

# Oracle® Solaris 11 소프트웨어 패키지 추가 및 업데이트

본 소프트웨어와 관련 문서는 사용 제한 및 기밀 유지 규정을 포함하는 라이선스 계약서에 의거해 제공되며, 지적 재산법에 의해 보호됩니다. 라이선스 계약서 상에 명시적으로 허용되어 있는 경우나 법규에 의해 허용된 경우를 제외하고, 어떠한 부분도 복사, 재생, 번역, 방송, 수정, 라이선스, 전송, 배포, 진열, 실행, 발행, 또는 전시될 수 없습니다. 본 소프트웨어 리버스 엔지니어링, 디스어셈블리 또는 디컴파일하는 것은 상호 운용에 대한 법규에 의해 명시된 경우를 제외하고는 금지되어 있습니다.

이 안의 내용은 사전 공지 없이 변경될 수 있으며 오류가 존재하지 않음을 보증하지 않습니다. 만일 오류를 발견하면 서면으로 통지해 주시기 바랍니다.

만일 본 소프트웨어나 관련 문서를 미국 정부나 또는 미국 정부를 대신하여 라이선스한 개인이나 법인에게 배송하는 경우, 다음 공지 사항이 적용됩니다.

#### U.S. GOVERNMENT END USERS:

Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

본 소프트웨어 혹은 하드웨어는 다양한 정보 관리 애플리케이션의 일반적인 사용을 목적으로 개발되었습니다. 본 소프트웨어 혹은 하드웨어는 개인적인 상해를 초래할 수 있는 애플리케이션을 포함한 본질적으로 위험한 애플리케이션에서 사용할 목적으로 개발되거나 그 용도로 사용될 수 없습니다. 만일 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서 사용할 경우, 라이선스 사용자는 해당 애플리케이션의 안전한 사용을 위해 모든 적절한 비상-안전, 백업, 대비 및 기타 조치를 반드시 취해야 합니다. Oracle Corporation과 그 회사는 본 소프트웨어 혹은 하드웨어를 위험한 애플리케이션에서의 사용으로 인해 발생하는 어떠한 손해에 대해서도 책임지지 않습니다.

Oracle과 Java는 Oracle Corporation 및/또는 그 자회사의 등록 상표입니다. 기타의 명칭들은 각 해당 명칭을 소유한 회사의 상표일 수 있습니다.

Intel 및 Intel Xeon은 Intel Corporation의 상표 내지는 등록 상표입니다. SPARC 상표 일체는 라이선스에 의거하여 사용되며 SPARC International, Inc.의 상표 내지는 등록 상표입니다. AMD, Opteron, AMD 로고, 및 AMD Opteron 로고는 Advanced Micro Devices의 상표 내지는 등록 상표입니다. UNIX는 The Open Group의 등록 상표입니다.

본 소프트웨어 혹은 하드웨어와 관련 문서(설명서)는 제 3자로부터 제공되는 콘텐츠, 제품 및 서비스에 접속할 수 있거나 정보를 제공합니다. Oracle Corporation과 그 자회사는 제 3자의 콘텐츠, 제품 및 서비스와 관련하여 어떠한 책임도 지지 않으며 명시적으로 모든 보증에 대해서도 책임을 지지 않습니다. Oracle Corporation과 그 자회사는 제 3자의 콘텐츠, 제품 및 서비스에 접속하거나 사용으로 인해 초래되는 어떠한 손실, 비용 또는 손해에 대해 어떠한 책임도 지지 않습니다.

# 목차

---

머리말 .....	7
<b>1 이미지 패키징 시스템 소개 .....</b>	<b>11</b>
이미지 패키징 시스템 .....	11
설치 권한 .....	12
IPS 개념 .....	12
IPS 패키지 .....	12
FMRI(Fault Management Resource Identifier) .....	13
게시자, 저장소 및 패키지 아카이브 .....	14
저장소 원본 및 미러 .....	14
이미지 및 부트 환경 .....	14
패키지 페이지 및 변형 .....	15
<b>2 IPS 그래픽 사용자 인터페이스 .....</b>	<b>17</b>
패키지 관리자 사용 .....	17
패키지 관리자 명령줄 옵션 .....	18
웹 설치 사용 .....	18
업데이트 관리자 사용 .....	20
업데이트 관리자 명령줄 옵션 .....	21
<b>3 소프트웨어 패키지 정보 보기 .....</b>	<b>23</b>
패키지 설치 상태 정보 표시 .....	23
패키지 설명 또는 라이선스 표시 .....	25
패키지 매니페스트의 정보 표시 .....	27
패키지를 통해 설치된 파일 나열 .....	27
그룹 패키지의 설치 가능한 모든 패키지 나열 .....	28
라이선스 요구 사항 표시 .....	28

패키지 검색 .....	29
특정 파일을 제공하는 패키지 식별 .....	29
범주별로 패키지 나열 .....	29
종속 항목 패키지 표시 .....	30
그룹 패키지의 모든 패키지 나열 .....	31
<b>4 소프트웨어 패키지 설치 및 업데이트 .....</b>	<b>33</b>
작업 미리 보기 .....	33
패키지 설치 및 업데이트 .....	35
부트 환경 옵션 .....	35
새 패키지 설치 .....	36
새 부트 환경에 패키지 설치 .....	37
패키지 거부 .....	39
패키지 업데이트 .....	39
패키지 문제 수정 .....	40
패키지 설치 검증 .....	40
검증 오류 수정 .....	41
파일 복원 .....	41
패키지 제거 .....	42
<b>5 설치된 이미지 구성 .....</b>	<b>43</b>
게시자 구성 .....	43
게시자 정보 표시 .....	43
패키지 게시자 추가, 수정 또는 제거 .....	44
지정한 버전으로 패키지 잠금 .....	46
패키지를 무시할 패키지로 표시 .....	47
선택적 구성 요소의 설치 제어 .....	47
변형 표시 .....	49
변형 변경 .....	49
페이셋 표시 .....	50
페이셋 변경 .....	50
이미지 업데이트 .....	51
이미지 및 게시자 등록 정보 구성 .....	52
부트 환경 정책 이미지 등록 정보 .....	52
패키지 서명에 필요한 등록 정보 .....	54

추가 이미지 등록 정보 .....	56
이미지 등록 정보 설정 .....	57
이미지 만들기 .....	58
작업 기록 보기 .....	59



# 머리말

---

**Oracle Solaris 11 소프트웨어 패키지 추가 및 업데이트**에서는 Oracle Solaris IPS(이미지 패키징 시스템)의 소프트웨어 설치 기능에 대해 설명합니다. IPS 명령을 사용하여 Oracle Solaris 11 운영 체제용 소프트웨어 패키지를 나열, 검색, 설치, 업데이트 및 제거할 수 있습니다. 단일 IPS 명령을 사용하여 이미지를 새 운영 체제 릴리스로 업데이트할 수 있습니다. IPS 명령을 사용하면 설치할 수 있는 패키지 또는 패키지 버전을 제한할 수 있습니다.

또한 IPS 명령을 사용하면 IPS 패키지 저장소를 복사하거나 만들고 IPS 패키지를 만들 수 있습니다. 이러한 도구에 대한 자세한 내용은 [8 페이지 “관련 문서”](#)를 참조하십시오.

IPS를 사용하려면 Oracle Solaris 11 OS를 실행해야 합니다. Oracle Solaris 11 OS를 설치하려면 [Oracle Solaris 11 시스템](#)를 참조하십시오.

## 이 책의 대상

이 책은 소프트웨어를 설치 및 관리하고 시스템 이미지를 관리하는 시스템 관리자를 대상으로 합니다.

## 이 책의 구성

- 1 장, “[이미지 패키징 시스템 소개](#)”에서는 이미지 패키징 시스템과 패키지, 게시자, 저장소 등의 구성 요소에 대해 설명합니다.
- 2 장, “[IPS 그래픽 사용자 인터페이스](#)”에서는 패키지 관리자 및 업데이트 관리자 사용 방법과 웹 설치 사용 방법에 대해 설명합니다.
- 3 장, “[소프트웨어 패키지 정보 보기](#)”에서는 패키지를 검색하고 패키지에 대한 정보를 표시하는 방법을 보여줍니다.
- 4 장, “[소프트웨어 패키지 설치 및 업데이트](#)”에서는 패키지 설치, 업데이트 및 제거 방법을 보여줍니다.
- 5 장, “[설치된 이미지 구성](#)”에서는 패키지 게시자 구성, 설치 가능한 패키지 제한 등 전체 이미지에 적용되는 특성을 구성하는 방법을 보여줍니다.

## 관련 문서

아래 책과 함께 패키지 관리자 온라인 도움말과 pkg(1M) 및 beadm(1M) 매뉴얼 페이지를 참조하십시오.

- 이미지 패키징 시스템 매뉴얼 페이지
- Oracle Solaris 11 패키지 저장소 복사 및 만들기
- Oracle Solaris 11 부트 환경 만들기 및 관리
- Oracle Solaris 11 시스템

## Oracle Support에 액세스

Oracle 고객은 My Oracle Support를 통해 온라인 지원에 액세스할 수 있습니다. 자세한 내용은 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>를 참조하거나, 청각 장애가 있는 경우 <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>를 방문하십시오.

## 활자체 규약

다음 표는 이 책에서 사용되는 활자체 규약에 대해 설명합니다.

