요합니다. 첫 번째는 시스템이 필요한 대로 강화되는지 확인하는 것입니다. 두 번째는 시스템에 대해 정의된 보안 프로파일이 Solaris Security Toolkit 구성에 적절히 반영되도록 보장하는 것입니다. 이 점검은 구성 정보가 시스템의 전체 전개 수명 동안 시스템의 보안 프로파일을 유지 보수하는 데 사용되기 때문에 중요합니다.

감사 기능에 대한 자세한 정보는 6 장을 참조하십시오.

설치 후 작업 수행

전개된 시스템에 소프트웨어를 설치한 경우, 전개된 시스템에 설치 후 작업 수행에 대한 정보는 29페이지의 "설치 후 작업 수행"을 참조하십시오.

4장

시스템 변경 취소

이 장은 강화 작업 중에 Solaris Security Toolkit 소프트웨어를 통해 수행된 변경사항의 역전(실행 취소)에 대한 정보 및 절차를 제공합니다. 이 옵션은 시스템을 Solaris Security Toolkit 강화 작업 또는 일련의 작업 이전의 상태로 되돌릴 수 있는 자동화 체계를 제공합니다.

이 장에서는 다음 주제를 다룹니다.

변경사항이 로깅되고 취소되는 방법 이해

각 Solaris Security Toolkit 강화 작업은 JASS_REPOSITORY에 작업 디렉토리를 작성합니다. 이들 디렉토리의 이름은 작업이 시작되는 날짜 및 시간을 토대로 합니다. 출력을 화면에 표시하는 것 외에 Solaris Security Toolkit 소프트웨어는 변경을 추적하고 조작을 로깅하는 파일 세트를 디렉토리에 작성합니다.

이 디렉토리에 저장되는 파일이 시스템에 수행된 수정사항을 추적하고 실행 취소 기능이 작동할 수 있게 합니다.

주의 – JASS_REPOSITORY에 있는 파일의 내용은 절대 관리자에 의해 수정되지 않아야 합니다.

JumpStart 모드나 독립형 모드에서 Solaris Security Toolkit 소프트웨어를 사용하여 시스템을 강화할 때 소프트웨어는 JASS_REPOSITORY/jass-manifest.txt 파일에 변경을 로깅합니다. 이 파일은 실행 취소 기능이 변경을 취소하는 데 사용하는 조작을

나열합니다. 이 파일에는 작성, 복사, 이동 또는 제거된 파일을 포함하여 Solaris Security Toolkit 소프트웨어에 의해 구현되는 강화 조작에 대한 정보가 들어있습니다. 또한 이 파일은 소프트웨어 패키지 설치 같이 더 복잡한 변경을 취소할 때 필요한 표준 및 사용자 정의 항목을 포함할 수 있습니다. 각 강화 작업에 대해 별도의 `jass-manifest.txt` 파일이 작성됩니다.

주 – Solaris Security Toolkit 소프트웨어 실행 취소 기능은 목록 파일에 항목이 있는 변경만 취소합니다.

실행 취소 작업은 Solaris Security Toolkit 작업 중에 생성되고 `JASS_REPOSITORY`에 저장되는 목록 파일을 조사합니다. 이 작업은 백업된 파일을 원래 위치로 복원합니다. 파일이 백업되지 않은 경우 실행 취소 기능을 사용할 수 없습니다.

Solaris Security Toolkit 작업의 실행 취소시 연결된 디렉토리는 제거되지 않습니다. 대신 `JASS_REPOSITORY` 디렉토리에 다음 두 파일 `jass-undo-log.txt` 및 `reverse-jass-manifest.txt`가 작성됩니다. 그 후, 다음에 `jass-execute -u`가 실행될 때 실행 취소된 작업이 나열되지 않습니다. 강화 작업은 한 번만 실행 취소할 수 있습니다.

시스템 변경 실행 취소 요구사항

Solaris Security Toolkit 소프트웨어의 실행 취소 기능 사용에 대한 다음 한계 및 요구사항을 알고 있어야 합니다.

- Solaris Security Toolkit 버전 0.3에서 4.1까지 독립형 또는 JumpStart 모드에서 시작된 실행에 대해 실행 취소 기능을 사용할 수 있습니다. 그러나 독립형 모드에서만 변경을 실행 취소할 수 있습니다. JumpStart 설치 중에는 실행 취소 기능을 사용할 수 없습니다.
- JumpStart 또는 독립형 모드를 통해 백업 파일을 작성하지 않는 Solaris Security Toolkit 옵션을 선택하는 경우 실행 취소 기능을 사용할 수 없습니다. 백업 파일 사본의 작성은 `JASS_SAVE_BACKUP` 매개변수를 `0`으로 설정하여 비활성화 됩니다.
- 작업은 한 번만 실행 취소할 수 있습니다.
- Solaris Security Toolkit 프레임워크 기능을 사용하지 않는 새로운 종료 스크립트를 개발하는 경우 대응하는 감사 스크립트를 작성하고 `add_to_manifest` 기능을 사용하여 목록 파일에 항목을 추가해야 합니다. 그렇지 않으면 Solaris Security Toolkit이 사용자 정의 개발에 대해 알 수 있는 방법이 없습니다.
- 어떤 상황에서도 `JASS_REPOSITORY` 디렉토리의 내용을 수정하지 마십시오. 파일을 수정하면 내용이 손상되고 실행 취소 기능을 사용할 때 예기치 않은 오류 또는 시스템 손상이 유발될 수 있습니다.

변경을 실행 취소하도록 스크립트 사용자 정의

Solaris Security Toolkit 프레임워크는 종료 스크립트 디자인 및 구축에 대한 유연성을 제공합니다. 이 프레임워크는 Solaris Security Toolkit 소프트웨어의 기능을 확장하여 시스템의 수명 주기 동안 시스템 구성을 잘 관리하도록 도와주며 조직의 필요성에 부응하도록 해줍니다.

스크립트를 사용자 정의할 때 사용자가 취하는 조치가 실행 취소 기능에 영향을 줄 수 있는 방법을 이해하는 것이 중요합니다. 스크립트 사용자 정의를 단순화하기 위해 지원 프로그램 기능이 목록 파일을 적합하게 변경합니다. (실행 취소 기능은 강화 작업을 취소하기 위해 목록 파일의 내용에 의존합니다.) 대부분의 경우 이들 지원 프로그램 기능이 조직에 맞게 스크립트를 사용자 정의하기 위해 필요한 것을 제공합니다.

지원 프로그램 기능의 목록과 해당 기능의 사용에 대한 정보는 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오. 실행 취소 작업이 목록 파일의 관련 항목을 참조할 수 있도록 시스템 명령 대조물 대신 이들 지원 프로그램 기능을 사용하십시오.

어떤 경우에는 지원 프로그램 기능이 없는 기능을 수행해야 할 수 있습니다. 이러한 경우에는 `add_to_manifest`라는 특수 기능을 사용하십시오. 이 기능을 사용하여 지원 프로그램 기능 중 하나를 호출할 필요 없이 수동으로 목록 파일에 항목을 삽입할 수 있습니다. 이 특수 기능을 주의해서 사용하여 시스템 및 Solaris Security Toolkit 저장소의 무결성을 보호하십시오. 이 특수 기능을 사용할 수 있는 시기의 예는 Sun의 `pkg` 형식이 아닌 소프트웨어 패키지를 추가할 때입니다. 이 예제에서는 실행 취소 기능에 강화 작업 중에 다른 형식으로 추가된 패키지를 제거하는 방법을 지시해야 합니다.

지원 프로그램 기능과 특수 `add_to_manifest` 기능을 사용하여 Solaris Security Toolkit 소프트웨어는 스크립트를 사용자 정의하고 변경사항이 실행 취소 작업으로 확장되게 하는 간단하고 유연성 있는 방법을 제공합니다.

이러한 기능을 사용하지 않고 종료 스크립트의 작동을 변경할 경우, Solaris Security Toolkit 소프트웨어는 변경사항이 발생했는지 알 수 없습니다. 그러므로 목록 파일에 참조되지 않은 모든 변경사항을 수동으로 실행 취소해야 합니다.

다른 예제에서는, 시스템의 파일을 수정하기 전에 파일의 원래 버전을 먼저 저장해야 합니다. Solaris Security Toolkit 소프트웨어의 컨텍스트 밖에서, 일반적으로 사용자는 `/usr/bin/cp` 명령을 사용하여 이 작업을 수행합니다. 그러나 Solaris Security Toolkit 소프트웨어의 컨텍스트 내에서 이 명령을 직접 사용할 경우, Solaris Security Toolkit 소프트웨어는 목록 항목의 작성 여부를 알 수 없습니다. `cp` 명령 대신 `backup_file` 지원 프로그램 기능을 사용하십시오. 이 기능은 `JASS_SUFFIX`의 접미어와 함께 원본 파일의 사본을 저장하고 Solaris Security Toolkit 소프트웨어에 파일의 사본이 작성되었음을 알리는 목록 항목을 추가합니다. 또한 이 기능으로 인해 파일 체크섬이 계산됩니다. 파일 체크섬은 `jass-check-sum` 명령뿐 아니라 실행 취소 기능 모두에 의해 사용됩니다.

수동으로 변경된 파일 점검

jass-execute -u 명령을 사용하는 것이 강화 작업 후에 수동으로 변경된 파일을 자동으로 점검하지만, 때로는 jass-check-sum 명령을 사용하여 변경된 파일을 나열하고 검토하는 것이 도움이 되는 것을 알 수 있습니다.

이 명령을 사용하면 JASS_REPOSITORY의 내용을 검토하고 목록 파일에 나열되는 모든 파일에 대해 체크섬을 수행하여 나열된 파일이 체크섬이 강화 작업 중에 기록된 이후에 변경되었는지 판별할 수 있습니다. 강제 실행 취소 작업을 계속하기 전에 이 점검을 수행하는 것이 많은 불필요한 문제 해결 시간을 저장할 수 있는 가치 있는 정보를 제공합니다.

다음은 예제 출력입니다.

코드 예 **4-1** 수동으로 변경된 파일의 예제 출력

```
# ./jass-check-sum
File Name              Saved CkSum             Current CkSum
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
/etc/inet/inetd.con    1643619259:6883         2801102257:6879
/etc/logadm.conf       2362963540:1042         640364414:1071
/etc/default/inetd     3677377803:719          2078997873:720
```

이 출력은 강화 작업이 완료된 후에 3개의 파일이 변경되었음을 나타냅니다.

실행 취소 기능에서 옵션 사용

이 절에서는 실행 취소 작업을 실행할 때 사용할 수 있는 jass-execute -u 명령 및 옵션에 대해 설명합니다.

주 – 실행 취소 기능과 함께 -d, -a, -h, -l 또는 -H 옵션을 사용할 수 없습니다. 대기 모드에서 undo 명령 실행시, -b, -k 또는 -f 옵션을 제공해야 합니다.

---

jass-execute -u 명령은 실행 취소 작업 실행을 위한 표준입니다. 이 명령은 마지막 강화 작업 이후 수동으로 수정된 모든 파일을 자동으로 발견합니다. Solaris Security Toolkit 소프트웨어가 강화 실행 후 수동으로 변경된 파일을 발견할 경우, 다음 응답중 하나를 선택하도록 요청합니다.

1. 원본(강화 작업 전에 존재했던 파일)을 복원하기 전에 최신 파일을 백업합니다.
2. 최신 파일을 보존하고 원본 파일을 복원하지 않습니다.

3. 수동으로 변경된 모든 파일을 강제로 겹쳐 쓰고(데이터를 유실할 수 있음) 원본 파일을 복원합니다.

기본 실행 취소 작동을 변경하려는 경우, 실행 취소 명령을 실행할 때 `-b`, `-k` 및 `-f` 옵션을 사용하십시오.

표 4-1은 실행 취소와 함께 사용할 수 있는 명령줄 옵션을 나열합니다. 각 옵션에 대한 자세한 정보는 다음 절을 참조하십시오.

표 **4-1** 실행 취소 명령과 함께 명령줄 옵션 사용

옵션	설명
`-b`	최종 강화 작업 이후에 수동으로 변경된 모든 파일을 백업한 후 시스템을 원래 상태로 복원합니다.
`-f`	파일이 강화 작업 후에 수동으로 변경된 경우에도 사용자에게 예외에 대해 알리지 않고 강화 작업 중에 작성된 변경을 취소합니다.
`-k`	강화 작업 후에 파일에 작성한 모든 수동 변경을 보존합니다.
`-m`	출력을 전자 우편 주소로 보냅니다.
`-o`	출력을 파일로 보냅니다.
`-q`	화면에 대한 출력 표시를 금지합니다. 또한 정숙 옵션이라고도 합니다. 출력이 `JASS_REPOSITORY/jass-undo-log.txt`에 저장됩니다.

백업 옵션

`-b` 옵션은 최종 강화 작업 이후에 수동으로 변경된 모든 파일을 자동으로 백업한 후 파일을 강화 작업 전의 원래 상태로 복원합니다. 수동 변경을 구현하기 위해 복원된 파일을 백업된 파일과 비교하고 그 차이를 수동으로 조정해야 합니다. 파일이 이 옵션을 사용하여 백업되는 경우 다음 예제와 비슷하게 나타납니다.

```
/etc/motd.BACKUP.JASS_SUFFIX
```

강제 옵션

`-f` 옵션은 파일이 강화 작업 후에 수동으로 변경된 경우에도 예외 없이 강화 작업 중에 작성된 변경사항을 취소합니다. 실행 취소 작업은 저장된 파일 체크섬을 파일의 현재 버전에 비교하지 않습니다. 결국 강화 작업 후에 파일을 수동으로 변경한 경우 실행 취소 작업 후에 변경사항이 겹쳐 써지고 유실됩니다.

실행 취소 작업이 완료된 후 변경사항을 수동으로 다시 구현해야 할 수 있습니다. 게다가 작성된 변경의 유형에 따라서 파일의 그룹 사이의 차이를 조정해야 할 수도 있습니다. 이런 문제점을 막으려면 `jass-check-sum` 명령이나 이전에 언급된 `-b` 명령줄 옵션을 사용하십시오.