표 P-1 활자체 규약

활자체	설명	예
AaBbCc123	명령 및 파일, 디렉토리 이름; 컴퓨터 화면에 출력되는 내용입니다.	.login 파일을 편집하십시오.  모든 파일 목록을 보려면 <code>ls -a</code> 명령을 사용하십시오.  <code>machine_name% you have mail.</code>
AaBbCc123	사용자가 입력하는 내용으로 컴퓨터 화면의 출력 내용과 대조됩니다.	<code>machine_name% su</code>  <code>Password:</code>
AaBbCc123	새로 나오는 용어, 강조 표시할 용어입니다. 명령줄 변수를 실제 이름이나 값으로 바꾸십시오.	<code>rm filename</code> 명령을 사용하여 파일을 제거합니다.

표 P-1 활자체 규약 (계속)

활자체	설명	예
AaBbCc123	책 제목, 장, 절	<p>사용자 설명서의 6장을 읽으십시오.</p> <p>캐시는 로컬로 저장된 복사본입니다.</p> <p>파일을 저장하면 <b>안 됩니다</b>.</p> <p>주: 일부 강조된 항목은 온라인에서 굵은체로 나타납니다.</p>

## 명령 예의 셸 프롬프트

다음 표에는 Oracle Solaris OS에 포함된 셸의 기본 UNIX 시스템 프롬프트 및 슈퍼유저 프롬프트가 나와 있습니다. 명령 예제에 표시된 기본 시스템 프롬프트는 Oracle Solaris 릴리스에 따라 다릅니다.

표 P-2 셸 프롬프트

셸	프롬프트
Bash 셸, Korn 셸 및 Bourne 셸	\$
슈퍼유저용 Bash 셸, Korn 셸 및 Bourne 셸	#
C 셸	machine_name%
슈퍼유저용 C 셸	machine_name#



# 이미지 패키징 시스템 소개

---

Oracle Solaris 이미지 패키징 시스템(IPS)은 Oracle Solaris 11 운영 체제용 소프트웨어 패키지를 나열, 검색, 설치, 업데이트 및 제거할 수 있도록 지원하는 프레임워크입니다. 단일 IPS 명령을 사용하여 이미지를 새 운영 체제 릴리스로 업데이트할 수 있습니다.

## 이미지 패키징 시스템

Oracle Solaris 11 소프트웨어는 IPS 패키지로 배포됩니다. IPS 패키지는 IPS 게시자가 채우는 IPS 패키지 저장소에 저장됩니다. IPS 패키지는 Oracle Solaris 11 이미지에 설치됩니다. IPS 명령줄 인터페이스를 통해 사용할 수 있는 기능 중 일부는 패키지 관리자 그래픽 사용자 인터페이스를 통해 사용할 수 있습니다.

IPS 도구가 제공하는 기능은 다음과 같습니다. 게시자, 저장소 등의 용어에 대한 정의는 [12 페이지 “IPS 개념”](#)을 참조하십시오.

- 소프트웨어 패키지를 나열, 검색, 설치, 업데이트 및 제거하고 설치를 제한합니다.
- 패키지 게시자를 나열, 추가 및 제거합니다. 검색 우선 순위, 고착성 같은 게시자 속성을 변경합니다. 서명 정책 같은 게시자 등록 정보를 설정합니다.
- 이미지를 새 운영 체제 릴리스로 업데이트합니다.
- 기존 IPS 패키지 저장소의 복사본을 만듭니다. 새 패키지 저장소를 만듭니다.
- 패키지를 만들고 게시합니다.
- 부트 환경을 만듭니다.

IPS를 사용하려면 Oracle Solaris 11 OS를 실행해야 합니다. Oracle Solaris 11 OS를 설치하려면 [Oracle Solaris 11 시스템](#)를 참조하십시오.

## 설치 권한

3 장, “소프트웨어 패키지 정보 보기”에 설명되어 있는 명령은 특별한 권한이 없어도 사용할 수 있습니다. IPS 패키지 설치 및 업데이트, 게시자 설정, 이미지 수정 등의 작업을 수행하려면 추가 권한이 필요합니다.

추가 권한을 얻으려면 다음 방법 중 하나를 사용하십시오.

- `profiles` 명령을 사용하여 할당 받은 권한 프로파일을 나열합니다. 소프트웨어 설치 권한 프로파일이 있으면 `pfexec` 명령을 사용하여 패키지를 설치하고 업데이트할 수 있습니다.

```
$ pfexec pkg install editor/gnu-emacs
```

시스템 관리자 권한 프로파일 등 다른 권한 프로파일도 설치 권한을 제공합니다.

- 사이트의 보안 정책에 따라 사용자 암호와 함께 `sudo` 명령을 사용하여 권한이 있는 명령을 실행할 수 있는 경우도 있습니다.

```
$ sudo pkg install editor/gnu-emacs
```

- `roles` 명령을 사용하여 할당 받은 역할을 나열합니다. 루트 역할이 있는 경우 루트 암호와 함께 `su` 명령을 사용하여 루트 역할을 가정할 수 있습니다.

## IPS 개념

이 절에서는 본 설명서의 나머지 부분에서 사용되는 용어와 개념을 정의합니다.

### IPS 패키지

IPS 패키지는 매니페스트라는 텍스트 파일로 정의할 수 있습니다. 패키지 매니페스트는 키/값 쌍과 데이터 페이로드라는 정의된 형식으로 패키지 작업에 대해 설명합니다.

패키지 작업에는 파일, 디렉토리, 링크, 드라이버, 종속성, 그룹, 사용자 및 라이선스 정보가 포함됩니다. 패키지 작업은 패키지의 설치 가능한 객체를 나타냅니다.

"set(설정)"이라는 작업은 분류, 요약, 설명 같은 패키지 메타데이터를 정의합니다.

패키지 작업과 작업 키를 지정하여 패키지를 검색할 수 있습니다. 패키지 작업에 대한 설명은 [pkg\(5\)](#)를 참조하십시오.

통합은 지정된 패키지 세트의 버전을 제한하는 패키지입니다. 예를 들어, 설치된 통합 내의 패키지가 버전 1.4.3이면 1.4.3 이전 버전이나 1.4.4 버전 또는 그 이후 버전은 설치할 수 없습니다. 그러나 1.4.3.7 같이 단순히 점 뒤에 숫자가 추가된 버전은 설치가 가능합니다. 통합은 통합 패키지가 동기적으로 업그레이드되도록 강제합니다. 통합 패키지는 제거는 가능하지만 패키지를 설치하거나 업데이트할 경우 버전이 제한됩니다.

그룹 패키지는 기능이나 도구를 구성하는 패키지 세트를 지정합니다. 그룹 패키지에 지정된 패키지에는 패키지 버전이 할당되지 않습니다. 그룹 패키지는 버전 관리 도구가 아니라 콘텐츠 관리 도구입니다.

## FMRI(Fault Management Resource Identifier)

각 패키지는 FMRI(Fault Management Resource Identifier)로 나타냅니다. 패키지의 전체 FMRI는 체계, 게시자, 패키지 이름 및 버전 문자열로 구성되며 형식은 다음과 같습니다. 체계, 게시자 및 버전 문자열은 선택 사항입니다. IPS 명령을 사용할 경우 패키지를 고유하게 식별하는 패키지 이름의 가장 작은 부분을 사용할 수 있습니다.

형식:

```
scheme://publisher/package_name@version:dateTtimeZ
```

예:

```
pkg://Solaris/editor/vim@7.3.254,5.11-0.174.0.0.0.0.504:20110921T002716Z
```

체계                    pkg

게시자                Solaris

게시자를 지정할 경우 게시자 이름 앞에 pkg:// 또는 //를 붙여야 합니다.

패키지 이름          editor/vim

패키지 이름 공간은 계층 구조로 되어 있으며 수준은 저마다 다릅니다. IPS 명령에서는 패키지를 고유하게 식별하는 패키지 이름의 가장 작은 부분을 지정할 수 있습니다. 전체 패키지 이름을 지정하되 게시자 이름을 생략할 경우 전체 패키지 이름 앞에 pkg:// 또는 //가 아니라 pkg:/ 또는 /를 붙여야 합니다. 패키지 이름을 약어로 지정하는 경우 패키지 이름 왼쪽에 다른 문자를 사용하지 마십시오.

버전                    패키지 버전은 네 부분으로 구성됩니다.

구성 요소 버전      7.3.254

운영 체제에 긴밀하게 바인딩된 구성 요소의 경우 일반적으로 해당 운영 체제 버전의 `uname -r` 값입니다.

빌드 버전            5.11

빌드 버전 앞에는 쉼표(.)가 나와야 합니다. 빌드 버전은 패키지 콘텐츠가 생성된 운영 체제의 버전을 지정합니다.

분기 버전            0.174.0.0.0.0.504

분기 버전 앞에는 하이픈(-)이 나와야 합니다. 분기 버전은 공급업체 관련 정보를 제공합니다.

## 시간 기록

20110921T002716Z

시간 기록 앞에는 콜론(:)이 나와야 합니다. 시간 기록은 패키지가 게시된 시간을 나타내며 ISO-8601 기본 형식(YYYYMMDDT HHMMSSZ)으로 표시됩니다.

## 게시자, 저장소 및 패키지 아카이브

**게시자**는 하나 이상의 패키지를 제공하는 개인이나 조직을 나타냅니다. 게시자는 패키지 저장소나 패키지 아카이브를 사용하여 패키지를 배포할 수 있습니다. 기본 검색 순서로 게시자를 구성할 수 있습니다. 패키지 사양에 게시자 이름을 포함하지 않고 패키지 설치 명령을 실행하면 해당 패키지에 대해 검색 순서의 첫번째 게시자가 검색됩니다. 패키지를 찾지 못하면 검색 순서의 두번째 게시자가 검색되고, 패키지를 찾거나 모든 게시자를 검색할 때까지 이런 식으로 다음 게시자가 검색됩니다.