보존 옵션

`-k` 옵션은 원본 파일을 복원하는 대신 강화 작업 후 파일에 작성한 모든 수동 변경사항을 자동으로 보존합니다. `-k` 옵션은 파일의 모든 불일치를 발견하고 주의사항을 생성하고 로그하며 파일을 원본으로 겹쳐 쓰지 않습니다. 취소되는 유일한 변경사항은 `jass-checksums.txt` 파일의 저장된 체크섬이 유효한 변경사항입니다.

이 옵션은 단점이 없지 않습니다. 예를 들어 종료 스크립트에 의해 수정된 파일의 서브세트가 나중에 수정되는 경우 시스템이 일치하지 않는 상태가 될 수 있습니다.

`remove-unneeded-accounts.fin` 종료 스크립트를 고려하십시오. 이 스크립트는 시스템의 `/etc/passwd` 및 `/etc/shadow` 파일을 모두 수정합니다. 사용자가 강화 작업이 종료된 후 수동으로 암호를 변경하는 경우 `/etc/shadow` 파일과 연관된 체크섬이 Solaris Security Toolkit 소프트웨어에 의해 저장된 값과 일치하지 않습니다. 결국 보존 옵션이 사용되는 경우 `/etc/passwd` 파일만이 원래 상태로 다시 복사됩니다. `/etc/shadow` 파일은 현재 양식으로 유지됩니다. 두 파일은 더 이상 일치하지 않습니다.

출력 파일 옵션

`-o` 출력파일 옵션은 `jass-execute` 실행의 콘솔 출력을 별개 파일인 출력파일로 경로 재지정합니다.

이 옵션은 `JASS_REPOSITORY` 디렉토리에 보존되는 로그에는 아무 영향도 주지 않습니다. 이 옵션은 특히 느린 터미널 연결을 통해 수행될 때 Solaris Security Toolkit 실행 취소 실행에 의해 생성되는 상당한 양의 출력이 있기 때문에 유용합니다.

정숙 출력 옵션

`-q` 옵션은 Solaris Security Toolkit 소프트웨어가 화면에 출력을 표시하지 않게 합니다. 이 옵션은 `JASS_REPOSITORY` 디렉토리에 보존되는 로그에는 아무 영향도 주지 않습니다. `-o` 옵션과 비슷하게, 이 옵션은 특히 `cron` 작업을 통하거나 느린 네트워크 연결을 통해 Solaris Security Toolkit을 실행할 때 도움이 됩니다.

전자 우편 통지 옵션

-m 전자 우편 주소 옵션은 Solaris Security Toolkit 소프트웨어에 완료된 작업의 사본을 전자 우편 주소로 보내도록 지시합니다. 전자 우편 보고서는 다른 옵션을 사용하여 시스템에 생성되는 모든 로그에 추가로 작성됩니다.

시스템 변경 실행 취소

때로는 하나 또는 복수의 Solaris Security Toolkit 강화 작업 중에 작성된 변경사항을 취소해야 합니다. 강화 작업 중에 작성된 변경이 시스템에 부정적인 영향을 주는 경우 변경사항을 실행 취소 하십시오.

예를 들어 강화 작업 후 NFS 같은 필수 서비스가 사용 불가능함을 발견하는 경우 강화 작업을 실행 취소 하십시오. 그런 다음 NFS를 사용 가능하게 하고 개정된 보안 프로파일로 강화 작업을 반복하십시오.

이 절은 하나 또는 복수 강화 작업 중에 작성된 변경을 취소하기 위한 지시를 제공합니다. 강화 작업을 효과적으로 취소하기 위한 한계 및 요구사항이 있음을 주의하십시오. 56페이지의 "시스템 변경 실행 취소 요구사항"을 참조하십시오.

▼ Solaris Security Toolkit 작업 실행 취소

1. **시스템을 백업하고 재부트합니다.**

   시스템이 알려지고 작동 중인 상태로 되돌아가거나 되돌아갈 수 있도록 각 실행 취소 작업 전에 시스템을 재부트하고 백업하십시오.

2. **jass-execute -u 명령과 함께 사용하려는 옵션을 판별합니다.**

   58페이지의 "실행 취소 기능에서 옵션 사용"을 참조하십시오.

   다음 지시는 jass-execute -u 명령을 사용 중이라고 가정합니다.

3. **표준 -u 옵션을 사용하여 하나 이상의 강화 작업을 실행 취소 하려면 JASS_HOME_DIR에서 다음 명령을 입력하십시오.**

```
# ./jass-execute -u
```

Solaris Security Toolkit 소프트웨어는 JASS_REPOSITORY에 있는 모든 목록 파일을 찾아서 각 강화 작업에 대한 정보를 수집합니다. 목록 파일이 비어 있거나 존재하지 않는 경우 실행 취소 할 변경이 없고 작업이 생략되었다고 가정합니다. 또한 jass-undo-

log.txt라는 파일이 목록 파일과 동일한 디렉토리에 존재하는 경우 작업이 이미 취소되었으므로 해당 작업이 생략되었다고 가정합니다. 수집 프로세스가 완료된 후 그 결과가 표시됩니다. 다음은 예제 출력입니다.

코드 예 **4-2** 실행 취소할 수 있는 작업의 예제 출력

```
# ./jass-execute -u
[NOTE] Executing driver, undo.driver
Please select a JASS run to restore through:
1. January 24, 2003 at 13:57:27
(/var/opt/SUNWjass/run/20030124135727)
2. January 24, 2003 at 13:44:18
(/var/opt/SUNWjass/run/20030124134418)
3. January 24, 2003 at 13:42:45
(/var/opt/SUNWjass/run/20030124134245)
4. January 24, 2003 at 12:57:30
(/var/opt/SUNWjass/run/20030124125730)

Choice? (‘q’ to exit)?
```

이 예제에서 4개의 개별 강화 작업이 발견됩니다. 이들 작업이 시스템에 변경을 작성했고 실행 취소되지 않았습니다. 강화 작업의 목록은 항상 연대기 역순으로 표시됩니다. 목록의 첫 번째 항목은 가장 최근의 강화 작업입니다.

4. 출력을 검토하여 실행 취소 하려는 작업을 판별한 후 대응하는 번호를 입력합니다.

임의의 선택된 항목에 대해 Solaris Security Toolkit 소프트웨어는 선택된 값보다 작거나 같은 색인 번호를 갖는 각 작업을 취소합니다. 즉, 실행 취소 작업이 가장 최근의 강화 작업으로 시작하여 사용자가 선택한 작업까지 계속해서 변경이 원래 작성된 역순으

로 변경사항을 실행 취소 합니다. 앞의 예제를 지침으로 사용하여 작업 3을 선택하는 경우 `undo run`은 먼저 작업 1에 대한 변경을 취소한 후 작업 2에 대한 변경을 취소하려 이동하고 작업 3에 대한 변경을 취소하여 종료합니다.

다음 예제는 실행 취소 작업이 두 개의 목록 파일 항목을 처리할 때 생성되는 출력을 보여줍니다.

코드 예 **4-3** 복수 목록 파일 항목을 처리하는 실행 취소 작업의 예제 출력

```
[...]

================================================================
undo.driver: Performing UNDO of
//var/opt/SUNWjass/run/20030124135727.
================================================================

[...]

================================================================
undo.driver: Undoing Finish Script: update-cron-allow.fin
================================================================

[NOTE] Undoing operation COPY.
cp -p /etc/cron.d/cron.allow.JASS.20030125223417
/etc/cron.d/cron.allow
rm -f /etc/cron.d/cron.allow.JASS.20030125223417

[NOTE] Removing a JASS-created file.
rm -f /etc/cron.d/cron.allow

[...]
```

이 예제에서 Solaris Security Toolkit 소프트웨어는 복사 조작을 실행 취소하고 강화 작업 중에 추가된 파일을 제거합니다. 실행 취소 작업의 출력은 시스템을 복원하기 위한 실제 명령을 문서화하므로, 시스템 구성의 문제를 해결해야 할 경우 프로세스를 명확히 이해하고 참조할 수 있습니다.

실행 취소 작업은 모든 작업 및 대응하는 목록 파일이 처리되고 변경사항이 취소될 때까지 계속됩니다.

`JASS_REPOSITORY`에 위치한 모든 목록 파일을 찾아서 각 강화 작업에 대한 정보를 수집하는 Solaris Security Toolkit 소프트웨어 외에, Solaris Security Toolkit 소프트웨어는 수정된 각 파일의 체크섬을 비교합니다. 체크섬 파일의 모든 불일치는 주의사항이 생성되고 로깅되게 합니다. 이들 파일에 대해 실행 취소 작업은 사용하고자 하는 조치를 묻습니다.

5. 실행 취소 작업이 예외(강화 작업 후 수동으로 변경된 파일)를 발견하는 경우 옵션 중 하나를 입력합니다.

   다음은 예외 및 예외 처리용 선택사항을 보여주는 예제 출력입니다.

코드 예 **4-4** 실행 취소 예외의 예제 출력

```
[...]

========================================================================
undo.driver: Undoing Finish Script: install-templates.fin
========================================================================

[NOTE] Undoing operation COPY.
cp -p /etc/skel/local.login.JASS.20030125223413
/etc/skel/local.login
rm -f /etc/skel/local.login.JASS.20030125223413

[NOTE] Undoing operation COPY.
[WARN] Checksum of current file does not match the saved value.
[WARN]    filename = /etc/.login
[WARN]    current = 3198795829:585, saved = 1288382808:584

Please select the course of action:

1. Backup.  Save current file before restoring original.
2. Keep.    Keep the current file, making no changes.
3. Force.   Ignore manual changes and overwrite current file.

Enter 1, 2, or 3:
```

이 예제에서 항목 1을 선택하는 경우 다음 출력이 표시됩니다.

코드 예 **4-5** 실행 취소 중 백업 옵션 선택의 예제 출력

```
Enter 1, 2, or 3: 1

[NOTE] BACKUP specified, creating backup copy of /etc/.login.
[NOTE] File to be backed up is from an undo operation.
[NOTE] Copying /etc/.login to /etc.login.BACKUP.JASS.20030125224926
cp -p /etc/.login.JASS.20030125223413 /etc/.login
rm -f /etc/.login.JASS.20030125223413

[...]
```

임의의 강화 작업 후에 수동으로 수정된 파일에 관한 적당한 조치를 취하십시오.

실행 취소 작업에서 수정된 파일이 발견되고 해당 파일을 겹쳐 쓰지 않을 것을 선택하는 경우 시스템을 재부트하기 전에 이를 조정하십시오.

주 – 이 예제에서 수정된 파일은 다음의 새 이름으로 저장됩니다. `/etc/.login.BACKUP.JASS.20030125224926.` 실행 취소 작업이 완료된 후 해당 파일을 `/etc/.login`에 비교하여 추가 조정이 필요한지 판별하십시오.

6. 계속하기 전에 모든 예외를 조정합니다.

7. 모든 예외를 조정한 후 시스템을 재부트합니다.

   시스템이 강화되기 전에 사용 가능한 서비스를 중지하고 시작하기 위해서는 시스템 재부트가 필요합니다.

5장

JumpStart 서버 구성 및 관리

이 장은 Solaris Security Toolkit 소프트웨어를 사용하기 위한 JumpStart 서버 구성 및 관리 정보를 제공합니다. Sun의 네트워크 기반 Solaris OS 설치 체계인 JumpStart 기술은 설치 프로세스 중에 Solaris Security Toolkit 소프트웨어를 실행할 수 있습니다.

Solaris Security Toolkit의 JumpStart 모드는 버전 2.1 이후의 Solaris OS 제품에 사용 가능한 JumpStart 기술을 기초로 합니다. JumpStart 기술은 Solaris OS 및 시스템 소프트웨어 설치, 시스템의 정확성 및 표준화 촉진을 완전히 자동화하여 복잡성을 관리하는 데 도움을 줍니다. 시스템을 빨리 설치 및 전개하는 요구사항을 충족시키는 방법을 제공합니다.

JumpStart 기술 사용의 장점은 시스템 보안 영역에서 두드러집니다. Solaris Security Toolkit 소프트웨어에 JumpStart 기술을 사용하여 자동화된 Solaris OS 설치시 시스템을 보안할 수 있습니다. 이 기술은 시스템 설치시 시스템 보안이 표준화되고 주소화되도록 도와줍니다. JumpStart 기술에 대한 자세한 정보는 Sun BluePrints 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

이 장에서는 다음 주제를 다룹니다.

JumpStart 서버 및 환경 구성

JumpStart 환경에서 사용하기 위해서는 JASS_HOME_DIR(tar 다운로드의 경우) 또는 /opt/SUNWjass(pkg 다운로드의 경우) 중 하나의 Solaris Security Toolkit 소스를 JumpStart 서버의 기본 디렉토리에 복사해야 합니다. 기본 디렉토리는 JumpStart 서버의 /jumpstart입니다. 이 작업이 완료된 후 JASS_HOME_DIR이 JumpStart 서버의 기본 디렉토리가 됩니다.

이 절에서는 사용자가 JumpStart 기술에 익숙하고 기존 JumpStart 환경을 사용할 수 있다고 가정합니다.

Solaris Security Toolkit 소프트웨어를 JumpStart 구조에 통합하기 위해서는 몇 개의 단계만 필요합니다.

▼ JumpStart 모드 구성

1. **JumpStart 서버의 루트 디렉토리에 Solaris Security Toolkit 소스를 복사합니다.**

   예를 들어 Solaris Security Toolkit 아카이브가 JASS_REPOSITORY로 추출되었고 JumpStart 서버 루트 디렉토리가 /jumpstart인 경우 다음 명령이 Solaris Security Toolkit 소스를 복사합니다.

```
# pwd
/opt/SUNWjass
# tar cf - . | (cd /jumpstart; tar xf -)
```

   일반적으로 Solaris Security Toolkit 소프트웨어는 보통은 JASS_HOME_DIR이기도 한 JumpStart 서버의 SI_CONFIG_DIR에 설치됩니다.