**저장소**는 패키지가 게시되고 검색되는 위치입니다. 이 위치는 URI(Universal Resource Identifier)로 지정됩니다. **카탈로그**는 저장소에 있는 모든 패키지의 목록입니다.

**패키지 아카이브**는 게시자 정보와 해당 게시자가 제공한 하나 이상의 패키지가 포함된 파일입니다.

## 저장소 원본 및 미러

**원본**은 패키지 **메타 데이터**(예: 카탈로그, 매니페스트, 검색 색인)와 패키지 **컨텐츠**(파일)가 모두 포함된 패키지 저장소입니다. 이미지의 지정된 게시자에 대해 원본이 여러 개 구성된 경우 IPS 클라이언트는 패키지 데이터를 검색할 수 있는 최상의 원본을 선택하려고 시도합니다.

**미러**는 패키지 컨텐츠만 포함된 패키지 저장소입니다. IPS 클라이언트는 패키지 컨텐츠를 미러에서 다운로드하는 경우에도 게시자의 카탈로그를 구하기 위해 원본에 액세스합니다. 게시자에 대해 하나의 미러가 구성된 경우 IPS 클라이언트는 해당 미러를 패키지 컨텐츠 검색용으로 선택합니다. 이미지의 지정된 게시자에 대해 미러가 여러 개 구성된 경우 IPS 클라이언트는 패키지 컨텐츠를 검색할 수 있는 최상의 미러를 선택하려고 시도합니다. 모든 미러에 연결할 수 없거나 필요한 컨텐츠가 없거나 미러가 너무 느린 경우 IPS 클라이언트는 원본에서 컨텐츠를 검색합니다.

## 이미지 및 부트 환경

**이미지**는 IPS 패키지를 설치할 수 있으며 다른 IPS 작업을 수행할 수 있는 위치입니다.

BE(부트 환경)는 이미지의 부트 가능한 인스턴스입니다. 시스템에서 여러 BE를 유지 관리할 수 있으며 각 BE에 서로 다른 소프트웨어 버전을 설치할 수 있습니다. 시스템을 부트할 때 시스템에 있는 BE 중 하나로 부트하는 옵션이 제공됩니다. 패키지 작업의 결과로 새 BE가 자동으로 생성될 수 있습니다. 새 BE를 명시적으로 만들 수도 있습니다. 새 BE가 생성되는지 여부는 52 페이지 “부트 환경 정책 이미지 등록 정보”에 설명된 대로 이미지 정책에 따라 달라집니다.

## 패키지 페이지 및 변형

소프트웨어에는 선택적 구성 요소와 상호 배타적인 구성 요소가 포함될 수 있습니다. 선택적 구성 요소의 예로는 로켈과 설명서가 있습니다. 상호 배타적 구성 요소의 예로는 SPARC 또는 x86 및 디버그 또는 비디버그 이진이 있습니다. IPS에서는 선택적 구성 요소를 **페이지**이라 하고 상호 배타적 구성 요소를 **변형**이라고 합니다.

페이지와 변형은 특수한 이미지 등록 정보로 개별 패키지에 설정할 수 없습니다.

패키지 매니페스트의 개별 작업에 페이지 및 변형 태그를 지정할 수 있습니다. 단일 작업에 페이지 및 변형 태그를 여러 개 지정할 수 있습니다.

작업의 페이지 및 변형 태그 값을 이미지에 설정된 페이지 및 변형 값과 비교한 결과에 따라, 패키지 작업을 설치할 수 있는지 여부가 결정됩니다.

- 페이지 또는 변형 태그가 없는 작업은 항상 설치됩니다.
- 페이지 태그가 있는 작업은 태그와 일치하는 모든 페이지 또는 페이지 패턴이 이미지에 false로 설정되지 않는 한 설치됩니다. 어떤 페이지가 true로 설정되거나 명시적으로 설정되지 않은 경우(true가 기본값) 작업이 설치됩니다.
- 변형 태그가 있는 작업은 모든 변형 태그의 값이 이미지에 설정된 값과 동일한 경우에만 설치됩니다.
- 페이지 및 변형 태그가 모두 있는 작업은 페이지와 변형이 모두 작업 설치를 허용하는 경우 설치됩니다.

이미지에 설정된 페이지와 변형의 값을 보거나 수정하려면 47 페이지 “선택적 구성 요소의 설치 제어”를 참조하십시오.



## IPS 그래픽 사용자 인터페이스

---

IPS에는 두 개의 GUI(그래픽 사용자 인터페이스) 도구가 포함되어 있습니다.

- 패키지 관리자는 대부분의 패키지 및 게시자 작업과 몇몇 BE(부트 환경) 작업을 제공합니다. Oracle Solaris OS 및 IPS 기술을 처음 접하는 경우 패키지 관리자를 사용하여 신속하게 패키지를 식별하고 설치할 수 있습니다.
- 업데이트 관리자는 업데이트가 있는 이미지의 모든 패키지를 업데이트합니다.

### 패키지 관리자 사용

패키지 관리자는 명령줄에서 수행할 수 있는 작업 중 일부를 제공합니다.

- 패키지 나열, 검색, 설치, 업데이트, 제거
- 패키지 소스 추가 및 구성
- BE 활성화, 이름 바꾸기 및 제거

다음 방법 중 하나를 사용하여 패키지 관리자를 시작할 수 있습니다.

- |                 |   |
|-----------------|---|
| <b>도구 모음</b>    | 도구 모음에서 패키지 관리자 아이콘을 누릅니다. 패키지 관리자 아이콘은 원모양으로 회전하는 화살표가 있는 상자입니다. |
| <b>데스크탑 아이콘</b> | 데스크탑에서 패키지 관리자 아이콘을 두 번 누릅니다.                                     |
| <b>메뉴 표시줄</b>   | System(시스템)>Administration(관리)>Package Manager(패키지 관리자)를 선택합니다.   |
| <b>명령줄</b>      | <code># packagemanager</code>                                     |

전체 패키지 관리자 설명서를 보려면 패키지 관리자 메뉴 표시줄에서 Help(도움말)>Contents(목차)를 선택합니다.

## 패키지 관리자 명령줄 옵션

다음은 `packagemanager(1)` 명령에 지원되는 옵션입니다.

표 2-1 패키지 관리자 명령 옵션

옵션	설명
<code>--image-dir</code> 또는 <code>-R dir</code>	<i>dir</i> 에 루트 지정된 이미지에 대해 작업을 수행합니다. 기본 동작은 현재 이미지에 대해 작업을 수행하는 것입니다.  다음 명령은 <code>/aux0/example_root</code> 에 저장된 이미지에 대해 작업을 수행합니다.  <b># packagemanager -R /aux0/example_root</b>
<code>--update-all</code> 또는 <code>-U</code>	사용 가능한 업데이트가 있는 설치된 패키지를 모두 업데이트합니다. 이 옵션을 지정하는 것은 패키지 관리자 GUI에서 Updates(업데이트) 옵션을 선택하는 것과 같습니다. 모든 패키지 업데이트에 대한 자세한 내용은 20 페이지 “업데이트 관리자 사용”을 참조하십시오.
<code>--info-install</code> 또는 <code>-i file.p5i</code>	패키지 관리자를 웹 설치 모드로 실행하려면 <code>.p5i</code> 파일을 지정합니다. 지정된 파일은 확장자가 <code>.p5i</code> 여야 합니다. 자세한 내용은 18 페이지 “웹 설치 사용”을 참조하십시오.
<code>--help</code> 또는 <code>-h</code>	명령 사용 정보를 표시합니다.

## 웹 설치 사용

웹 설치 프로세스에 대한 자세한 내용은 패키지 관리자 도움말을 참조하십시오.

패키지 관리자는 한 번만 누르는 간단한 웹 설치 프로세스를 사용한 패키지 설치를 지원합니다. 웹 설치 프로세스에서는 `.p5i` 파일을 사용합니다. `.p5i` 파일에는 게시자 및 이러한 게시자에서 설치할 수 있는 패키지를 추가하는 데 필요한 정보가 들어 있습니다. `.p5i` 파일의 정보는 웹 설치 프로세스에서 읽고 사용됩니다.

## 웹 설치를 통해 파일 내보내기

시스템에 설치한 패키지를 다른 사용자도 설치할 수 있도록 하려는 경우 웹 설치 프로세스를 사용하여 해당 패키지 파일에 대한 설치 지침을 내보내면 됩니다. 웹 설치 프로세스에서는 설치할 패키지 및 게시자에 대한 설치 지침으로 이루어진 `.p5i` 파일을 만듭니다.

선택한 패키지 및 게시자의 설치 지침을 .p5i 파일로 내보내려면 다음 단계를 수행합니다.

1. Package Manager Publisher(패키지 관리자 게시자) 드롭다운 메뉴에서 패키지를 .p5i 파일에 포함시킬 게시자를 선택합니다.
2. 패키지 관리자 패키지 목록 창에서 설치 지침을 배포할 패키지를 선택합니다.
3. File(파일)>Export Selections(선택 항목 내보내기)를 선택하여 Export Selections Confirmation(선택 항목 내보내기 확인) 창을 표시합니다.
4. OK(확인) 버튼을 눌러 선택 항목을 확인합니다. Export Selections(선택 항목 내보내기) 창이 표시됩니다.
5. .p5i 파일의 기본 이름은 자동으로 제공됩니다. 이 파일의 이름은 변경할 수 있지만 .p5i 확장자는 변경하지 마십시오.
6. .p5i 파일의 기본 위치는 제공됩니다. 이 위치를 변경할 수 있습니다.
7. Save(저장) 버튼을 눌러 파일 이름과 위치를 저장합니다.