2. **Solaris 2.5.1 OS sysidcfg 파일을 수정하는 경우 해당 수정사항을 JASS_HOME_DIR/Sysidcfg/Solaris_2.5.1 디렉토리의 하나로 만듭니다.**

   Solaris 2.5.1 OS를 사용 중인 경우, JASS_HOME_DIR/Sysidcfg/Solaris_2.5.1의 sysidcfg 파일은 직접 사용할 수 없습니다. 이 Solaris 버전은 SI_CONFIG_DIR에 있고 별도의 하위 디렉토리에 있지 않은 sysidcfg 파일만 지원하기 때문입니다. Solaris 2.5.1 OS에서 이 제한을 다루기 위해 Solaris Security Toolkit 소프트웨어에는 SI_CONFIG_DIR/sysidcfg이 있으며, 이것은 JASS_HOME_DIR/Sysidcfg/Solaris_2.5.1/sysidcfg 파일에 링크됩니다.

3. 다음 명령으로 JASS_HOME_DIR/Drivers/user.init.SAMPLE을 JASS_HOME_DIR/Drivers/user.init로 복사합니다.

```
# pwd
/jumpstart/Drivers
# cp user.init.SAMPLE user.init
```

4. 멀티홈 **JumpStart** 서버에 문제점이 있는 경우 JASS_PACKAGE_MOUNT 및 JASS_PATCH_MOUNT에 대한 두 항목을 JASS_HOME_DIR/Patches 및 JASS_HOME_DIR/Packages 디렉토리에 대한 올바른 경로로 수정합니다.

5. SI_CONFIG_DIR/path/to/JASS 같이 SI_CONFIG_DIR의 하위 디렉토리에 **Solaris Security Toolkit** 소프트웨어를 설치하려는 경우 user.init 파일에 다음을 추가합니다.

```
if [ -z "${JASS_HOME_DIR}" ]; then
       if [ "${JASS_STANDALONE}" = 0 ]; then
              JASS_HOME_DIR="${SI_CONFIG_DIR}/path/to/JASS"
       fi
fi
export JASS_HOME_DIR
```

6. **Solaris Security Toolkit** 드라이버**(**예**:** 기본 secure.driver**)**를 선택하거나 작성합니다.
   - hardening.driver 및 config.driver에 나열되는 모든 스크립트가 사용되어야 하는 경우 Drivers/secure.driver 경로를 rules 파일에 추가하십시오.
   - 선택된 스크립트만이 사용될 경우 해당 파일을 복사한 후 사본을 수정하십시오.

7. 드라이버를 완료한 후 rules 파일에 적당한 항목을 작성합니다.

   항목은 다음과 비슷해야 합니다.

```
hostname imbulu - Profiles/core.profile Drivers/secure.driver
```

주의 – Solaris Security Toolkit 소프트웨어에 포함된 원본 스크립트를 절대 수정하지 마십시오. Solaris Security Toolkit 소프트웨어의 새 릴리스로의 효율적인 마이그레이션을 허용하기 위해 원본 파일과 사용자 정의 파일을 개별적으로 유지하십시오.

Solaris Security Toolkit 소프트웨어를 성공적으로 기존 JumpStart 환경에 통합하기 위해 또 다른 수정이 필요할 수 있습니다.

8. **Solaris Security Toolkit** 소프트웨어와 함께 제공되는 `sysidcfg` 파일을 사용하여 **JumpStart** 클라이언트 설치를 자동화하려는 경우 적용 가능한지 검토합니다**.**

   JumpStart 서버에서 `sysidcfg` 파일을 구문 분석하는 중에 오류가 발생하는 경우 파일의 전체 내용이 무시됩니다.

   이 절에 있는 모든 구성 단계를 완료한 후 클라이언트에서 JumpStart 기술을 사용하고 설치 프로세스 중에 OS를 강화하거나 최소화할 수 있습니다.

JumpStart 프로파일 템플리트 사용

JumpStart 프로파일 템플리트는 JumpStart 모드에서만 사용되는 파일입니다. 프로파일의 필수 및 선택적 내용은 Sun BluePrints 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

JumpStart 프로파일 템플리트를 개별 사이트 수정을 수행하는 예제로 사용하십시오. 프로파일을 검토하여 사용자 환경에서 사용해야 하는 변경 사항(있는 경우)을 판별하십시오.

프로파일의 사본을 작성한 후 사이트에 맞게 수정하십시오. Solaris Security Toolkit 소프트웨어에 대한 업데이트가 사용자 정의를 겹쳐 쓸 수 있으므로 원본을 수정하지 마십시오.

다음 JumpStart 프로파일이 Solaris Security Toolkit 소프트웨어에 포함되어 있습니다.

- `32-bit-minimal.profile`
- `core.profile`
- `end-user.profile`
- `developer.profile`
- `entire-distribution.profile`
- `oem.profile`
- `minimal-Sun_ONE-WS-Solaris*.profile`
- `minimal-SunFire_Domain*.profile`

다음 하위 절은 이러한 프로파일에 대해 설명합니다.

`32-bit-minimal.profile`

이 JumpStart 프로파일은 32비트 최소화 시스템에 대한 상대적으로 일반적인 JumpStart 프로파일입니다. 이 프로파일은 최소화된 시스템의 개발을 위한 합리적인 시작점이며 `minimal-Sun_ONE-WS-Solaris* .profile` 최소화 스크립트에 대한 시작점으로 사용되었습니다.

core.profile

이 JumpStart 프로파일은 가장 작은 Solaris OS 클러스터인 SUNWCreq를 설치합니다. 디스크의 분할이 루트 및 스왑 파티션을 포함하도록 지정하는 것 외에 다른 구성 수정은 하지 않습니다.

end-user.profile

이 JumpStart 프로파일은 End User Solaris OS 클러스터인 SUNWCuser 및 프로세스 회계가 제대로 작동하기 위해 필요한 두 개의 Solaris OS 패키지를 설치합니다. 또한 루트 및 스왑 파티션만 포함하도록 디스크 분할이 정의됩니다.

developer.profile

이 JumpStart 프로파일은 Developer Solaris OS 클러스터 SUNWCprog 및 프로세스 회계가 제대로 작동하기 위해 필요한 두 개의 Solaris OS 패키지를 설치합니다. core.profile 정의에서와 같이 Solaris OS 클러스터 외에 작성된 유일한 다른 구성 정의는 디스크 분할이 루트 및 스왑을 포함하는 것입니다.

entire-distribution.profile

이 JumpStart 프로파일은 Entire Distribution Solaris OS 클러스터 SUNWCall을 설치합니다. 다른 프로파일에서와 같이 루트 및 스왑 파티션을 포함하도록 디스크 분할이 정의됩니다.

oem.profile

이 JumpStart 프로파일은 OEM Solaris OS 클러스터 SUNWCXall을 설치합니다. 이 클러스터는 Entire Distribution 클러스터의 수퍼세트이며 OEM으로 제공되는 소프트웨어를 설치합니다.

minimal-Sun_ONE-WS-Solaris*.profile

다음의 모든 프로파일은 Sun BluePrints OnLine 기사 *Minimizing the Solaris Operating Environment for Security*를 기초로 합니다. 이 기사에서 다루어지는 모든 Solaris OS 버전은 특정 프로파일을 갖고 있습니다. 다음 JumpStart 프로파일은 기사에서 참조되는 것과 동일합니다.

- minimal-Sun_ONE-WS-Solaris.26.profile
- minimal-Sun_ONE-WS-Solaris7-32bit.profile
- minimal-Sun_ONE-WS-Solaris7-64bit.profile
- minimal-Sun_ONE-WS-Solaris8-32bit.profile
- minimal-Sun_ONE-WS-Solaris8-64bit.profile
- minimal-Sun_ONE-WS-Solaris9-64bit.profile

minimal-SunFire_Domain*.profile

아래의 모든 프로파일은 Sun BluePrints OnLine 기사 *Minimizing Domains for Sun Fire V1280, 12K, and 15K Systems*를 기초로 합니다. 다음 JumpStart 프로파일은 이 기사의 참조 내용과 동일합니다.

- minimal-SunFire_Domain-Apps-Solaris8.profile
- minimal-SunFire_Domain-Apps-Solaris9.profile
- minimal-SunFire_Domain-NoX-Solaris8.profile
- minimal-SunFire_Domain-NoX-Solaris9.profile
- minimal-SunFire_Domain-X-Solaris8.profile
- minimal-SunFire_Domain-X-Solaris9.profile

클라이언트 추가 및 제거

다음 정보는 JumpStart 모드에서 사용할 수 있는 스크립트에 대해 설명합니다. 모드는 JumpStart 서버의 rules 파일에 삽입된 Solaris Security Toolkit 드라이버에 의해 제어됩니다.

JumpStart 모드를 사용하도록 환경을 구성하지 않은 경우 68페이지의 "JumpStart 서버 및 환경 구성"을 참조하십시오.

add-client 스크립트

JumpStart 서버로부터 클라이언트 추가를 단순화하려면 Solaris Security Toolkit 소프트웨어에 포함되어 있는 이 스크립트를 사용하십시오. 명령과 옵션은 다음 문단에 설명되어 있지만 기초 JumpStart 기술은 설명되지 않습니다. Sun BluePrints 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

add-client 스크립트는 add_install_client 명령 주위의 랩퍼이며 다음 인수를 승인합니다.

사용 예제:

```
# add-client -c client -i server -m client-class -o client-OS -s sysidcfg
```

표 5-1은 `add-client` 명령에 대한 올바른 입력을 설명합니다.

표 **5-1** JumpStart `add-client` 명령

값	설명
-c 클라이언트	JumpStart 클라이언트의 해석 가능한 호스트 이름
-h	사용 정보 표시 다른 옵션 없이 사용합니다. 추가 옵션은 모두 무시됩니다.
-i 서버	이 JumpStart 클라이언트에 대한 JumpStart 서버 인터페이스의 IP 주소 또는 해석 가능한 호스트 이름. 값을 지정하지 않으면 로컬 호스트에서 사용할 수 있는 인터페이스 목록이 표시됩니다.
-m 클라이언트-클래스	JumpStart 클라이언트의 기계 클래스. 이 값은 `uname -m` 명령의 출력과 동일한 형식으로 되어 있습니다.
-o 클라이언트-*OS*	`JASS_HOME_DIR/OS` 디렉토리에서 사용 가능하며 클라이언트에 설치될 Solaris OS의 개정판. 값을 지정하지 않으면 `JASS_HOME_DIR/OS` 디렉토리의 사용 가능한 Solaris OS 버전 목록이 표시됩니다.
-s *sysidcfg*	시스템 식별 및 구성에 사용하려는 `sysidcfg` 파일이 들어있는 대체 디렉토리에 대한 선택적 경로 이름. 기본적으로 이 값은 버전이 이 클라이언트에 대해 지정된 OS에서 추출되는 `JASS_HOME_DIR/Sysidcfg/Solaris_version/` 디렉토리로 설정됩니다. 지정되는 경우 `JASS_HOME_DIR` 디렉토리에 상대적인 경로 이름을 사용해야 합니다. `sysidcfg` 파일에 대한 경로만 지정하십시오.
-v	이 프로그램에 대한 버전 정보
-?	사용 정보 표시 다른 옵션 없이 사용합니다. 추가 옵션은 모두 무시됩니다.

`jordan`이라는 JumpStart 클라이언트를 `nomex-jumpstart`라는 인터페이스의 Solaris 8 OS(4/01)을 사용하여 `nomex`라는 JumpStart 서버에 추가하려면 다음 `add-client` 명령을 사용합니다.

```
#./add-client -c jordan -o Solaris_8_2001-04 -m sun4u -i nomex-jumpstart
updating /etc/bootparams
```

sysidcfg 옵션을 사용하여 동일한 JumpStart 클라이언트(jordan)를 추가하려면 다음 명령을 사용합니다.

```
#./add-client -c jordan -o Solaris_8_2001-04 -m sun4u -i nomex-jumpstart -s
Hosts/jordan
updating /etc/bootparams
```

rm-client 스크립트

JumpStart 서버로부터 클라이언트 제거를 단순화하려면 Solaris Security Toolkit 소프트웨어에 포함되어 있는 이 스크립트를 사용하십시오. 명령과 옵션은 다음 문단에 설명되어 있지만 기초 JumpStart 기술은 설명되지 않습니다. JumpStart 기술에 대한 정보는 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

rm-client 스크립트는 add-client와 동일한 방법으로 rm_install_client 명령 주위의 랩퍼입니다.

사용 예제: **rm-client** [-c] *client*

여기서 *client*는 JumpStart 클라이언트의 해석 가능한 호스트 이름입니다.

표 5-2는 rm-client 명령에 대한 올바른 입력을 설명합니다.

표 **5-2** JumpStart rm-client 명령

값	설명
-c 클라이언트	JumpStart 클라이언트의 해석 가능한 호스트 이름
-h	사용 정보 표시 다른 옵션 없이 사용합니다. 추가 옵션은 모두 무시됩니다.
-v	이 프로그램의 버전 정보.
-?	사용 정보 표시 다른 옵션 없이 사용합니다. 추가 옵션은 모두 무시됩니다.

jordan이라는 JumpStart 클라이언트를 제거하려면 다음 rm-client 명령을 사용하십시오.

```
# ./rm-client -c jordan
removing jordan from bootparams
```

6 장

시스템 보안 감사

이 장은 Solaris Security Toolkit 소프트웨어를 사용하여 시스템의 보안을 감사(검증)하는 방법에 대해 설명합니다. 강화 후에 설정된 보안 프로파일에 대해서는 이 장의 정보와 절차를 사용하십시오. 이미 전개된 시스템의 경우 이 장에 있는 정보를 사용하여 강화하기 전에 보안을 평가할 수 있습니다.

주 – 감사라는 용어는 이 장과 설명서에서 보안 상태를 사전 정의된 보안 프로파일과 비교하여 보안 상태를 검증하는 Solaris Security Toolkit 소프트웨어의 자동화된 프로세스를 정의하는 데 사용됩니다. 그러나, 이 용어의 사용이 감사 옵션 사용후 시스템이 완벽하게 안전함을 보장하는 것은 아닙니다.