## 웹 설치를 사용하여 게시자 추가 및 패키지 설치

웹 설치 프로세스를 사용하면 .p5i 파일을 통해 패키지를 설치할 수 있습니다. 이 파일은 데스크탑이나 웹 사이트에 있을 수 있습니다.

1. 패키지 관리자를 웹 설치 모드로 시작하려면 다음 방법 중 하나를 사용합니다.
  - 데스크탑에서 .p5i 파일을 선택합니다.
  - 명령줄에서 .p5i 파일을 지정하여 패키지 관리자를 시작합니다.
 

```
# packagemanager ./wifile.p5i
```

    - .p5i 파일에 대한 링크가 있는 URL 위치로 이동합니다.
      - .p5i 파일이 이 MIME 형식이 등록된 웹 서버에 있으면 .p5i 파일에 대한 링크만 누르면 됩니다.
      - .p5i 파일이 이 MIME 형식이 등록되지 않은 웹 서버에 있으면 .p5i 파일을 데스크탑에 저장한 후 선택합니다.
2. Install/Update(설치/업데이트) 창이 표시됩니다. 창 위쪽에 "Package Manager Web Installer/The following will be added to your system.(패키지 관리자 웹 설치 프로그램/다음의 내용들이 여러분의 시스템에 추가됩니다.)"라는 레이블이 표시됩니다. 설치할 게시자 및 패키지가 나열됩니다. Proceed(진행) 버튼을 눌러 설치를 계속합니다.
3. 지정한 패키지 게시자가 시스템에 아직 구성되어 있지 않으면 Add Publisher(게시자 추가) 창이 표시됩니다. 게시자의 이름과 URI는 이미 입력되어 있습니다.
 

추가할 게시자가 보안 게시자이면 SSL 키와 인증서를 제공해야 합니다. 시스템에서 SSL 키와 SSL 인증서를 찾아봅니다.

게시자가 성공적으로 추가되면 Adding Publisher Complete(게시자 추가 완료) 대화 상자가 표시됩니다. 확인(OK) 버튼을 눌러 설치를 계속합니다.

4. .p5i 파일에 사용 안함으로 설정된 게시자의 패키지가 포함되어 있으면 웹 설치에서는 Enable Publisher(게시자 사용) 대화 상자가 열립니다. 이 대화 상자를 사용하여 패키지 설치가 가능하도록 게시자를 사용으로 설정할 수 있습니다.

이제 Install/Update(설치/업데이트) 창이 Package Manager Install/Update(패키지 관리자 설치/업데이트) 옵션을 선택할 때와 같습니다.

모든 패키지가 설치되면 응용 프로그램이 닫힙니다.

## 업데이트 관리자 사용

업데이트 관리자는 설치된 모든 패키지를 설치된 패키지 및 게시자 구성에 따라 시스템에서 허용하는 제약 조건 내에서 가장 최신 버전으로 업데이트합니다. 이 기능은 다음 기능과 같습니다.

- 패키지 관리자 GUI에서 Updates(업데이트) 버튼을 선택하거나 Package(패키지)>Updates(업데이트) 메뉴 옵션을 선택합니다.
- `packagemanager` 명령을 사용합니다.

```
# packagemanager --update-all
```

- `pkg` 명령을 사용합니다.

```
# pkg update
```

다음 방법 중 하나를 사용하여 업데이트 관리자를 시작할 수 있습니다.

**상태 표시줄** 업데이트가 있으면 상태 표시줄에 알림이 표시됩니다. 알림에 표시된 위치를 누릅니다. 업데이트 관리자 아이콘은 세 개의 상자가 쌓여 있는 모양입니다.

**메뉴 표시줄** System(시스템)>Administration(관리)>Update Manager(업데이트 관리자)를 선택합니다.

**명령줄** `# pm-updatemanager`

Updates(업데이트) 창이 표시되고 업데이트 프로세스가 시작됩니다.

1. 시스템에서 모든 카탈로그를 새로 고칩니다.
2. 시스템에서 설치된 모든 패키지를 평가하여 업데이트가 있는 패키지를 확인합니다.
  - 업데이트가 있는 패키지가 없으면 "No Updates Available(사용 가능한 업데이트가 없음)"이라는 메시지가 표시되고 처리가 중지됩니다.
  - 패키지 업데이트가 있으면 검토를 위해 업데이트 대상 패키지가 나열됩니다. 이때가 Cancel(취소) 버튼을 눌러 업데이트를 중단할 수 있는 마지막 기회입니다.
3. 업데이트를 계속하려면 Proceed(진행) 버튼을 누릅니다. 모든 패키지 업데이트가 다운로드되어 설치됩니다.

다음 패키지는 업데이트가 있는 경우 가장 먼저 업데이트되고 그런 다음 다른 패키지가 업데이트됩니다.

```
package/pkg
package/pkg/packagemanager
package/pkg/updatemanager
```

기본적으로 각 패키지는 설치 시 원래 사용되었던 게시자에서 업데이트됩니다. 원래 게시자가 비고정인 경우 이 이미지와 호환되는 패키지의 최신 버전을 다른 게시자에서도 설치할 수 있습니다. 게시자를 고정 또는 비고정으로 설정하려면 Package Manager Manage Publisher(패키지 관리자 게시자 관리) 창이나 `pkg set-publisher` 명령을 사용합니다.

업데이트되는 패키지 및 이미지 정책에 따라 새 BE가 생성될 수 있습니다.

업데이트 프로세스 도중 오류가 발생하면 Details(세부 정보) 패널이 확장되고 오류 세부 정보가 표시됩니다. 실패한 단계 옆에 오류 상태 표시기가 나타납니다.

4. 시스템에서 업데이트 전에 새 BE를 만든 경우 기본 BE 이름을 편집할 수 있습니다. BE 이름을 그대로 사용하려면 Restart Now(지금 다시 시작) 버튼을 눌러 시스템을 바로 다시 시작합니다. 나중에 시스템을 다시 시작하려면 Restart Later(다음에 다시 시작) 버튼을 누릅니다. 새 BE로 부트하려면 다시 시작해야 합니다. 새 BE가 기본 부트 옵션으로 나타납니다. 현재 BE는 대체 부트 옵션으로 사용할 수 있습니다.

## 업데이트 관리자 명령줄 옵션

다음은 `pm-updatemanager(1)` 명령에 지원되는 옵션입니다.

표 2-2 업데이트 관리자 명령 옵션

옵션	설명
<code>--image-dir</code> 또는 <code>-R dir</code>	<i>dir</i> 에 루트 지정된 이미지에 대해 작업을 수행합니다. 기본 동작은 현재 이미지에 대해 작업을 수행하는 것입니다.  다음 명령은 <code>/aux0/example_root</code> 에 있는 이미지를 업데이트합니다.  <code># pm-updatemanager -R /aux0/example_root</code>
<code>--help</code> 또는 <code>-h</code>	명령 사용 정보를 표시합니다.



## 소프트웨어 패키지 정보 보기

이 장에서는 다음과 같은 패키지 정보를 보는데 사용할 수 있는 명령에 대해 설명합니다.

- 패키지가 설치되었는지 여부 또는 업데이트가 가능한지 여부
- 패키지에 대한 설명, 크기 및 버전
- 그룹 패키지에 포함된 패키지
- 특정 범주에 해당하는 패키지
- 지정한 파일을 제공하는 패키지

이러한 명령은 특별한 권한이 없어도 실행할 수 있습니다.

### 패키지 설치 상태 정보 표시

`pkg list` 명령은 패키지가 현재 이미지에 설치되었는지 여부와 업데이트를 사용할 수 있는지 여부를 알려줍니다. 옵션이나 피연산자가 없는 경우 이 명령은 현재 이미지에 설치된 모든 패키지를 나열합니다. 결과 범위를 좁히려 하면 하나 이상의 패키지 이름을 제공합니다. 패키지 이름에 와일드카드를 사용할 수 있습니다. 이 이미지와 일치하지 않는 아키텍처나 영역 유형에 대한 패키지 변형은 나열되지 않습니다.

```
/usr/bin/pkg list [-Hafnsuv] [-g path_or_uri ...] [--no-refresh] [pkg_fmri_pattern ...]
```

`pkg list` 명령은 각 패키지에 대해 한 라인의 정보를 표시합니다.

```
$ pkg list *toolkit*
NAME (PUBLISHER)          VERSION          IFO
isvtoolkit (isv.com)      1.0             i--
system/dtrace/dtrace-toolkit 0.99-0.174.0.0.0.17765 i--
```

괄호 안의 게시자 이름은 `isv.com` 게시자가 이 이미지의 게시자 검색 순서에서 첫번째 게시자가 아님을 나타냅니다. 이 이미지에 설치되어 있는 `dtrace-toolkit` 패키지는 검색 순서의 첫번째 게시자가 게시했습니다.