이 장에서는 다음 주제를 다룹니다.

보안 유지

보안 유지는 주기적으로 검토하고 확인해야 하는 진행 중 프로세스입니다. 시스템에 대한 기본 보안 구성이 시간이 지남에 따라 점차 열리기 때문에 보안 시스템 유지 보수는 경계가 필요합니다. (보안 유지에 대한 자세한 정보는 30페이지의 "시스템 보안 유지보수"를 참조하십시오.)

사용자 경험 및 요청을 기초로 지정된 보안 프로파일에 대한 준수 레벨을 판별하여 Solaris Security Toolkit 소프트웨어가 시스템의 보안 상태를 감사하기 위한 자동화된 방법을 개발했습니다.

주 – 이 방법은 `jass-execute -a` 명령을 사용하여 독립형 모드에서만 사용할 수 있으며 JumpStart 설치 중에는 사용할 수 없습니다.

수동 또는 자동으로(예를 들어 `cron` 작업이나 `rc` 스크립트를 통해) 주기적으로 시스템의 보안 상태를 감사하십시오. 예를 들어 새로운 설치를 강화한 후 5일 후에 Solaris Security Toolkit 소프트웨어 감사 명령(`jass-execute -a` 드라이버이름)을 실행하여 시스템 보안이 보안 프로파일에 의해 정의된 상태에서 변경되었는지 판별하십시오.

보안을 감사하는 빈도는 환경의 중요성과 보안 방침에 따라 달라집니다. 일부 사용자는 매 시간마다, 매일 또는 1개월에 한 번만 감사를 실행합니다. 일부 사용자는 매 시간마다 소형 스캔(제한된 수의 검사)을 실행하고 하루에 한 번 전체 스캔(가능한 모든 검사 사용)을 실행합니다.

필수 구성요소를 감사하여 전개된 시스템의 보안 상태를 유지하십시오. 보안 상태를 주기적으로 감사하지 않는 경우 원하는 보안 상태를 모르고서 또는 악의적으로 변경하는 엔트로피 또는 수정으로 인해 구성이 종종 시간에 따라서 변화합니다. 주기적 검토가 없으면 이들 변경은 검출되지 않고 수정되지 않습니다. 결과적으로 시스템은 덜 안정적이고 더 취약하게 됩니다.

주기적 감사 외에, 업그레이드, 패치, 및 기타 중요한 시스템 구성 변경 후에 감사를 수행하십시오.

강화하기 전 시스템 검토

일부 경우에는 전개된 시스템을 강화하기 전에 해당 시스템의 보안 상태를 검토하는 것이 유용함을 알 수 있습니다. 예를 들어 다른 사람이 관리한 전개된 시스템에 대한 책임을 지는 경우 시스템 상태를 파악하고 필요한 경우 다른 시스템에서 사용되는 동일한 보안 프로파일을 준수하게 만들 수 있도록 시스템의 상태를 조사하십시오.

보안 감사 사용자 정의

감사 옵션은 시스템 상태 평가를 위한 높은 융통성과 확장성을 갖는 체계를 제공합니다. 강화 스크립트에서와 같이 감사 스크립트의 조치를 사용자 정의할 수 있습니다. 예를 들어 환경 변수를 사용자 정의하고 프레임워크와 지원 프로그램 기능을 사용자 정의하고 새로운 점검을 추가하고 감사 프레임워크에 기능을 추가할 수 있습니다.

대체로 표준 및 제품 특정 감사 스크립트는 해당 환경에 대한 감사를 사용자 정의할 템플리트로 적합합니다. 이 시나리오의 경우 드라이버, 종료 스크립트, 환경 변수 및 파일 템플리트를 통해 감사 스크립트 조치를 사용자 정의하십시오. 이러한 사용자 정의 변경

은 아주 간단하며 코드를 수정하지 않고도 가능합니다. 강화를 위해 수행하는 모든 변경 사항은 사용자가 감사를 수행할 때 Solaris Security Toolkit 소프트웨어에 의해 자동으로 알려집니다.

때때로 Solaris Security Toolkit 소프트웨어가 제공하지 않은 검사나 기능을 추가할 필요가 있을 수도 있습니다. 이 시나리오의 경우 감사 스크립트에 검사나 새 기능을 추가하십시오. (대응하는 종료 스크립트에서 관련된 변경을 수행할 수도 있습니다.) 경우에 따라 코드를 수정해야 할 수 있습니다. 코드 추가 및 수정을 수행할 때 버그 및 실패를 도입하지 않도록 주의하십시오.

일부 사용자는 완전히 새로운 독점적 또는 사이트에 고유한 드라이버와 스크립트를 작성해야 합니다. 새로운 드라이버 및 스크립트를 코드화할 때 템플리트와 예제를 지침으로 사용하십시오. 감사 옵션을 사용할 때 사이트 특정 드라이버, 종료 스크립트, 변수 및 기능이 Solaris Security Toolkit 소프트웨어에 자동으로 알려지지 않음을 주의하십시오. 예를 들어 사이트에 고유한 종료 스크립트인 `abcc-nj-install-foo.fin`이 들어 있는 `abcc-nj-secure.driver`라는 사이트 특정 드라이버를 추가하는 경우 사이트 특정 감사 스크립트 `abcc-nj-install-foo.aud`를 작성해야 합니다. 비슷하게 감사 스크립트만으로 시작하는 경우 대응하는 종료 스크립트를 작성해야 합니다.

새로운 드라이버, 스크립트, 변수 및 기능을 사용자 정의하거나 작성하려면 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.

예를 들어 Solaris Security Toolkit 소프트웨어가 설치하지 않는 패치를 추가할 필요가 있을 수 있습니다. 표준 또는 제품 특정 템플리트 중 하나를 확장하거나 고유한 것을 작성할 수 있습니다. 자체 템플리트를 작성하는 경우 패치를 추가하는 종료 스크립트를 작성한 후 대응하는 감사 스크립트를 작성하여 패치 설치를 점검하십시오.

보안 감사 준비

이 장의 지시 및 지침을 사용하려면 보안 프로파일이 필요합니다. 보안 프로파일 개발 및 구현에 대한 정보는 2 장을 참조하십시오.

다양한 보안 프로파일 템플리트가 Solaris Security Toolkit 배포에 드라이버로서 포함되어 있습니다. 이 설명서의 앞에서 언급한 것처럼 기본 보안 프로파일 및 이들 드라이버로 작성된 변경이 사용자 시스템에 적합하지 않을 수 있습니다. 일반적으로 이들 드라이버에 의해 구현되는 보안 프로파일이 보안을 위한 "최고 수위선" 입니다. 이에 의하면 필수가 아닌 서비스를 비활성화 하고 기본적으로 사용 불가능한 선택적 보안 기능을 활성화 함을 의미합니다.

많은 Solaris Security Toolkit 소프트웨어 사용자는 표준 및 제품 특정 보안 프로파일 템플리트가 해당 환경에 적용할 수 있음을 알 수 있습니다. 이 경우에 해당하는 경우 어떤 보안 프로파일이 사용자가 원하는 보안 상태에 가장 가까운지를 판별하고 시스템 평가 및 강화에 사용하십시오.

사용자 환경에 대해 보안 프로파일 템플리트를 검토 및 사용자 정의하거나 새로 개발하십시오. 보안 프로파일 사용자 정의에 대한 방법과 지침은 *Solaris Security Toolkit 4.1 Reference Manual*에 있습니다. 이 접근법은 사용자 조직에 맞는 보안 상태를 제공하며, 보안 평가 중에 반환되는 거짓 오류의 양을 최소화합니다. 예를 들어 텔넷을 사용해야 함을 알고 있는 경우 보안 평가를 수행할 때 소프트웨어가 텔넷을 취약점으로 고려하지 않도록 보안 프로파일을 사용자 정의할 수 있습니다. 예를 들어 인증 및 암호화를 위해 Telnet with Kerberos를 사용하는 사이트가 텔넷 사용을 취약점으로 간주하지 않습니다.

옵션 사용 및 감사 출력 제어

이 절에서는 감사 실행 실행에 사용 가능한 옵션과 출력 제어용 옵션에 대해 설명합니다. 이 절에는 다음 항목이 들어있습니다.

- 78페이지의 "명령줄 옵션"
- 82페이지의 "배너 및 메시지 출력"
- 84페이지의 "호스트 이름, 스크립트 이름 및 시간 소인 출력"

명령줄 옵션

보안 프로파일에 대해 시스템을 감사하는 사용 예제:

```
# jass-execute -a driver [ -V [0-4]] [ -q | -o output-file ]
[ -m email-address ]
```

Solaris Security Toolkit 소프트웨어 감사 명령을 실행할 때 표 6-1에 나열된 다음 옵션을 사용할 수 있습니다.

표 **6-1** 감사 명령과 함께 명령줄 옵션 사용

옵션	설명
`-a`	시스템이 보안 프로파일을 준수하는지 판별합니다.
`-h`	사용 가능한 옵션의 개요를 제공하는 `jass-execute` 도움말 메시지를 표시합니다.
`-m`	출력을 전자 우편 주소로 보냅니다.
`-o`	출력을 파일로 보냅니다.
`-q`	콘솔에 대한 출력 표시를 금지합니다. 또한 정숙 옵션이라고도 합니다.
`-V`	감사 실행에 대한 상세(verbosity) 레벨을 지정합니다.

jass-execute -a 명령과 함께 사용할 수 있는 옵션에 대한 자세한 정보는 다음 절을 참조하십시오.

- 79페이지의 "도움말 표시 옵션"
- 80페이지의 "전자 우편 통지 옵션"
- 80페이지의 "출력 파일 옵션"
- 80페이지의 "정숙 옵션"
- 81페이지의 "상세 옵션"

도움말 표시 옵션

-h 옵션은 사용 가능한 옵션의 개요를 제공하는 jass-execute 도움말 메시지를 표시합니다.

-h 옵션은 다음과 비슷한 출력을 작성합니다.

코드 예 **6-1** 예제 -h 옵션 출력

```
# ./jass-execute -h

To apply this Toolkit to a system, using the syntax:
  jass-execute [-r root_directory -p os_version ]
  [ -q | -o output_file ] [ -m e-mail_address ]
  [ -V [3|4] ] [ -d ] driver

To undo a previous application of the Toolkit from a system:
  jass-execute -u [ -b | -f | -k ] [ -q | -o output_file ]
     [ -m e-mail_address ] [ -V [3|4] ]

To audit a system against a pre-defined profile:
  jass-execute -a driver [ -V [0-4] ] [ -q | -o output_file ]
     [ -m e-mail_address ]

To display the history of Toolkit applications on a system:
  jass-execute -H

To display the last application of the Toolkit on a system:
  jass-execute -l

To display this help message:
  jass-execute -h
  jass-execute -?

To display version information for this program:
  jass-execute -v
```

전자 우편 통지 옵션

-m 전자 우편 주소 옵션은 실행이 완료될 때 출력이 Solaris Security Toolkit 소프트웨어에 의해 자동으로 전자 우편을 통해 보내질 수 있는 체계를 제공합니다. 전자 우편 보고서는 다른 옵션을 사용하여 시스템에 생성되는 모든 로그에 추가로 작성됩니다.

전자 우편 옵션을 사용하여 sunfire_15k_sc-config.driver를 호출하는 Solaris Security Toolkit 실행은 다음과 비슷합니다.

```
# ./jass-execute -m root -a sunfire_15k_sc-config.driver
[...]
```

출력 파일 옵션

-o 출력 파일 옵션은 jass-execute 실행의 콘솔 출력을 별개 파일인 출력 파일로 경로 재지정합니다.

이 옵션은 JASS_REPOSITORY 디렉토리에 보존되는 로그에는 아무 영향도 주지 않습니다. 이 옵션은 Solaris Security Toolkit 실행에 의해 생성되는 상당한 양의 출력이 있기 때문에 특히 느린 터미널 연결을 통해 수행될 때 유용합니다.

이 옵션은 -d, -u 또는 -a 옵션 중 하나와 함께 사용할 수 있습니다.

-o 옵션은 다음과 비슷한 출력을 표시합니다.

코드 예 **6-2** 예제 -o 옵션 출력

```
# ./jass-execute -o jass-output.txt -a secure.driver
[NOTE] Executing driver, secure.driver
[NOTE] Recording output to jass-output.txt
#
```

정숙 옵션

-q 옵션은 강화 작업 중에 표준 입출력(stdio) 스트림에 대한 Solaris Security Toolkit 출력을 비활성화 합니다.

이 옵션은 JASS_REPOSITORY 디렉토리에 보존되는 로그에는 아무 영향도 주지 않습니다. -o 옵션과 비슷하게, 이 옵션은 특히 cron 작업을 통하거나 느린 네트워크 연결을 통해 Solaris Security Toolkit 소프트웨어를 실행할 때 도움이 됩니다.

이 옵션은 -d, -u 또는 -a 옵션 중 하나와 함께 사용할 수 있습니다.

-q 옵션은 다음과 비슷한 출력을 작성합니다.

코드 예 **6-3** 예제 -q 옵션 출력

```
# ./jass-execute -q -a secure.driver
[NOTE] Executing driver, secure.driver
```

상세 옵션

-V 옵션은 감사 실행에 대한 상세(verbosity) 레벨을 지정합니다. 이 옵션은 감사에만 사용할 수 있습니다. 상세 레벨은 감사 실행의 결과를 표시하는 융통성이 높은 방법을 제공합니다. 예를 들어 100대의 시스템을 감사하는 경우 각 시스템에 대해 출력을 하나의 행으로 제한하여 어느 시스템이 감사를 통과하는지 간단히 판별할 수도 있습니다. 그런 다음 실패하는 시스템에 대해 문제점 영역에 집중하기 위해 확장된 출력을 생성하는 감사를 실행할 수 있습니다.

5가지 상세 레벨(0 - 4)이 -V 옵션으로 제어됩니다. 레벨이 올라갈수록 어떤 점검이 통과하고 어느 것이 실패하는지 보다 완전하게 이해하기 위해 사용할 수 있는 추가 세부사항을 제공합니다. 표 6-2에 상세 레벨이 설명되어 있습니다.