I 열의 "i"는 이러한 패키지가 이 이미지에 설치되어 있음을 나타냅니다. 설치되어 있는 패키지와 이 이미지에 설치되어 있지 않지만 설치가 가능한 패키지의 최신 버전을 나열하려면 `-a` 옵션을 사용합니다.

```
$ pkg list -a *toolkit*
NAME (PUBLISHER)          VERSION          IFO
image/nvidia/cg-toolkit   3.0.15-0.174.0.0.0.0.0 ---
isvtoolkit (isv.com)      1.0             i--
system/dtrace/dtrace-toolkit 0.99-0.174.0.0.0.0.17765 i--
```

이 출력 결과는 `image/nvidia/cg-toolkit`을 이 이미지에 설치할 수 있음을 나타냅니다.

이 이미지에 설치할 수 없는 패키지를 포함하여 일치하는 모든 패키지를 나열하려면 `-af` 옵션을 사용합니다. 패키지 중 최신 버전만 나열하려면 `@latest`를 지정합니다.

```
$ pkg list -af *toolkit@latest
NAME (PUBLISHER)          VERSION          IFO
developer/dtrace/toolkit  0.99-0.173.0.0.0.1.0  --r
image/nvidia/cg-toolkit   3.0.15-0.174.0.0.0.0.0 ---
isvtoolkit (isv.com)      1.0             i--
system/dtrace/dtrace-toolkit 0.99-0.174.0.0.0.0.17765 i--
```

이 출력 결과는 `developer/dtrace/toolkit` 패키지를 이 이미지에 설치할 수 없음을 나타냅니다. O 열의 "r"은 이 패키지 이름이 변경되었음을 나타냅니다.

`developer/dtrace/toolkit` 패키지는 `system/dtrace/dtrace-toolkit`으로 이름이 변경되었고 `system/dtrace/dtrace-toolkit`은 이미 설치되어 있습니다.

다음 예에서는 `web/amp` 패키지 이름이 변경되었습니다. `web/amp` 패키지는 `group/feature/amp`로 이름이 변경되었고 `group/feature/amp` 패키지는 설치되지 않았기 때문에 `-f` 옵션을 지정하지 않아도 이 패키지가 표시됩니다. `web/amp` 패키지를 설치하도록 명령을 지정하면 `group/feature/amp` 패키지가 자동으로 설치됩니다.

```
$ pkg list -a amp
NAME (PUBLISHER)          VERSION          IFO
group/feature/amp         0.5.11-0.174.0.0.0.0.2559 ---
web/amp                   0.5.11-0.174.0.0.0.0.0  --r
```

`pkg list` 명령은 이름이 변경된 패키지의 새 이름을 표시하지 않습니다. 이 정보를 얻으려면 25 페이지 “패키지 설명 또는 라이선스 표시”에 나온 대로 `pkg info` 명령을 사용합니다.

`-n` 옵션은 알려진 각 패키지의 최신 버전을 나열합니다. O 열의 "o"는 패키지가 오래되었음을 나타냅니다. 오래된 패키지는 설치할 수 없습니다.

```
$ pkg list -n *mysql-5?
NAME (PUBLISHER)          VERSION          IFO
database/mysql-50        5.0.91-0.171    --o
database/mysql-51        5.1.37-0.174.0.0.0.0.504 ---
```

이 출력 결과는 `database/mysql-50` 패키지를 이 이미지에 설치할 수 없음을 나타냅니다. 이 패키지는 이름이 변경되지 않았습니다. `mysql-50` 패키지를 설치하도록 명령을 지정하면 `mysql-51` 패키지가 설치되지 않습니다. 이 경우 어떤 패키지도 설치되지 않습니다.

F 열의 "f"는 패키지가 고정되어 있음을 나타냅니다. 패키지가 고정된 경우 고정된 버전과 일치하는 패키지만 설치하거나 해당 패키지로만 업데이트할 수 있습니다. 패키지 고정에 대한 자세한 내용은 46 페이지 “지정한 버전으로 패키지 잠금”을 참조하십시오.

```
$ pkg list mercurial
NAME (PUBLISHER)          VERSION          IFO
developer/versioning/mercurial  1.8.4-0.174.0.0.0.504  if-
```

-s 옵션은 패키지 이름과 요약만 나열합니다.

```
$ pkg list -ns mysql-51 feature/amp
NAME (PUBLISHER)  SUMMARY
database/mysql-51  MySQL 5.1 Database Management System
group/feature/amp  AMP (Apache, MySQL, PHP) Deployment Kit for Oracle Solaris
```

-v 옵션은 전체 패키지 FMRI를 나열합니다.

```
$ pkg list -nv mysql-51
FMRI                                                    IFO
pkg://solaris/database/mysql-51@5.1.37,5.11-0.174.0.0.0.504:20110920T230125Z ---
```

-u 옵션은 설치된 패키지 중 최신 버전이 있는 패키지를 모두 나열합니다.

작업에 대한 패키지 데이터 소스로 사용할 저장소 또는 패키지 아카이브를 지정하려면 -g 옵션을 사용합니다.

--no-refresh 옵션을 사용하면 pkg가 사용 가능한 패키지의 최신 목록을 검색하기 위해 이미지 게시자 저장소에 연결하지 않습니다.

## 패키지 설명 또는 라이선스 표시

`pkg info` 명령은 이름, 설치 상태, 버전, 패키징 날짜, 패키지 크기, 전체 FMRI 등의 패키지 정보를 표시합니다. 옵션이나 피연산자가 없는 경우 이 명령은 현재 이미지에 설치된 모든 패키지에 대한 정보를 표시합니다. 결과 범위를 좁히려면 하나 이상의 패키지 이름을 제공합니다. 패키지 이름에 와일드카드를 사용할 수 있습니다.

```
/usr/bin/pkg info [-lr] [-g path_or_uri ...] [--license] [pkg_fmri_pattern ...]
```

`info`와 `list` 하위 명령은 모두 패키지 이름, 게시자 및 버전 정보를 표시합니다. `pkg list` 명령은 패키지의 업데이트가 있는지 여부, 업데이트를 이 이미지에 설치할 수 있는지 여부 및 패키지가 오래되었거나 이름이 변경되었는지 여부를 보여줍니다. `pkg info` 명령은 패키지 요약, 설명, 범주, 크기를 표시하며 라이선스 정보만 별도로 표시할 수도 있습니다.

-r 옵션은 현재 설치되어 있지 않은 패키지에 대한 정보를 구성된 게시자 저장소에서 검색하여 사용 가능한 최신 버전을 표시합니다.

```
$ pkg info -r group/feature/amp
```

```
Name: group/feature/amp
Summary: AMP (Apache, MySQL, PHP) Deployment Kit for Oracle Solaris
Description: Provides a set of components for deployment of an AMP (Apache,
MySQL, PHP) stack on Oracle Solaris
Category: Meta Packages/Group Packages (org.opensolaris.category.2008)
Web Services/Application and Web Servers (org.opensolaris.category.2008)
State: Not installed
Publisher: solaris
Version: 0.5.11
Build Release: 5.11
Branch: 0.174.0.0.0.0.2559
Packaging Date: Wed Sep 21 19:12:55 2011
Size: 5.45 kB
FMRI: pkg://solaris/group/feature/amp@0.5.11,5.11-0.174.0.0.0.0.2559:20110921T191255Z
```

이름이 변경된 패키지의 새 이름을 찾으려면 pkg info 명령을 사용합니다. 다음 예에서는 developer/dtrace/toolkit 패키지의 새 이름이 system/dtrace/dtrace-toolkit임을 보여줍니다.

```
$ pkg info -r developer/dtrace/toolkit
```

```
Name: developer/dtrace/toolkit
Summary:
State: Not installed (Renamed)
Renamed to: pkg://system/dtrace/dtrace-toolkit@0.99,5.11-0.173.0.0.0.0.0
consolidation/osnet/osnet-incorporation
Publisher: solaris
Version: 0.99
Build Release: 5.11
Branch: 0.173.0.0.0.1.0
Packaging Date: Fri Aug 26 14:55:51 2011
Size: 5.45 kB
FMRI: pkg://solaris/developer/dtrace/toolkit@0.99,5.11-0.173.0.0.0.1.0:20110826T145551Z
```

--license 옵션은 패키지의 라이선스 텍스트를 표시합니다. 이 정보는 내용이 매우 길 수 있습니다. --license 옵션을 지정하지 않으면 위에 나온 정보가 표시되지 않습니다.

```
$ pkg info -r --license x11/server/xorg
```

```
Copyright (c) 2011, Oracle and/or its affiliates. All rights reserved.
The following software...
```

작업에 대한 패키지 데이터 소스로 사용할 저장소 또는 패키지 아카이브를 지정하려면 -g 옵션을 사용합니다.

## 패키지 매니페스트의 정보 표시

`pkg contents` 명령은 패키지의 파일 시스템 콘텐츠를 표시합니다. 옵션이나 피연산자가 없는 경우 이 명령은 현재 이미지에 설치된 모든 패키지에 대한 경로 정보를 표시합니다. 명령 옵션을 사용하여 표시할 특정 패키지 콘텐츠를 지정합니다. 결과 범위를 좁히려면 하나 이상의 패키지 이름을 제공합니다. 패키지 이름에 와일드카드를 사용할 수 있습니다.

```
/usr/bin/pkg contents [-Hmr] [-a attribute=pattern ...] [-g path_or_uri ...] [-o attribute ...] [-s sort_key] [-t ...]
```

`contents`와 `search` 하위 명령은 모두 패키지의 콘텐츠를 질의합니다. `pkg contents` 명령은 패키지의 작업과 속성을 표시합니다. `pkg search` 명령은 질의와 일치하는 패키지를 나열합니다.