표 **6-2** 감사 상세 레벨

레벨	출력
0	통과 또는 실패를 나타내는 단일 행.
1	각 스크립트에 대해 통과 또는 실패를 나타내는 단일 행. 모든 스크립트 행 아래에 하나의 총계 점수 행.
2	각 스크립트에 대해 모든 점검 결과를 제공합니다.
3	배너 및 헤더 메시지를 포함하여 전체 출력을 제공하는 여러 행.
4	여러 행(레벨 3에서 제공되는 모든 데이터) 및 logDebug 로깅 기능에 의해 생성되는 모든 항목. 이 레벨은 디버깅용입니다.

주 – jass-execute -V 명령에 대한 기본 상세 레벨은 3입니다.

상세 레벨의 자세한 설명은 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.

배너 및 메시지 출력

배너 및 메시지를 보고하거나 생략하도록 Solaris Security Toolkit 감사 옵션을 구성할 수 있습니다. JASS_LOG_BANNER 변수는 상세 레벨 0-2로 사용할 수 없습니다. 이런 출력 옵션은 상세 레벨 3과 4에 적용합니다. 예를 들어, (JASS_LOG_FAILURE 변수)만을 보고하고 집중할 수 있도록 출력에서 통과 메시지(JASS_LOG_SUCCESS 변수)를 제거할 수 있습니다.

표 6-3은 로깅 변수를 통해 제어할 수 있는 배너 및 메시지 목록입니다. (로깅 변수에 대한 자세한 정보는 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.) 로깅 변수가 0으로 설정되면 해당 유형의 메시지에 대해 출력이 생성되지 않습니다. 반대로 로깅 변수가 1로 설정되면 메시지가 표시됩니다. 이러한 각 변수에 대한 기본 조치는 출력을 표시하는 것입니다. 표 6-3에서 로깅 변수에 대해 설명합니다.

표 **6-3** 감사 출력에 배너 및 메시지 표시

로깅 변수	로그 접두어	설명
JASS_LOG_BANNER	모든 배너 출력	이 매개변수는 배너 메시지의 표시를 제어합니다. 이들 메시지는 대개 등호("=") 또는 대시("-") 문자 중 하나로 구성되는 분리자로 둘러쌉니다.
JASS_LOG_ERROR	[ERR]	이 매개변수는 오류 메시지의 표시를 제어합니다. 0으로 설정되면 오류 메시지가 생성되지 않습니다.
JASS_LOG_FAILURE	[FAIL]	이 매개변수는 실패 메시지의 표시를 제어합니다. 0으로 설정되면 실패 메시지가 생성되지 않습니다.
JASS_LOG_NOTICE	[NOTE]	이 매개변수는 알림 메시지의 표시를 제어합니다. 0으로 설정되면 알림 메시지가 생성되지 않습니다.
JASS_LOG_SUCCESS	[PASS]	이 매개변수는 성공 또는 통과 상태 메시지의 표시를 제어합니다. 0으로 설정되면 성공 메시지가 생성되지 않습니다.
JASS_LOG_WARNING	[WARN]	이 매개변수는 경고 메시지의 표시를 제어합니다. 0으로 설정되면 경고 메시지가 생성되지 않습니다.

이들 옵션을 사용하면 특정 메시지만을 보아야 할 때 매우 유용합니다. 이들 옵션을 설정하여 중요하다고 생각하는 영역에 집중하면서도 출력을 최소화할 수 있습니다. 예를 들어 JASS_LOG_FAILURE(기본값인 1로 그대로 둠)를 제외한 모든 로깅 변수를 0으로 설정하면 감사가 logFailure에 의해 생성되는 실패에 대해서만 보고합니다.

코드 예 **6-4** 감사 실패만을 보고하는 예제 출력

```
# JASS_LOG_FAILURE=1
# export JASS_LOG_FAILURE
[setting of other parameters to 0 omitted]
# ./jass-execute -a secure.driver -V 2
update-at-deny        [FAIL] User test is not listed in
   /etc/cron.d/at.deny.
update-at-deny        [FAIL] Audit Check Total : 1 Error(s)
update-inetd-conf     [FAIL] Service ftp is enabled in
   /etc/inet/inetd.conf.
update-inetd-conf     [FAIL] Service telnet is enabled in
   /etc/inet/inetd.conf.
update-inetd-conf     [FAIL] Service rstatd is enabled in
   /etc/inet/inetd.conf.
update-inetd-conf     [FAIL] Audit Check Total : 3 Error(s)
```

호스트 이름, 스크립트 이름 및 시간 소인 출력

상세 레벨 0-2에 대한 호스트 이름, 스크립트 이름 및 시간 소인 정보를 포함하도록 Solaris Security Toolkit 감사 옵션을 구성할 수 있습니다. 예를 들어 감사할 시스템이 많은 경우 호스트 이름, 스크립트 이름 또는 시간 소인으로 출력을 정렬할 수 있기 원합니다. 표 6-4에 변수가 나열되어 있습니다.

표 **6-4** 호스트 이름, 스크립트 이름 및 시간 소인 감사 출력 표시

변수 이름	변수 설명
JASS_DISPLAY_HOSTNAME	이 매개변수를 1로 설정하면 Solaris Security Toolkit 소프트웨어가 각 로그 항목 앞에 시스템의 호스트 이름을 첨부합니다. 이 정보는 JASS_HOSTNAME 매개변수에 기초합니다. 기본적으로 이 매개변수는 비어 있으므로 Toolkit은 이 정보를 표시하지 않습니다.
JASS_DISPLAY_SCRIPTNAME	기본적으로 이 매개변수는 1로 설정되므로 Solaris Security Toolkit 소프트웨어가 각 로그 항목 앞에 현재 실행되고 있는 감사 스크립트의 이름을 첨부합니다. 이 매개변수를 다른 값으로 설정하면 Toolkit은 이 정보를 표시하지 않습니다.
JASS_DISPLAY_TIMESTAMP	이 매개변수를 1로 설정하면 Solaris Security Toolkit 소프트웨어가 각 로그 항목 앞에 감사 실행과 연관된 시간 소인을 첨부합니다. 이 정보는 JASS_TIMESTAMP 매개변수에 기초합니다. 기본적으로 이 매개변수는 비어 있으므로 소프트웨어는 이 정보를 표시하지 않습니다.

호스트, 스크립트 및 시간 소인 정보를 앞에 첨부하도록 Solaris Security Toolkit 소프트웨어를 구성함으로써 단일 시스템이나 시스템 그룹의 많은 감사 실행을 결합하고 핵심 데이터를 기초로 정렬할 수 있습니다. 이 정보를 사용하여 여러 시스템에 걸쳐있거나 전개 프로세스의 증상인 문제점을 찾을 수 있습니다. 예를 들어 관리자는 이 방식으로 정보를 사용하여 주어진 프로세스를 사용하는 모든 시스템 구축에 항상 동일한 실패한 검사가 있는지 알 수 있습니다.

예를 들어 JASS_DISPLAY_TIMESTAMP 매개변수를 1로 설정하고 JASS_DISPLAY_SCRIPTNAME 값을 0으로 설정하면 다음과 비슷한 출력이 생성됩니다.

코드 예 **6-5** 감사 로그 항목의 예제 출력

```
# JASS_DISPLAY_SCRIPTNAME=0
# JASS_DISPLAY_TIMESTAMP=1
# export JASS_DISPLAY_SCRIPTNAME JASS_DISPLAY_TIMESTAMP
# ./jass-execute -a secure.driver -V 2
20030101233525 [FAIL] User test is not listed in
   /etc/cron.d/at.deny.
20030101233525 [FAIL] Audit Check Total : 1 Error(s)
20030101233525 [FAIL] Service ftp is enabled in
   /etc/inet/inetd.conf.
20030101233525 [FAIL] Service telnet is enabled in
   /etc/inet/inetd.conf.
20030101233525 [FAIL] Service rstatd is enabled in
   /etc/inet/inetd.conf.
20030101233525 [FAIL] Audit Check Total : 3 Error(s)
```

보안 감사 수행

시스템에 대해 보안 평가를 주기적으로 수행하면 사용자가 구현한 보안 프로파일에 대해 시스템이 얼마나 가깝게 일치하는지의 벤치마크를 얻게 됩니다. 보안 평가 수행에 대한 가장 일반적인 시나리오는 새로운 설치를 수행한 후 보안 유지 작업으로 수행하는 것입니다. 단순히 시스템을 강화하는 데 사용한 동일한 강화 드라이버를 실행하도록 보안 평가 옵션을 디자인했지만, 이제는 -a 옵션을 사용하여 강화 중에 구현된 보안 프로파일과 비교한 현재 상태를 점검합니다. 이 디자인은 복잡하지 않고 유연합니다. 예를 들어 보안 프로파일을 업데이트할 때 후속 보안 평가는 업데이트된 보안 프로파일을 사용합니다.

또 다른 가능한 시나리오에서 사용자는 이미 전개된 시스템 보안의 책임자일 수도 있습니다. 해당 시스템을 강화하기 전에 보안 평가를 수행하고자 합니다. 이 시나리오에서는 사용자 고유의 보안 프로파일을 정의하거나 Solaris Security Toolkit 보안 프로파일 템플리트를 사용자 정의하거나 보안 프로파일 템플리트 중 하나를 있는 그대로 사용합니다.

▼ 보안 감사 수행

감사를 수행하기 전에 보안 프로파일을 정의 또는 선택해야 합니다. 자세한 정보는 77 페이지의 "보안 감사 준비"를 참조하십시오.

주의 – 이전에 강화하지 않은 전개된 시스템에 대해 보안 평가를 수행하려는 경우 먼저 시스템을 백업하고 재부트하여 시스템이 알려지고 작동 중이며 일관성 있는 구성 상태에 있는지 확인하십시오. 보안 평가를 계속하기 전에 이 예비 재부트 중에 검출되는 모든 오류나 경고를 정정하거나 기록해야 합니다.

1. 사용하려는 보안 프로파일(강화 드라이버)을 선택합니다.
   - 이전에 시스템을 강화한 경우 동일한 보안 프로파일을 사용하십시오.

     예를 들면 `secure.driver`.
   - 시스템을 강화한 적이 없는 경우 표준 보안 프로파일이나 자체 프로파일 중 하나를 사용하십시오.

     예를 들면 `secure.driver` 또는 `abccorp-secure.driver`.

   사용 가능한 전체 드라이버의 최신 목록의 경우 다음 웹 사이트에서 Solaris Security Toolkit 소프트웨어의 최신 버전을 다운로드하십시오.

   `http://www.sun.com/security/jass`

   표준 및 제품 특정 드라이버에 대한 정보는 *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오. 드라이버의 가장 최근 목록에 대해서는 Drivers 디렉토리를 참조하십시오.

2. 원하는 명령줄 옵션과 출력을 제어하기 원하는 방법을 결정합니다.

   78페이지의 "옵션 사용 및 감사 출력 제어"를 참조하십시오.

3. jass-execute -a 명령, 보안 프로파일의 이름 및 원하는 옵션을 입력합니다.
   다음은 sunfire_15k_sc-secure.driver를 사용한 예제 감사 실행입니다.

코드 예 **6-6** 감사 실행의 예제 출력

```
# ./jass-execute -a sunfire_15k_sc-secure.driver
[NOTE] Executing driver, sunfire_15k_sc-secure.driver

[...]

================================================================
sunfire_15k_sc-secure.driver: Audit script: enable-rfc1948.aud
================================================================

#----------------------------------------------------------------
# RFC 1948 Sequence Number Generation
#
# Rationale for Audit:
#
# The purpose of this script is to audit that the system is
# configured and is in fact using RFC 1948 for its TCP sequence
# number generation algorithm (unique-per-connection ID). This is
# configured by setting the 'TCP_STRONG_ISS' parameter to '2' in
# the /etc/default/inetinit file.
#
# Determination of Compliance:
#
[...]
#----------------------------------------------------------------

[PASS] TCP_STRONG_ISS is set to '2' in /etc/default/inetinit.
[PASS] System is running with tcp_strong_iss=2.

# The following is the vulnerability total for this audit script.

[PASS] Audit Check Total : 0 Error(s)

================================================================

# The following is the vulnerability total for this driver profile.

[PASS] Driver Total : 0 Error(s)

================================================================
sunfire_15k_sc-secure.driver: Driver finished.
================================================================

[PASS] Grand Total : 0 Error(s)
```

감사 실행이 시작될 때 Solaris Security Toolkit 소프트웨어는 JASS_HOME_DIR/Audit 디렉토리의 파일에 액세스합니다. JASS_HOME_DIR/Audit 및 JASS_HOME_DIR/Finish 디렉토리 둘 다에 있는 파일이 동일한 기본 파일 이름을 공유하지만 서로 다른 파일 이름 접미어를 갖습니다. driver.run 스크립트가 접미어를 .fin에서 .aud로 변경하여 자동으로 JASS_SCRIPTS 변수에 의해 정의되는 종료 스크립트를 감사 스크립트로 변환합니다.

감사 실행이 시작하고 Solaris Security Toolkit 소프트웨어의 상태를 초기화합니다. 작업 중에 액세스되는 각 드라이버가 모든 파일 템플리트 및 감사 스크립트의 상태를 평가합니다. 각 검사 결과는 각각 0이거나 0이 아닌 취약성 값으로 표시되는 성공 또는 실패 상태로 나타납니다. 대부분의 경우 실패는 숫자 1로 표시됩니다. 실행되는 각 스크립트가 스크립트에 들어있는 각 검사의 총 취약성 값을 토대로 총 보안 점수를 작성합니다. 또한 각 드라이버에 대한 총 취약성 값 결과가 드라이버 평가 완료시에 표시됩니다. 모든 점수의 총계는 작업이 종료할 때 제시됩니다.

보안 평가 옵션은 평가 작업이 시작되는 시점에서 시스템 상태의 포괄적인 보기를 제공합니다. Solaris Security Toolkit 소프트웨어는 구성 파일을 조사하여 시스템의 저장된 상태를 점검하고 프로세스 테이블 정보, 장치 드라이버 정보 등을 조사하여 시스템의 실행 중 상태를 점검합니다. Solaris Security Toolkit 소프트웨어는 각 파일이나 서비스의 존재 여부를 점검하고 서비스와 연관된 소프트웨어가 설치, 구성, 사용 가능 및 실행 중인지 점검합니다. 이 접근 방식은 시스템의 현재 상태에 대한 정확한 스냅샷을 작성합니다.

7장

시스템 보안

이 장은 새로운 시스템의 설치 및 보안에 대해 이전 장에서 제공된 정보와 전문 지식을 실현 가능한 시나리오에 적용하는 방법에 대해 설명합니다. 이 장은 Solaris 8 OS용 Check PointFirewall-1 NG를 갖는 Solaris Security Toolkit 소프트웨어를 전개하는 방법에 대해 설명합니다.

이 장에 있는 정보를 새로운 시스템 및 응용 프로그램 보안을 위한 지침 및 사례 시나리오로 사용하십시오.

Sun BluePrint 설명서와 온라인 기사는 여러 Sun 시스템의 최소화 및 강화 프로세스 과정을 이해하는데 유용합니다. 최신 제품 관련 설명서 및 기사는 다음 웹사이트를 참조하십시오.

`http://www.sun.com/blueprints`

이 장에서는 다음 주제를 다룹니다.

계획 및 준비

이 사례 연구에서 설명된 것처럼 최소화 및 보안된 시스템을 효과적으로 그리고 효율적으로 전개하려면 계획과 준비가 중요합니다. 기초적인 네트워크 기반구조, 방침 및 절차가 제대로 준비되어 있어야 합니다. 또한, 시스템의 지원 및 유지에 대해 정의하고 이

해해야 합니다. 계획 및 준비에 대한 자세한 정보는 2 장을 참조하십시오. 이 장에 설명된 시나리오는 시스템 관리자(SA)가 방화벽 시스템을 위해 Solaris OS 이미지의 최소화 및 강화를 위해 수행할 과정 및 작업에 대해 상세히 설명합니다.

이 시나리오에서, 시스템 관리자는 고객에게 방화벽 서비스를 제공려는 서비스 제공자를 위해 Check PointFirewall-1 NG 시스템을 구축 및 전개하기 위한 자동화되고 확장 가능한 솔루션을 만드는 작업을 합니다. 이 시나리오의 경우, 서비스 제공자의 요구사항과 고려사항은 다음과 같습니다.

- 서비스 제공자는 이러한 시스템을 많이 전개할 계획이므로 각 시스템을 구축하고 전개하는 시간은 매우 중요하며 능률적으로 수행해야 합니다.
- 시스템은 각 시스템의 내부 이더넷 인터페이스에 연결된 전용 관리 네트워크를 사용하여 설치됩니다. 이 네트워크는 서비스 제공업체 직원만이 사용할 수 있으며 가입자는 사용할 수 없습니다.
- 다른 모든 네트워크는 분리된 물리적 네트워크 인터페이스에 있으며 필터링됩니다.
- 관리 네트워크의 보안은 전개된 방화벽 시스템의 전체 보안에 매우 중요합니다.

이러한 요구사항에 근거하여 시스템 관리자는 JumpStart 기술과 Solaris Security Toolkit 소프트웨어를 사용하여 OS 이미지의 설치, 최소화 및 강화를 자동화하기로 결정합니다.

가정 및 제한사항

이 장은 이미 작동 중인 Solaris Security Toolkit 소프트웨어와 JumpStart 기술 설치를 사용 중이라고 가정합니다. 이 문서의 다른 장에 소프트웨어 설치를 위한 지시와 지침이 제공되어 있습니다. 해당되는 장을 참조하십시오.

이 장은 특정 응용 프로그램의 최소화 및 강화를 위해 사용자 정의 구성을 개발 중이라고 가정합니다. Solaris Security Toolkit 소프트웨어에는 해당 응용 프로그램에 대한 드라이버나 JumpStart 프로파일이 없습니다. 따라서, 이 응용 프로그램에 대해 사용자 정의된 드라이버 및 프로파일을 작성해야 합니다. 이 작업은 기존 드라이버와 프로파일을 복사한 후, 응용 프로그램에 맞게 수정하면 됩니다.

이 사례 시나리오의 경우, 시스템 관리자의 기술 레벨은 다음과 같습니다.

- OS 및 응용 프로그램 구성에 대한 충분한 지식과 경험이 있습니다.
- 구성을 테스트하는 방법과 세밀하게 조정하기 위한 조정 방법을 알고 있습니다.
- 클라이언트 시스템이 설치되는 JumpStart 환경을 구축하는 방법을 알고 있습니다. 자세한 내용은 Sun BluePrint 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.
- OS 최소화 기술에 익숙합니다. *Enterprise Security: Solaris Operating Environment Security Journal, Solaris Operating Environment Versions 2.5.1, 2.6, 7, and 8*을 참조하십시오.

- Solaris Security Toolkit 소프트웨어에 대한 기본 지식이 있고, 최소화와 강화 기술 및 지침을 모두 사용하여 사용자 정의된 구성을 구축하려고 합니다. 1 장을 참조하십시오.

시스템 환경

예제 시나리오는 다음 하드웨어 및 소프트웨어 환경을 기초로 합니다.

- Check PointFirewall-1 NG
- Solaris 8 OS
- JumpStart 기술
- Solaris OS 클러스터(`SUNWCreq`)
- Solaris Security Toolkit 소프트웨어
- SPARC 기술에 기초를 둔 플랫폼
- 최소 두 개의 이더넷 인터페이스

보안 요구사항

이 시나리오의 경우, 높은 레벨의 요구사항 및 소프트웨어 패키지가 확인되었으나, 모든 패키지의 특정 구성 요소 및 서비스가 확인되어야 합니다. 또한, 시스템을 관리하기 위해 필요한 Solaris OS 기능이 확인되어야 합니다.

다음 목록은 소프트웨어 구성요소가 사용된 방법에 대한 자세한 보기를 제공합니다.

- 원격 관리를 위한 Secure Shell
- 파일을 복사하기 위한 FTP
- 디스크를 미러링하기 위한 Solstice DiskSuite™ 소프트웨어
- 중앙 저장소로 전달된 SYSLOG 메시지

이 목록에서 보안 프로파일을 개발할 수 있습니다. 보안 프로파일 개발과 프로파일 템플릿 사용에 대한 상세 정보는 27페이지의 "Solaris Security Toolkit 프로파일 개발 및 구현"을 참조하십시오.

보안 프로파일 작성

보안 프로파일은 Solaris Security Toolkit 소프트웨어가 시스템의 보안 구성을 강화하고 최소화할 때 수행하는 보안 수정사항을 정의합니다. Solaris Security Toolkit에 포함된 표준 보안 프로파일 또는 드라이버는 최소화된 Check PointFirewall-1 NG 시스템에 대한 요구사항에 부합되지 않습니다. 따라서, 사용자 정의 보안 프로파일을 작성하여 적절한 시스템 수정사항을 적용해야 합니다.

이 시나리오의 경우, 보안 프로파일의 작성 프로세스가 해당 시나리오에 적합할 경우 이 장의 여러 절에 설명되어 있습니다. 첫 번째, 기존 드라이버에 기초하여 새 드라이버 파일을 작성합니다. 그런 다음 이전에 요약된 보안 요구사항을 준수하도록 새 드라이버를 수정합니다. 최소화는 92페이지의 "소프트웨어 설치"에서 설명되고 강화 수정은 99페이지의 "강화 구성 사용자 정의"에서 설명됩니다.

소프트웨어 설치

이 절은 소프트웨어 설치 프로세스를 설명합니다. 예제 시나리오를 위해 모든 예외사항 또는 시나리오 관련 지침을 제공합니다. 소프트웨어 설치에 대한 일반 지침을 보려면 이 안내서의 다른 부분을 참조하십시오.

주 – 다음에 있는 지시를 관련 상황을 취급하기 위한 템플리트로서 사용할 수 있습니다.

이 절에는 다음 작업이 포함됩니다.

- 92페이지의 "보안 소프트웨어 다운로드 및 설치"
- 93페이지의 "패치 설치"
- 93페이지의 "OS Cluster 지정 및 설치"

보안 소프트웨어 다운로드 및 설치

아래와 같이 JumpStart 서버에 Solaris Security Toolkit과 패치를 포함한 추가 보안 소프트웨어를 다운로드 및 설치하십시오.

▼ 보안 소프트웨어 다운로드 및 설치

1. **Solaris Security Toolkit** 소프트웨어와 추가 보안 소프트웨어를 다운로드 및 설치하십시오.

   36페이지의 "보안 소프트웨어 다운로드"를 참조하십시오.

2. 다운로드한 **Solaris Security Toolkit** 소프트웨어와 추가 보안 소프트웨어를 설치하십시오.

   43페이지의 "소프트웨어 설치 및 실행"을 참조하십시오.

주의 – 아직 Solaris Security Toolkit 소프트웨어를 실행하지 마십시오. 우선 다음 절에 설명되어 있는 추가 구성 및 사용자 정의를 수행하십시오.

패치 설치

OS 패치는 보안 취약성, 가용성 문제, 성능 관련 사항 또는 시스템의 다른 측면을 다룰 수 있습니다. 새로운 OS 설치시, 그리고 OS 설치후 지속적으로 적절한 패치가 설치되었는지 확인하십시오.

Solaris Security Toolkit 소프트웨어는 SunSolve Online에서 이용 가능한 Recommended and Security Patch Cluster의 설치 방법을 제공합니다. OS 관련 클러스터 패치에는 가장 일반적으로 사용되는 패치가 들어있습니다.

▼ 패치 설치

1. 최소한 **Recommended and Security Patch Cluster**를 `Patches` 디렉토리에 다운로드하고 압축을 해제합니다.

   `install-recommended-patches.fin` 스크립트가 강화 드라이버에 포함되는 경우 해당 패치 클러스터가 자동으로 설치됩니다.

   Check PointFirewall-1 NG에 추가 문제점이 있습니다. 이 응용 프로그램은 Recommended and Security Patch Cluster에 들어있지 않은 특정 패치가 필요합니다. Check PointFirewall-1 NG는 다음 패치를 필요로 합니다.

   - 108434
   - 108435

2. 패치 **108434** 및 **108435**의 설치를 자동화하기 위해, **SunSolve OnLine**에서 최신 버전을 다운로드하여 `Patches` 디렉토리에 저장합니다.

3. 각 패치의 이름과 함께 `add_patch` 지원 프로그램 기능을 호출하는 새로운 종료 스크립트(예: `fw1-patch-install.fin`)를 작성합니다.

   이 종료 스크립트는 두 개의 Check PointFirewall-1 NG 필수 패치 ID와 함께 적절한 지원 프로그램 기능을 호출합니다. 예를 들면,

```
#!/bin/sh

# add_patch 108434-10

# add_patch 108435-10
```

OS Cluster 지정 및 설치

OS 설치를 위한 디스크 레이아웃을 정의한 후, 설치할 Solaris OS 클러스터를 지정합니다. Solaris OS와 함께 사용 가능한 다섯 개의 설치 클러스터 `SUNWCreq`, `SUNWCuser`, `SUNWCprog`, `SUNWCall` 및 `SUNWCXall` 중 하나를 선택하십시오.

▼ OS 클러스터 지정 및 설치

1. 설치할 **OS** 클러스터를 지정합니다**.**

이 사례 시나리오의 목표가 최소화 및 전용 방화벽 디바이스를 구축하는 것이기 때문에 사용 가능한 Solaris OS 클러스터 중 가장 작은 `SUNWCreq`를 선택합니다. 이 패키지를 Core하고도 합니다.

이 클러스터는 상대적으로 적은 수의 패키지를 포함하므로 다른 패키지가 필요할 수도 있습니다. 이러한 다른 필수 패키지가 Solaris OS 클러스터 정의가 있는 프로파일에 포함되어야 합니다.

기준 프로파일 정의는 다음을 이전에 정의된 프로파일에 추가합니다.

```
cluster          SUNWCreq
```

`SUNWCreq` 설치 클러스터는 방화벽 Sun 서버가 올바르게 기능하기 위해 필수적이지 않은 패키지를 포함합니다. 작업 기준을 정한 후 이러한 여분의 패키지를 제거하십시오. Sun BluePrints OnLine 기사 "Minimizing the Solaris Operating Environment for Security: Updated for the Solaris 9 Operating Environment"를 참조하십시오.

2. 적절히 정의된 보안 프로파일과 함께 설치를 실행하여 패키지 종속 문제가 있는지 판별하십시오**.**

일부 패키지 종속성이 설치시 발생되었고, 다음 Solaris OS 패키지가 Check PointFirewall-1 NG에 필요하다고 결정하였습니다.

- `SUNWter` - 터미널 정보
- `SUNWadmc` - 시스템 관리 코어 라이브러리
- `SUNWadmfw` - 시스템 및 네트워크 관리 프레임워크
- `SUNWlibC` 및 `SUNWlibCx` - 필수 Check PointNG 응용 프로그램

프로파일의 전체 패키지 목록은 다음과 같습니다.

```
cluster          SUNWCreq
package          SUNWter         add
package          SUNWlibC        add
package          SUNWlibCx       add
package          SUNWadmc        add
package          SUNWadmfw       add
```

이 목록은 본 사례 연구에는 완벽하지만, 이 구성이 전개될 실제 환경에 따라 추가 패키지를 추가 또는 제거해야 할 수 있습니다.

105페이지의 "품질 보증 검사"에 설명된 것처럼 시스템이 기능 및 보안 측면 모두에서 검증될 때까지 패키지의 최종 목록은 수정이 필요할 수 있습니다. 수정이 필요한 경우 프로파일을 수정하고 시스템을 재설치한 후 테스트를 반복하십시오.

3. 이전 두 단계의 패키지 종속성을 기초로 `minimize-firewall.fin` 스크립트를 작성합니다.

JumpStart 서버 및 클라이언트 구성

이 절은 최소화를 위해 사용자 정의 보안 프로파일을 사용하도록 JumpStart 서버 및 클라이언트를 구성하는 방법에 대해 설명합니다. JumpStart 환경에서 Solaris Security Toolkit 소프트웨어 사용에 대한 자세한 정보는 5 장을 참조하십시오.