다음 예에서는 `pkg contents`의 기본 동작을 보여줍니다. 표시할 작업과 속성을 지정하려면 옵션을 사용합니다.

```
$ pkg contents e1000g
PATH
kernel
kernel/drv
kernel/drv/amd64
kernel/drv/amd64/e1000g
kernel/drv/e1000g.conf
usr/share/man/man7d
usr/share/man/man7d/e1000g.7d
```

`-m` 옵션은 전체 패키지 매니페스트를 표시합니다.

`-r` 옵션은 현재 설치되어 있지 않은 패키지에 대한 정보를 구성된 게시자 저장소에서 검색하여 사용 가능한 최신 버전을 표시합니다.

작업에 대한 패키지 데이터 소스로 사용할 저장소 또는 패키지 아카이브를 지정하려면 `-g` 옵션을 사용합니다.

지정한 작업 속성을 기준으로 작업을 정렬하려면 `-s` 옵션을 사용합니다. 기본적으로 경로를 기준으로 또는 `-o` 옵션에 지정된 첫번째 속성을 기준으로 출력 결과가 정렬됩니다. `-s` 옵션은 여러 번 지정할 수 있습니다.

## 패키지를 통해 설치된 파일 나열

표시할 작업 유형을 지정하려면 `-t` 옵션을 사용합니다. 쉘표로 구분된 목록 형태를 사용하거나 `-t` 옵션을 여러 번 사용하여 여러 가지 유형을 지정할 수 있습니다.

출력 결과에 표시할 속성을 지정하려면 `-o` 옵션을 사용합니다. 쉘표로 구분된 목록 형태를 사용하거나 `-o` 옵션을 여러 번 사용하여 여러 가지 속성을 지정할 수 있습니다. 패키지 작업 및 속성 목록은 [pkg\(5\)](#) 매뉴얼 페이지를 참조하십시오. 이 예에서 `pkg.size` 의사 속성은 파일의 크기를 보여줍니다. `file` 작업에는 `size` 속성이 없습니다.

```
# pkg contents -t file -o owner,group,mode,pkg.size,path e1000g
OWNER GROUP MODE PKG.SIZE PATH
root sys 0755 420912 kernel/drv/amd64/e1000g
root sys 0644 4238 kernel/drv/e1000g.conf
root bin 0444 20 usr/share/man/man7d/e1000.7d
root bin 0444 12813 usr/share/man/man7d/e1000g.7d
```

패키지 매니페스트를 검토하면 e1000g 패키지에 7개의 파일 작업이 있음을 알 수 있습니다. 위의 출력 결과에 표시되지 않은 세 가지는 이 이미지에 설치할 수 없는 파일입니다. 이 이미지는 x86 아키텍처이며 디버그 파일을 포함하지 않습니다. x86 아키텍처용 디버그 파일은 위에 표시되지 않았으며 SPARC 아키텍처에 대해서는 디버그 파일과 비디버그 파일이 모두 표시되지 않습니다. 디버그 이미지 변형을 변경하여 이미지에 디버그 파일을 포함할지 여부를 변경할 수 있습니다. [47 페이지 “선택적 구성 요소의 설치 제어”](#)를 참조하십시오.

## 그룹 패키지의 설치 가능한 모든 패키지 나열

Oracle Solaris 11 GUI 설치 프로그램은 solaris-desktop 그룹 패키지를 설치합니다. 텍스트 설치 프로그램 및 자동 설치 프로그램 설치의 기본 AI 매니페스트는 solaris-large-server 그룹 패키지를 설치합니다. solaris-small-server 그룹 패키지는 서버에 더 작은 패키지 세트를 설치하는 데 사용할 수 있는 대안입니다. 다음 명령을 사용하여 각 그룹에 포함된 패키지 세트를 표시할 수 있습니다.

```
$ pkg contents -o fmri -H -rt depend -a type=group solaris-desktop
archiver/gnu-tar
audio/audio-utilities
...
```

-t 옵션은 패키지의 depend 작업과 일치합니다. -a 옵션은 group 유형의 depend 작업과 일치합니다. -o 옵션은 그룹 depend 작업의 fmri 속성만 표시합니다.

## 라이선스 요구 사항 표시

이 예에서는 패키지 라이선스에 동의하는 데 필요한 모든 통합 패키지를 표시합니다.

```
$ pkg contents -rt license -a must-accept=true \
-o must-accept,must-display,license,pkg.name *incorporation
MUST-ACCEPT MUST-DISPLAY LICENSE PKG.NAME
true true usr/src/pkg.license_files/lic_OTN consolidation/osnet/osnet-incorporation
```

## 패키지 검색

`pkg search` 명령을 사용하여 지정한 패턴과 일치하는 데이터가 포함된 패키지를 검색할 수 있습니다.

```
/usr/bin/pkg search [-Hlflpr] [-o attribute ...] [-s repo_uri] query
```

`pkg contents` 명령과 마찬가지로 `pkg search` 명령은 패키지의 콘텐츠를 검사합니다. `pkg contents` 명령은 콘텐츠를 반환하고 `pkg search` 명령은 질의와 일치하는 패키지의 이름을 반환합니다.

기본적으로 `pkg search` 질의어는 몇몇 경우를 제외하고는 정확하게 일치하는 항목을 찾습니다. 대소문자 구분 검색을 지정하려면 `-I` 옵션을 사용합니다. 질의어에 `?` 및 `*` 와일드카드를 사용할 수 있습니다.

질의어를 둘 이상 지정할 수 있습니다. 기본적으로 여러 질의어는 AND로 결합됩니다. OR을 사용하여 두 질의어를 명시적으로 결합할 수 있습니다.

다음과 같은 구조적 형태로 질의를 표현할 수 있습니다.

```
pkg_name:action_type:key:token
```

누락된 필드는 암시적으로 와일드카드로 처리됩니다. `pkg_name` 및 `token` 필드에는 명시적 와일드카드가 지원됩니다. `action_type`과 `key`는 정확히 일치해야 합니다. 패키지 작업 및 키 목록은 `pkg(5)` 매뉴얼 페이지를 참조하십시오.

기본적으로 이 이미지에 대해 구성된 모든 게시자와 관련된 저장소가 검색됩니다. 이 이미지에 설치된 패키지만 검색하려면 `-l` 옵션을 사용합니다. 검색할 저장소의 URI를 지정하려면 `-s` 옵션을 사용합니다.

기본적으로 현재 설치된 패키지 또는 최신 패키지 버전에 대해서만 일치하는 항목이 표시됩니다. 일치하는 버전을 모두 표시하려면 `-f` 옵션을 사용합니다.

## 특정 파일을 제공하는 패키지 식별

다음 예에서는 `libpower` 라이브러리가 `system/kernel/power` 패키지에 있음을 보여줍니다.

```
$ pkg search -l -H -o pkg.name /lib/libpower.so.1
system/kernel/power
```

## 범주별로 패키지 나열

다음 예에서는 `info.classification` 속성 값에 "Source Code Management"가 있는 모든 패키지를 식별합니다.

```
# pkg search ':set:info.classification:Source Code Management'
INDEX      ACTION VALUE      PACKAGE
info.classification set    Development/Source Code Management pkg:/developer/versioning/subversion@1.6.16
info.classification set    Development/Source Code Management pkg:/developer/versioning/git@1.7.3.2-0.174
info.classification set    Development/Source Code Management pkg:/developer/versioning/sccs@0.5.11-0.174
info.classification set    Development/Source Code Management pkg:/library/perl-5/subversion@1.6.16-0.174
info.classification set    Development/Source Code Management pkg:/library/java/subversion@1.6.16-0.174.0
info.classification set    Development/Source Code Management pkg:/library/python-2/subversion@1.6.16-0.1
info.classification set    Development/Source Code Management pkg:/developer/xopen/xcu4@0.5.11-0.174.0.0.
info.classification set    Development/Source Code Management pkg:/developer/quilt@0.47-0.174.0.0.0.504
info.classification set    Development/Source Code Management pkg:/developer/versioning/cvs@1.12.13-0.174
info.classification set    Development/Source Code Management pkg:/developer/versioning/mercurial@1.8.4-0
```

이 예에서는 반복되는 정보가 대량 표시되어 중요한 정보를 쉽게 찾아볼 수 없습니다.

다음 예에서는 `-o` 옵션을 사용하여 패키지 이름만 표시하고 `-H` 옵션을 사용하여 열 제목을 생략합니다.

```
# pkg search -o pkg.name -H ':set:info.classification:Source Code Management'
developer/versioning/subversion
developer/versioning/git
developer/versioning/sccs
library/perl-5/subversion
library/java/subversion
library/python-2/subversion
developer/xopen/xcu4
developer/quilt
developer/versioning/cvs
developer/versioning/mercurial
```

## 종속 항목 패키지 표시

아래의 예에서는 지정한 패키지에 종속된 패키지를 보여줍니다.

다음 예에서는 `system/kernel/power` 패키지에 대해 필수 종속성을 갖는 패키지를 보여줍니다. `pkg contents` 명령을 사용하여 `i86pc` 및 `system/hal` 패키지에 대해 `require` 유형 `depend` 작업을 표시한 경우 두 패키지 모두에 대해 `system/kernel/power`가 나열될 것입니다.

```
$ pkg search -l -H -o pkg.name 'depend:require:system/kernel/power'
system/kernel/dynamic-reconfiguration/i86pc
system/hal
```

다음 예에서는 많은 패키지에 `pkg:/x11/server/xorg@1.10.99`에 대한 제외 종속성이 있음을 보여줍니다.

```
$ pkg search -l -o pkg.name,fmri 'depend:exclude:'
PKG.NAME      FMRI
x11/server/xorg/driver/xorg-video-ati    pkg:/x11/server/xorg@1.10.99
x11/server/xorg/driver/xorg-video-intel  pkg:/x11/server/xorg@1.10.99
x11/server/xvnc                            pkg:/x11/server/xorg@1.10.99
desktop/remote-desktop/tigervnc        pkg:/x11/server/xorg@1.10.99
x11/server/xserver-common                pkg:/x11/server/xorg@1.10.99
...
```

## 그룹 패키지의 모든 패키지 나열

Oracle Solaris 11 GUI 설치 프로그램은 `solaris-desktop` 그룹 패키지를 설치합니다. 텍스트 설치 프로그램 및 자동 설치 프로그램 설치의 기본 AI 매니페스트는 `solaris-large-server` 그룹 패키지를 설치합니다. `solaris-small-server` 그룹 패키지는 서버에 더 작은 패키지 세트를 설치하는 데 사용할 수 있는 대안입니다. 다음 검색 품을 사용하여 각 그룹에 포함된 패키지 세트를 표시할 수 있습니다.

```
$ pkg search -o fmri -H '*/solaris-desktop:depend:group:'
archiver/gnu-tar
audio/audio-utilities
...
```

이 예에서 `-o pkg.name`은 질의의 `pkg_name` 필드에 지정된 패키지의 이름만 반환합니다.

```
group/system/solaris-desktop
```

`-o fmri` 옵션은 `solaris-desktop` 패키지에서 `group` 유형 종속성으로 지정된 패키지의 FMRI를 반환합니다.

기본적으로 이 이미지에 설치할 수 있는 패키지만 반환됩니다. 이 예에서는 일치하는 패키지가 반환되지 않고 대신 지정된 패키지에 있는 작업의 속성 값이 반환됩니다. 이 예에서는 속성 값이 패키지 이름입니다. 이 명령으로 반환되는 결과는 이와 비슷한 `pkg contents` 명령으로 반환되는 결과보다 결과의 수가 많습니다. 이 명령의 검색 결과에는 설치 가능한 패키지뿐 아니라 지정된 패키지의 그룹 종속 작업에 지정된 모든 패키지의 이름이 포함되기 때문입니다. 예를 들어, 이 이미지에 설치할 수 없는 패키지 변형이 포함될 수도 있습니다. 이 검색의 출력 결과를 [28 페이지 “그룹 패키지의 설치 가능한 모든 패키지 나열”](#)에 나온 `pkg contents` 명령의 출력 결과와 비교해 보십시오.

---

**참고** - 일반적으로 지정된 패키지의 콘텐츠를 표시하려면 `pkg contents` 명령을 사용하고 질의와 일치하는 패키지를 표시하려면 `pkg search` 명령을 사용합니다. 원하는 콘텐츠를 제공하는 패키지를 확인하려면 `pkg contents` 명령을 사용합니다.

---



## 소프트웨어 패키지 설치 및 업데이트

---

패키지 설치 및 업데이트는 일부 패키지를 특정 버전으로 제한하거나 게시자 검색 순서를 구성하거나 패키지 서명 등록 정보를 설정하는 등의 이미지 구성의 영향을 받습니다. 이미지 구성에 대한 설명은 5 장, “설치된 이미지 구성”을 참조하십시오. 이 장에 나온 지침과 결과는 기본 이미지 구성을 사용한 경우를 가정한 것입니다.

이미 설치된 패키지, 설치 가능한 패키지 및 업데이트가 있는 패키지를 확인하는 방법은 3 장, “소프트웨어 패키지 정보 보기”를 참조하십시오.

이 장에서는 다음 작업을 수행하는 방법을 보여줍니다.

- 시험 설치를 실행하여 설치가 성공할지 여부 및 어떤 항목이 설치되는지를 확인합니다.
- 패키지를 설치, 업데이트 및 제거합니다.
- 패키지를 검증합니다.
- 설치된 패키지의 문제를 수정합니다.
- 설치된 파일을 원래 콘텐츠로 복원합니다.
- 패키지를 제거합니다.

패키지 설치, 업데이트 및 제거 작업을 수행하려면 높은 수준의 권한이 필요합니다. 자세한 내용은 12 페이지 “설치 권한”을 참조하십시오.

### 작업 미리 보기

이 장과 5 장, “설치된 이미지 구성”에 나오는 대부분의 명령에는 실제로 변경하지 않고 명령으로 어떤 작업이 수행될지 미리 보는데 사용할 수 있는 `-n` 옵션이 있습니다.

**참고** - 가능하면 항상 `-n` 옵션을 사용하는 것이 좋습니다. `-n` 옵션과 하나 이상의 상세 정보 표시 옵션(`-nv`, `-nvv`)을 함께 사용하여 먼저 명령의 결과를 검토한 후 `-n` 옵션을 생략하고 명령을 실행할 수 있습니다.

다음 예에서는 실제로 수행되지 않은 패키지 설치에 대한 정보를 보여줍니다.

```
# pkg install -nv group/feature/amp
  Packages to install: 8
  Estimated space available: 112.19 GB
  Estimated space to be consumed: 452.42 MB
  Create boot environment: No
  Create backup boot environment: No
  Services to change: 2
  Rebuild boot archive: No
Changed packages:
solaris
  database/mysql-51
  None -> 5.1.37,5.11-0.174.0.0.0.0.504:20110920T230125Z
  group/feature/amp
  None -> 0.5.11,5.11-0.174.0.0.0.0.2559:20110921T191255Z
  web/php-52
  None -> 5.2.17,5.11-0.174.0.0.0.0.504:20110921T041858Z
  web/php-52/extension/php-apc
  None -> 3.0.19,5.11-0.174.0.0.0.0.504:20110921T041245Z
  web/php-52/extension/php-mysql
  None -> 5.2.17,5.11-0.174.0.0.0.0.504:20110921T041411Z
  web/server/apache-22/module/apache-dtrace
  None -> 0.3.1,5.11-0.174.0.0.0.0.504:20110921T042357Z
  web/server/apache-22/module/apache-fcgid
  None -> 2.3.6,5.11-0.174.0.0.0.0.504:20110921T042430Z
  web/server/apache-22/module/apache-php5
  None -> 5.2.17,5.11-0.174.0.0.0.0.504:20110921T042738Z
Services:
  restart_fmri:
    svc:/system/manifest-import:default
    svc:/system/rbac:default
```

다음 명령은 대량의 출력 결과를 생성하는데 이는 영향을 받는 패키지가 너무 많기 때문입니다. 사용될 추가 공간의 양이 메가바이트가 아니라 기가바이트 단위라는 점에 주목하십시오. 이 작업은 완료되는데 많은 시간이 필요할 수 있으며 이 이미지와 패키지 저장소 간에 대량의 네트워크 전송이 발생할 수 있습니다. 새 BE는 기본적으로 생성되지 않지만 대신 백업 BE가 생성됩니다. BE 생성 시점에 대한 자세한 내용은 [52 페이지 “부트 환경 정책 이미지 등록 정보”](#)를 참조하십시오.

```
# pkg change-facet -nv facet.locale.*=true
  Packages to update: 831
  Variants/Facets to change: 1
  Estimated space available: 112.19 GB
  Estimated space to be consumed: 2.96 GB
  Create boot environment: No
  Create backup boot environment: Yes
  Rebuild boot archive: No
```

```

Changed variants/facets:
  facet facet.locale.*: True
Changed packages:
solaris
...

```

## 패키지 설치 및 업데이트

`pkg install` 명령은 현재 설치되어 있지 않은 패키지를 설치하고 이미 설치된 패키지를 업데이트합니다. `pkg install` 명령에는 하나 이상의 패키지 이름이 필요합니다.

`pkg update` 명령은 설치된 패키지를 업데이트합니다. 아직 설치되어 있지 않은 패키지를 `pkg update` 명령에 지정하면 해당 패키지가 설치되지 않습니다. `pkg update` 명령에는 이미 설치된 패키지 이름이 없거나 하나 이상 사용될 수 있습니다. 패키지 이름을 지정하지 않으면 이미지의 설치된 모든 패키지가 업데이트됩니다.

`pkg(5)` 매뉴얼 페이지에서 `file` 작업의 `preserve` 및 `overlay` 속성을 참조하여 설치 및 업데이트 중 이러한 속성을 가진 파일이 어떻게 처리되는지 확인할 수 있습니다.

## 부트 환경 옵션

패키지를 설치, 업데이트 또는 제거하거나 파일을 되돌릴 때 새 BE나 백업 BE가 생성될 수 있습니다. BE 관련 이미지 정책이 허용하는 제약 조건 내에서 아래에 설명된 옵션을 사용하여 새 BE 및 백업 BE 생성을 제어할 수 있습니다. 새 BE와 백업 BE 및 BE 관련 이미지 정책 설정에 대한 자세한 내용은 52 페이지 “부트 환경 정책 이미지 등록 정보”를 참조하십시오.

새 BE 또는 백업 BE를 강제로 만들거나 만들지 않도록 설정하고, BE에 사용자 정의 이름을 제공하고, 새 BE가 활성화되지 않도록 지정하려면 BE 옵션을 사용합니다.