이 절에는 다음 작업이 포함됩니다.

기반구조 준비

기반구조를 준비하려면 다음 작업을 수행하십시오. 다음 작업은 기존 드라이버, 프로파일 및 종료 스크립트를 사용하는 클라이언트를 위한 기준 구성 작성 프로세스를 설명합니다. 이 기준이 적용된 후, 제대로 작동하는지 확인하고 선택된 응용 프로그램에 맞게 사용자 정의하십시오.

▼ 기반구조 준비

1. **JumpStart** 서버 및 환경을 구성합니다.

   자세한 지침에 대해서는 5 장을 참조하십시오.

2. `add-client` 명령을 사용하여 **JumpStart** 서버에 클라이언트를 추가합니다.

   코드 예 **7-1** JumpStart 서버에 클라이언트 추가

```
# pwd
/jumpstart
# bin/add-client -c jordan -o Solaris_8_2002-02 -m sun4u
-s nomex-jumpstart
cleaning up preexisting install client "jordan"
removing jordan from bootparams
updating /etc/bootparams
```

3. 적절한 **JumpStart** 프로파일 및 종료 스크립트를 지정하여 클라이언트를 위한 `rules` 파일 항목을 작성합니다. 예를 들면,

```
hostname jordan - Profiles/xsp-minimal-firewall.profile \
  Drivers/xsp-firewall-secure.driver
```

4. **Solaris Security Toolkit** 소프트웨어와 함께 제공된 파일을 복사하여 `xsp-minimal-firewall.profile`이라는 프로파일 파일 및 `xsp-firewall-secure.driver`라는 드라이버 파일을 작성합니다.

   이러한 파일을 작성해야 다음 단계를 성공적으로 완료할 수 있습니다. 처음에 이러한 파일은 Solaris Security Toolkit 소프트웨어와 함께 배포된 파일의 사본일 수 있습니다. Solaris Security Toolkit 소프트웨어와 함께 배포된 원본 파일을 절대 수정하지 마십시오. 다음 예제는 파일을 작성하는 방법을 보여줍니다.

   코드 예 **7-2** 프로파일 작성

```
# pwd
/jumpstart/Drivers
# cp install-Sun_ONE-WS.driver xsp-firewall-secure.driver
# cp hardening.driver xsp-firewall-hardening.driver
[...]
# pwd
/jumpstart/Profiles
# cp minimal-Sun_ONE-WS-Solaris8-64bit.profile \
     xsp-minimal-firewall.profile
```

   이 예제는 전용 방화벽을 개발하기 좋은 기준선이므로 전용 웹 서버 구성을 기준으로 합니다.

5. 프로파일 및 드라이버 파일을 작성한 후 다음과 같이 파일을 수정합니다.

   a. `hardening.driver`에 대한 `xsp-firewall-secure.driver` 참조를 `xsp-firewall-hardening.driver`로 교체합니다.

b. JASS_SCRIPTS에 정의된 두 종료 스크립트를 `minimize-firewall.fin`에 대한 참조 및 사용자의 종료 스크립트(예: `fw1-patch-install.fin`)로 교체합니다.

수정된 스크립트는 다음과 유사해야 합니다.

코드 예 **7-3** 수정된 스크립트의 출력 예제

```
DIR="'/bin/dirname $0'"
export DIR
. ${DIR}/driver.init
. ${DIR}/config.driver
JASS_SCRIPTS="
                minimize-firewall.fin
                fw1-patch-install.fin"
. ${DIR}/driver.run
. ${DIR}/xsp-firewall-hardening.driver
```

6. 다음 명령을 사용하여 `rules` 파일 항목이 올바른지 확인하십시오.

코드 예 **7-4** `rules` 파일의 정확성 검사

```
# pwd
/jumpstart
# ./check
Validating rules...
Validating profile Profiles/end-user.profile...
Validating profile Profiles/xsp-minimal-firewall.profile...
Validating profile Profiles/test.profile...
Validating profile Profiles/entire-distribution.profile...
Validating profile Profiles/oem.profile...
The custom JumpStart configuration is ok.
```

이 시점에서, 이 예제의 `jordan` 클라이언트에서 JumpStart 설치를 시작하는 것이 가능해야 합니다. 작성한 JumpStart 구성과 Solaris Security Toolkit 드라이버, 종료 스크립트 및 프로파일을 사용합니다.

7. `rules` 파일을 검사할 때 문제점이 발생하는 경우 **98**페이지의 **"Rules** 파일 확인 및 검사**"**를 참조하십시오.

8. 클라이언트의 ok 프롬프트에서 다음 명령을 입력하여 **JumpStart** 기반구조를 사용하는 클라이언트를 설치합니다.

```
ok> boot net - install
```

클라이언트가 설치되지 않는 경우 구성을 검토하여 적절히 동작할 때까지 구성을 수정하십시오. 이 절에서 JumpStart 구성의 모든 부분이 언급되지는 않습니다. 자세한 내용은 Sun BluePrint 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

rules 파일의 올바른 실행을 달성하고 패치가 제대로 설치되었음을 확인한 다음, 클라이언트 시스템의 기본 레벨 설치 및 그의 최소화 및 강화를 시작할 수 있습니다.

Rules 파일 확인 및 검사

rules 파일의 정확성을 확인할 때, 다양한 문제점이 발생할 수 있습니다. 일반적인 문제 몇 가지가 이 절에서 언급됩니다.

rules 파일을 처음 실행하면 다음 출력이 나타납니다.

코드 예 **7-5** rules 파일의 예제 출력

```
# pwd
/jumpstart
# ./check
Validating rules...
Validating profile Profiles/xsp-minimal-firewall.profile...
Error in file "rules", line 20
hostname jordan - Profiles/xsp-minimal-firewall.profile
Drivers/xsp-firewall-secure.driver
오류: Profile missing:
   Profiles/xsp-minimal-firewall.profile
```

이 예제에서, jordan에 대한 rules 항목에 지정된 프로파일이 존재하지 않습니다. xsp-minimal-firewall.profile 프로파일은 profiles 디렉토리에 존재하지 않았습니다. 일반적으로 이 오류는 파일 이름의 잘못된 철자, 프로파일의 올바른 디렉토리 지정 생략 또는 프로파일을 작성하지 않은 이유로 발생됩니다. 문제점을 수정하고 검사를 재실행하십시오.

두 번째 실행은 다른 두 개의 문제점을 보입니다. 첫 번째 문제점은 xsp-firewall-secure.driver에서 호출되는 드라이버입니다. xsp-firewall-hardening.driver를 호출하는 대신, xsp-firewall-secure.driver가 여전히 hardening.driver를 호출하고 있습니다.

두 번째 문제점은 JASS_SCRIPTS 변수가 minimize-firewall.fin 대신에 minimize-Sun_ONE-WS.fin으로 잘못 설정된다는 것입니다.

다음은 잘못된 스크립트입니다.

코드 예 **7-6** 잘못된 스크립트의 예제

```
#!/bin/sh
DIR="`/bin/dirname $0`"
export DIR
. ${DIR}/driver.init
. ${DIR}/config.driver
JASS_SCRIPTS="minimize-Sun_ONE-WS.fin"
. ${DIR}/driver.run
. ${DIR}/hardening.driver
```

다음은 올바른 스크립트의 예제입니다.

코드 예 **7-7** 올바른 스크립트의 예제

```
#!/bin/sh
DIR="`/bin/dirname $0`"
export DIR
. ${DIR}/driver.init
. ${DIR}/config.driver
JASS_SCRIPTS="
minimize-firewall.fin"
. ${DIR}/driver.run
. ${DIR}/xsp-firewall-hardening.driver
```

강화 구성 사용자 정의

제안된 방화벽의 강화 구성은 사용자 정의되고 정교하게 조절될 준비가 되었습니다. 초기 스크립트는 hardening.driver를 기초로 합니다. 이는 시스템이 모든 서비스를 사용할 수 없는 "warm-brick" 상태임을 의미합니다.

Solaris 8 OS에 Secure Shell 클라이언트가 포함되어 있지 않으므로, 방화벽의 원격 네트워크 기반 관리가 가능하도록 수정해야 합니다. 이 사례 시나리오의 방화벽의 경우, FTP 서비스가 사용 가능 상태로 남아있고 원격 관리를 위해 Secure Shell 클라이언트를 설치해야 합니다. 이들 서비스를 모두 개인 관리 네트워크만으로 제한함으로써 다른 네

트워크 인터페이스에서 접속하지 못하도록 하십시오. 이러한 서비스 제한에 대한 정보에 대해서는, Sun BluePrints OnLine 기사 "Solaris Operating Environment Security: Updated for the Solaris 9 Operating Environment"를 참조하십시오.

이들 두 서비스를 사용 가능한 상태로 두는 것 외에, 디스크 미러링을 위해 Solstice DiskSuite를 구성하는 데 Solstice DiskSuite 그래픽 사용자 인터페이스(GUI)를 사용할 수 있도록 RPC 서비스를 사용 가능한 상태로 두십시오. Solstice DiskSuite GUI를 사용하지 않을 경우, RPC 서비스는 필요하지 않습니다. 이 예제에서는 GUI가 필요하므로 RPC 서비스는 사용 가능한 상태로 남아있습니다. Solstice DiskSuite의 설치 및 구성은 이 설명서에서 다루지 않습니다.

이 클라이언트에 필요한 최종 수정은 xSP(서비스 제공자)의 중앙 집중된 SYSLOG 서버를 사용하는 사용자 정의된 `syslog.conf`가 정교하게 만들어지는 것입니다. 이 사용자 정의된 `syslog.conf` 파일이 각 방화벽 시스템에 설치되어야 합니다.

이 수정을 위해서는 Solaris Security Toolkit의 구성 옵션을 변경해야 합니다. 필요한 각 수정사항이 다음 절에서 상세히 설명됩니다.

- 100페이지의 "FTP 서비스 사용"
- 101페이지의 "Secure Shell 소프트웨어 설치"
- 102페이지의 "RPC 서버 사용"
- 103페이지의 "`syslog.conf` 파일 사용자 정의"

FTP 서비스 사용

이 사례 시나리오의 방화벽의 경우 FTP 서비스를 사용 가능하게 하십시오.

▼ FTP 서비스 사용

1. **FTP**를 사용 가능한 상태로 두기 위해, `JASS_SVCS_DISABLE` 및 `JASS_SVCS_ENABLE` 변수를 설정하여 `update-inetd-conf.fin` 파일의 기본 동작을 수정합니다.

   FTP를 제외한 모든 표준 Solaris OS 서비스를 사용 불가능하게 하려면, 본 사례 시나리오에 대한 최선의 방법은 `JASS_SVCS_DISABLE`이 `finish.init` 스크립트에서 받은 기본값으로 남아있는지 확인하면서 `JASS_SVCS_ENABLE`이 `ftp`가 되도록 정의하는 것이다. *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.

2. 환경 변수를 통해 변경을 구현하기 위해 `xsp-firewall-hardening.driver`에 대한 호출 전에 다음과 비슷한 항목을 `xsp-firewall-secure.driver`에 추가합니다.

```
JASS_SVCS_ENABLE="ftp"
```

3. **FTP**가 방화벽 소프트웨어를 통해 실행함으로써 시스템 관리자의 관리 네트워크에서만 사용 가능한지 확인합니다.

   또 다른 요구사항은 FTP가 시스템 관리자의 관리 네트워크에서만 사용 가능한지 여부입니다. Solaris 8 OS에서 TCP 랩퍼를 시스템에 통합함으로써 또는 방화벽 소프트웨어 자체를 통해서 이 요구사항을 수행할 수 있습니다. 이 사례 시나리오에서는 방화벽 소프트웨어를 통해서 수행하십시오.

Secure Shell 소프트웨어 설치

Solaris 8 OS에는 Secure Shell 클라이언트가 들어있지 않으므로, 원격 관리를 위해 Secure Shell 클라이언트를 설치하십시오.

Solaris Security Toolkit 소프트웨어를 구성하여 OpenSSH 도구를 설치할 수 있습니다. `xsp-firewall-secure.driver`에 의해 사용된 `config.driver` 파일에 나열되는 `install-openssh.fin` 스크립트를 사용하십시오.

▼ Secure Shell 설치

1. 기본 `config.driver`를 `xsp-firewall-config.driver`에 복사합니다.
2. 파일 사본에서 `install-openssh.fin`에 대한 항목을 주석으로 처리합니다.
3. `xsp-firewall-config.driver`에서 `config.driver`를 호출하는 항목을 수정하여 대신 `xsp-firewall-secure.driver`를 호출합니다.
4. **OpenSSH**의 최신 버전을 구합니다.

   패치 및 OS 릴리스의 경우와 마찬가지로 OpenSSH의 최신 버전을 사용하십시오. 최신 릴리스 정보는 다음 OpenSSH 웹 페이지를 참조하십시오.

   `http://www.openssh.org`
5. **OpenSSH** 패키지를 컴파일하고, 적절하게 이름을 지정한 후 `Packages`디렉토리에 설치합니다.

   이 패키지에 대한 자세한 정보는, Sun BluePrints 온라인 기사 "Configuring OpenSSH for the Solaris Operating Environment"를 참조하십시오.

6. 올바른 **OpenSSH** 패키지 이름을 반영하도록 `install-openssh.fin`스크립트를 업데이트합니다.

   `install-openssh.fin` 스크립트를 수정해야 하는 경우가 있습니다. 이 스크립트는 OpenSSH 패키지의 패키지 이름을 다음과 유사하게 형식화되도록 정의합니다.

```
OBSDssh-3.5p1-sparc-sun4u-5.8.pkg
```

   여기서 패키지 이름은 버전 번호(`3.5p1`), 구조(`sparc`), 구조의 버전(`sun4u`), 패키지가 컴파일되는 대상 OS(`5.8`) 및 `pkg` 접미어의 형식을 따릅니다.

7. **FTP**가 방화벽 소프트웨어를 통해 실행함으로써 시스템 관리자의 관리 네트워크에서만 사용 가능한지 확인합니다.

   또 다른 요구사항은 FTP가 시스템 관리자의 관리 네트워크에서만 사용 가능한지 여부입니다. Solaris 8 OS에서 TCP 랩퍼를 시스템에 통합함으로써 또는 방화벽 소프트웨어 자체를 통해 이 요구사항을 구현할 수 있습니다. 이 사례 시나리오의 경우, 방화벽 소프트웨어를 통해 구현합니다. 또한 Secure Shell 서버의 구성을 수정하여 구현할 수도 있습니다.

RPC 서버 사용

RPC가 필요한 디스크 미러링을 위해 SDS를 사용할 수 있도록 RPC 서비스를 사용 가능한 상태로 두십시오.

이 수정은 특정 종료 스크립트인 `disable-rpc.fin`가 Solaris Security Toolkit 실행 중에 RPC 서비스를 사용 불가능하게 할 수 있기 때문에 비교적 간단합니다.

주 – 시스템의 RPC 서비스에 원격으로 액세스하는 것은 시스템 방화벽 구성에 의해 명백히 거부되어야 합니다.

▼ RPC 사용

- `xsp-firewall-hardening.driver`에서 `disable-rpc.fin`에 대한 항목을 주석으로 처리합니다.

  스크립트를 제거하는 대신 주석으로 처리하여 드라이버에서 스크립트를 사용할 수 없도록 합니다. 주석 값의 특정 조합만이 허용되기 때문에 `JASS_SCRIPTS` 정의의 항목을 주석화할 때 주의하십시오.

  다음은 Solaris Security Toolkit 소프트웨어가 `JASS_SCRIPTS` 정의에서 주석 표시기로서 허용하는 것에 대해 `driver.funcs script`에 들어있는 주석입니다.

```
#Very rudimentary comment handler. This code will only recognize
#comments where a single '#' is placed before the file name
#(separated by white space or not). It then will only skip the
#very next argument.
```

`syslog.conf` 파일 사용자 정의

이 클라이언트에 필요한 최종 수정은 xSP(서비스 제공자)의 중앙 집중된 SYSLOG 서버를 사용하는 사용자 정의된 `syslog.conf`가 정교하게 만들어지는 것입니다. 이 사용자 정의된 `syslog.conf` 파일이 각 방화벽 시스템에 설치되어야 합니다.

▼ `syslog.conf` 파일 사용자 정의

1. **xSP** 표준 `syslog.conf` 파일을 복사한 후 `syslog.conf.jordan`로 이름을 바꾸고 `Files/etc` 디렉토리에 저장합니다.

   Solaris Security Toolkit 소프트웨어는 다양한 파일 복사 모드를 지원합니다. 이 구성에 가장 적합한 옵션은 `syslog.conf` 파일이 `jordan`에만 복사되도록 시스템의 호스트 이름을 파일에 접미어로 추가하는 것으로, 그것이 고유한 방화벽 특성 수정사항을 갖기 때문입니다. 이 경우, 클라이언트를 `jordan`이라고 하므로 `Files/etc`에 사용된 실제 파일 이름은 `syslog.conf.jordan`입니다. `JASS_FILES` 정의는 이 접미어가 추가되지 않아야 한다는 것을 주의하십시오. 접미어에 대한 자세한 정보는, *Solaris Security Toolkit 4.1 Reference Manual*을 참조하십시오.

2. **xSP** 표준 `syslog.conf` 파일을 사용할 수 없는 경우 다음과 같이 사용자 정의 `syslog.conf` 파일을 작성합니다.

   a. **Solaris Security Toolkit** 소프트웨어에 포함된 `syslog.conf` 파일을 복사하여 `syslog.conf.jordan`으로 이름을 바꾼 다음 `Files/etc` 디렉토리에 저장합니다.

   b. `SYSLOG`에 대해 **xSP** 표준을 준수하도록 `syslog.conf.jordan`을 수정합니다.

3. `/etc/syslog.conf` 파일이 `xsp-firewall-hardening.driver`의 `JASS_FILES` 정의에 나열되는지 확인합니다.

   기본적으로 `xsp-firewall-hardening.driver`의 수정된 `JASS_FILE` 정의는 다음과 같이 나타납니다.

   코드 예 **7-8** 수정된 `xsp-firewall-hardening.driver`의 출력 예제

```
JASS_FILES="
                    /etc/dt/config/Xaccess
                    /etc/init.d/inetsvc
                    /etc/init.d/nddconfig
                    /etc/init.d/set-tmp-permissions
                    /etc/issue
                    /etc/motd
                    /etc/notrouter
                    /etc/rc2.d/S00set-tmp-permissions
                    /etc/rc2.d/S07set-tmp-permissions
                    /etc/rc2.d/S70nddconfig
                    /etc/syslog.conf
"
```

   이제 모든 필수 수정사항이 만들어졌습니다. OS의 설치, 최소화 및 강화는 특정 응용 프로그램에 맞게 사용자 정의되고 완전히 자동화되었습니다. 완전히 자동화되지 않은 유일한 프로세스는 방화벽 소프트웨어와 Solstice DiskSuite의 구성 및 설치입니다. JumpStart 기술을 사용하여 이러한 구성을 수행할 수도 있지만, 이 책에서는 다루지 않습니다. Sun BluePrints 설명서 *JumpStart Technology: Effective Use in the Solaris Operating Environment*를 참조하십시오.

클라이언트 설치

드라이버에 대한 모든 수정사항을 작성한 뒤 이 절에서 설명한 것처럼 클라이언트를 설치하십시오.

▼ 클라이언트 설치

1. 드라이버에 대한 모든 필수 수정사항을 작성한 후, **JumpStart** 기반구조를 사용하여 클라이언트를 설치합니다.

   클라이언트의 `ok` 프롬프트에서 다음 명령을 사용합니다.

```
ok> boot net - install
```

2. 오류가 발생할 경우, 오류를 수정하고 클라이언트 **OS**를 재설치합니다.

품질 보증 검사

프로세스의 마지막 작업은 시스템이 제공하는 응용 프로그램 및 서비스가 올바르게 기능하고 있는지 확인하는 것입니다. 또한, 이 작업은 보안 프로파일이 필수 수정사항을 성공적으로 이행하고 있는지 확인합니다.

모든 이상 징후나 문제점이 감지되고 신속히 정정되도록 하려면, 강화 및 최소화된 플랫폼을 재부팅한 다음 이 작업을 신속하고 꼼꼼하게 수행해야 합니다. 이 프로세스는 프로파일 설치 확인 및 응용 프로그램과 서비스 기능성 확인의 두 작업으로 구분됩니다.

▼ 프로파일 설치 확인

Solaris Security Toolkit 소프트웨어가 보안 프로파일을 오류 없이 올바르게 설치했는지 확인하기 위해 다음을 검토하고 평가하십시오.

1. 설치 로그 파일을 검토합니다.

   이 파일은 `JASS_REPOSITORY/jass-install-log.txt`에 설치됩니다.

주 – 이 로그 파일은 Solaris Security Toolkit 소프트웨어가 시스템에 수행한 작업을 이해하기 위한 참조로서 사용될 수 있습니다. 시스템의 각 실행에 대해 실행 시작 시간을 기초로 새 로그 파일이 디렉토리에 저장됩니다. 이들 파일과 JASS_REPOSITORY 디렉토리의 다른 모든 파일은 절대 직접 수정해서는 안됩니다.

2. 감사 옵션을 사용하여 시스템의 보안 구성을 평가합니다.

감사 옵션에 대한 자세한 정보는 6 장을 참조하십시오. 이 시나리오의 경우, 클라이언트에서 Solaris Security Toolkit 소프트웨어가 설치된 디렉토리에서 다음 명령을 사용합니다.

코드 예 **7-9** 보안 구성 평가

```
# ./jass-execute -a xsp-firewall-secure.driver
[NOTE] Executing driver, xsp-firewall-secure.driver
================================================================
===
xsp-firewall-secure.driver: Driver started.
================================================================
===

================================================================
===
Solaris Security Toolkit Version:   4.1.0
[...]
```

Solaris Security Toolkit 검증 실행시 불일치가 발견되면, 해당 불일치는 기록됩니다. 실행 완료시 발견된 총 불일치 수가 요약에서 보고됩니다. 실행의 전체 출력은 JASS_REPOSITORY 디렉토리에 있습니다.

▼ 응용 프로그램 및 서비스 기능 확인

응용 프로그램 및 서비스에 대한 확인 프로세스에는 잘 정의된 테스트 및 허용 계획의 실행이 포함됩니다. 이 계획은 시스템 또는 응용 프로그램의 다양한 구성요소를 조사하여 구성요소가 사용 가능한 상태 및 작업 명령 상태에 있는지 판별하는데 사용됩니다. 이러한 계획이 사용 불가능할 경우, 시스템의 사용 방법을 기초로 시스템을 합리적으로 검사하십시오. 강화 프로세스는 해당 기능을 수행하기 위해 응용 프로그램 및 서비스 기능에 영향을 미치지 않음을 확인하는 것이 중요합니다.

1. 시스템이 강화된 후 응용 프로그램이나 서비스가 제대로 작동하지 않음을 발견하는 경우 **2** 장에 설명된 기법을 사용하여 문제점을 판별하십시오.

예를 들어, truss 명령을 사용하십시오. 이 명령은 응용 프로그램에 문제가 발생하는 지점을 결정하는데 사용될 수 있습니다. 이 지점을 찾으면 이 문제에 대해 Solaris Security Toolkit 소프트웨어로 작업한 변경사항을 다시 추적할 수 있습니다.

주 – Solaris Security Toolkit 소프트웨어를 전개한 여러 사용자의 경험을 토대로 이 설명서에 나오는 접근 방식을 사용하면 대다수의 문제점을 해결할 수 있습니다.

2. 유사한 방법으로 **Check PointFirewall-1 NG** 소프트웨어를 검사하고**, Solaris Security Toolkit** 소프트웨어 수정으로 인한 모든 문제점을 다시 추적하고**,** 문제를 정정합니다**.**
3. 패키지의 최종 목록에 수정이 필요한 경우**,** 프로파일을 수정하고 시스템을 재설치한 후 검사를 반복하십시오**.**

용어집

본 목록은 Solaris Security Toolkit에 나오는 축약어 및 두문자어를 정의합니다.

A

ab2 AnswerBook2

ABI 응용 프로그램 이진 인터페이스

ARP 주소 결정 프로토콜

ASPPP 비동기식 점대점 프로토콜

B

BIND 버클리 인터넷 네임 도메인

BSD 버클리 소프트웨어 분배

BSM 기본 보안 모델*(Solaris)*

C

CD 컴팩트 디스크

CD-ROM 컴팩트 디스크-읽기 전용 메모리

CDE 공통 데스크탑 환경

cp(1) 복사 파일

cron(1M) 클록 데몬

D

DHCP 동적 호스트 구성 프로토콜

DMI 데스크탑 관리 인터페이스

DMTF 분산 관리 작업 단체

DNS 도메인 네임 시스템

E

EEPROM 전자식 소거 및 프로그램 가능 읽기용 메모리

F

FTP 파일 전송 프로토콜

G

GID 그룹 식별자

H

HTTP 하이퍼텍스트 전송 프로토콜

I

ID ID

IETF 인터넷 엔지니어링 태스크 포스

INETD 인터넷 서비스 데몬

IP 인터넷 프로토콜

ISA 명령어 집합 아키텍처

J

JASS JumpStart 구조 및 보안 스크립트, 현재의 경우 Solaris Security Toolkit

K

KDC 커베로스 키 분배

L

LDAP 경량 디렉토리 액세스 프로토콜

lp(1) 프린터 입력 *(*프린트 요청 제출*)*

M

MAN 네트워크 관리 *(Sun Fire High-End Systems* 내부 *I1* 네트워크*)*

MD5 메시지 요약 5 알고리즘

MIP 이동식 인터넷 프로토콜

MSP 미드프레임 서비스 처리 장치

mv(1) 파일 이동

N

NFS 네트워크 파일 시스템

NG 차세대

NIS, NIS+ 네트워크 정보 서비스

NSCD 이름 서비스 캐쉬 데몬

O

OE 운영 환경, 이전에 *Solaris*용으로 사용됨

OEM 주문자 상표 부착 생산

OS 운영 시스템, 현재 *Solaris*용으로 사용됨

P

PAM 플러그 가능한 인증 모듈

PDF 휴대용 문서 형식

PICL 플랫폼 정보 및 제어 라이브러리

PPP 점대점 프로토콜

PROM 프로그램 가능 읽기용 메모리

Q

QA 품질 보증

R

RBAC 역할 기반 액세스 제어

rc 실행 제어 *(*파일 또는 스크립트*)*

rlogin(1) 원격 로그인

RFC 원격 기능 호출

RPC 원격 절차 호출

rsh(1) 원격 쉘

S

SA 시스템 관리자

SC 시스템 제어 *(Sun Fire* 고급 및 중급 시스템*)*

scp(1) 보안 복사(원격 파일 복사 프로그램)

SCCS 원시 코드 제어 시스템

SLP 서비스 지역 프로토콜

SMA 시스템 관리 에이전트

SMC Solaris 관리 콘솔

SNMP 간이 망 관리 프로토콜

SP 서비스 제공자

SPARC 스파크

SPC SunSoft Print Client

SSH 보안 쉘 *(Solaris)*

SSP 시스템 서비스 프로세서 *(Sun Enterprise 10000* 서버*)*

stdio 표준 입/출력

Sun ONE Sun Open Network Environment, 현재의 경우 Sun Java System, 이전의 경우 iPlanet

T

TCP 전송 제어 프로토콜

tftp(1) 단순 파일 전송 프로그램

ttl time-to-live

U

U.S. 미국

UDP 사용자 다이어그램 프로토콜

UID 사용자 식별자

UUCP UNIX 시스템간 복사

V

VOLD 볼륨 관리 데몬

W

WBEM 웹 기반 기업 관리

색인

I

J

K

L

M

N

O

P

Q

R

S

T

U

W

Z

ㄱ