- |   |   |
|---|---|
| <code>--no-be-activate</code>             | BE가 생성된 경우 해당 BE가 다음 부트 시 활성화 BE가 되도록 설정하지 않습니다. 활성화 BE를 표시하고 변경하려면 <code>beadm(1M)</code> 명령을 사용합니다.   |
| <code>--no-backup-be</code>               | 백업 BE를 만들지 않습니다.  |
| <code>--require-backup-be</code>          | 새 BE가 생성되지 않을 경우 백업 BE를 만듭니다. 이 옵션을 생략할 경우 백업 BE가 생성되는지 여부는 이미지 정책에 따라 다릅니다. 백업 BE가 자동으로 생성되는 경우에 대한 설명은 52 페이지 “부트 환경 정책 이미지 등록 정보”를 참조하십시오. |
| <code>--backup-be-name <i>name</i></code> | 백업 BE가 생성되면 기본 이름 대신 <i>name</i> 이라는 이름을 지정합니다. <code>--backup-be-name</code> 을 사용하면 <code>-require-backup-be</code> 가 지정된 것을 의미합니다.          |

<code>--deny-new-be</code>	새 BE를 만들지 않습니다. 새 BE가 필요한 경우 설치, 업데이트, 제거 또는 되돌리기 작업이 수행되지 않습니다.
<code>--require-new-be</code>	새 BE를 만듭니다. 이 옵션을 생략할 경우 BE가 생성되는지 여부는 이미지 정책에 따라 다릅니다. BE가 자동으로 생성되는 경우에 대한 설명은 52 페이지 “부트 환경 정책 이미지 등록 정보”를 참조하십시오. 이 옵션은 <code>--require-backup-be</code> 와 함께 사용할 수 없습니다.
<code>--be-name name</code>	BE가 생성되면 기본 이름 대신 <i>name</i> 이라는 이름을 지정합니다. <code>--be-name</code> 을 사용하면 <code>--require-new-be</code> 가 지정된 것을 의미합니다.

## 새 패키지 설치

기본적으로 나머지 이미지와 호환되는 패키지의 가장 최신 버전이 게시자 검색 순서의 패키지를 제공한 첫번째 게시자에서 설치됩니다.

패키지가 이미 설치되어 있으면 나머지 이미지와 호환되는 패키지의 가장 최신 버전을 현재 설치된 버전을 제공한 게시자에서 설치하여 패키지가 업데이트됩니다.

이미지에 대해 사용으로 설정된 게시자가 둘 이상인 경우 게시자 고착성과 검색 순서를 설정하거나 패키지 FMRI에 게시자를 지정하여 패키지를 제공하는 게시자를 제어할 수 있습니다. 설치할 버전도 패키지 FMRI에 지정할 수 있습니다. 패키지 FMRI에 대한 설명은 13 페이지 “FMRI(Fault Management Resource Identifier)”를 참조하십시오. 게시자 고착성 및 검색 순서 설정에 대한 자세한 내용은 43 페이지 “게시자 구성”을 참조하십시오.

```
/usr/bin/pkg install [-nvq] [-g path_or_uri ...] [--accept] [--licenses] [--no-index] [--no-refresh] [--no-be-activate]
```

특정 게시자의 패키지를 설치하려면 *pkg\_fmri\_pattern*에 게시자 이름을 지정합니다. 다음 예에서 *isv.com*은 게시자의 이름입니다.

```
# pkg install pkg://isv.com/developer/isvtool
```

패키지의 특정 버전을 설치하려면 *pkg\_fmri\_pattern*에 버전 정보를 지정합니다.

```
# pkg list -avH vim
pkg://solaris/editor/vim@7.3.254,5.11-0.174.0.0.0.504:20110921T002716Z ---
# pkg install vim@7.3.254,5.11-0.174
```

패키지의 최신 버전을 명시적으로 요청하려면 *pkg\_fmri\_pattern*의 버전 부분에 *latest*를 사용합니다.

```
# pkg install vim@latest
```

지정한 패키지 저장소나 패키지 아카이브를 이미지의 임시로 패키지 데이터를 검색할 소스 목록에 추가하려면 `-g` 옵션을 사용합니다. `install` 또는 `update` 뒤에 게시자가 제공했지만 이미지에 없는 패키지를 지정하면 해당 패키지가 원본 없이 이미지 구성에 추가됩니다.

업데이트되거나 설치된 패키지의 사용권 계약에 동의함을 나타내려면 `--accept` 옵션을 사용합니다. 이 옵션을 제공하지 않고 동의해야 할 패키지 라이선스가 있을 경우 설치 작업이 실패합니다. 이 작업의 일부로 설치되거나 업데이트되는 패키지의 모든 라이선스를 표시하려면 `--licenses` 옵션을 사용합니다.

`--no-index` 옵션을 지정하면 작업이 성공적으로 완료된 후 검색 색인은 업데이트되지 않습니다. 패키지를 다수 설치할 경우 이 옵션을 지정하면 상당 시간을 절약할 수 있습니다. 모든 `install`, `update` 및 `uninstall` 작업이 완료되면 `pkg refresh`를 사용하여 지정한 각 게시자의 게시자 메타 데이터 및 사용 가능한 패키지 목록을 업데이트합니다. 게시자를 지정하지 않으면 모든 게시자에 대해 새로 고침 작업이 수행됩니다.

`-no-refresh` 옵션을 지정하면 사용 가능한 패키지 및 기타 메타 데이터의 최신 목록을 검색하기 위해 이미지 게시자 저장소에 연결하지 않습니다.

## 새 부트 환경에 패키지 설치

**참고** - 가장 안전한 설치 또는 업데이트 방법은 새 BE를 명시적으로 지정하는 것입니다. BE 생성 시점에 대한 자세한 내용은 [52 페이지 “부트 환경 정책 이미지 등록 정보”](#)를 참조하십시오.

새 BE는 지정한 설치, 제거 또는 업데이트 변경 사항이 적용된 현재 BE의 복제본입니다. 현재 BE는 수정되지 않습니다. 시스템은 자동으로 다시 시작되지 않습니다. 새 BE는 다음에 시스템을 다시 시작할 때 기본적으로 선택되는 부트 옵션이지만 현재 BE로 부트하는 것도 가능합니다.

`--no-be-activate` 옵션을 지정할 경우 다음 재부트 시 새 BE가 기본 부트 옵션으로 선택되지 않습니다.

새 BE가 강제로 생성되도록 설정하거나 새 BE에 의미 있는 이름을 지정하려면 `--be-name` 옵션을 사용합니다.

```
# pkg install --be-name sllamp group/feature/amp
  Packages to install:  8
    Create boot environment: Yes
  Create backup boot environment:  No
```

DOWNLOAD	PKGS	FILES	XFER (MB)
Completed	8/8	640/640	70.9/70.9

```
PHASE
Install Phase                                ACTIONS
                                           942/942
```

```
PHASE
Package State Update Phase                 ITEMS
                                           8/8
Image State Update Phase                   2/2
```

```
PHASE
Reading Existing Index                     ITEMS
                                           8/8
Indexing Packages                         8/8
```

A clone of solaris-174 exists and has been updated and activated.  
On the next boot the Boot Environment s11amp will be  
mounted on '/'. Reboot when ready to switch to this updated BE.

```
# pkg list group/feature/amp
pkg list: no packages matching 'group/feature/amp' installed
```

pkg list 명령은 현재 BE에 group/feature/amp 패키지가 설치되어 있지 않기 때문에  
group/feature/amp 패키지가 설치되지 않았음을 보고합니다. group/feature/amp  
패키지는 새 s11amp BE에 설치되어 있습니다.

시스템에 s11amp라는 새로운 활성 BE가 있는지 확인하려면 beadm list 명령을  
사용합니다. "N" BE는 현재 부트된 BE이고 "R" BE는 재부트 시의 기본 BE입니다. 재부트  
시의 기본 BE를 변경하려면 beadm activate 명령을 사용합니다.

```
# beadm list
BE      Active Mountpoint Space  Policy Created
--      -
s11amp  R      -      20.75G  static 2011-09-23 13:58
solaris -      -      44.81M  static 2010-11-07 17:45
solaris-151a -    -      158.12M static 2010-11-12 14:37
solaris-174 N    /      30.04M  static 2011-09-02 12:38
```

group/feature/amp 패키지가 새 BE에 설치되어 있는지 확인합니다. I 열의 "i"는  
group/feature/amp 패키지가 설치되어 있음을 나타냅니다.

```
# beadm mount s11amp /mnt
# pkg -R /mnt list group/feature/amp
NAME (PUBLISHER)      VERSION          IFO
group/feature/amp    0.5.11-0.174.0.0.0.2559  i--
```

s11amp BE를 반드시 마운트 해제해야 합니다.

```
# beadm list
BE      Active Mountpoint Space  Policy Created
--      -
s11amp  R      /mnt    20.75G  static 2011-09-23 13:58
solaris -      -      44.81M  static 2010-11-07 17:45
solaris-151a -    -      158.12M static 2010-11-12 14:37
solaris-174 N    /      30.05M  static 2011-09-02 12:38
# beadm unmount s11amp
# beadm list
BE      Active Mountpoint Space  Policy Created
```

```
--
-----
sllamp      R      -      20.75G  static 2011-09-23 13:58
solaris    -      -      44.81M  static 2010-11-07 17:45
solaris-151a -    -      158.12M static 2010-11-12 14:37
solaris-174 N      /      30.06M  static 2011-09-02 12:38
```

## 패키지 거부

`pkg install` 명령의 `--reject` 옵션을 사용하여 지정한 `pkg_fmri_pattern`과 이름이 일치하는 패키지가 설치되지 않도록 할 수 있습니다. 일치하는 패키지가 이미 설치되어 있으면 이 작업의 일부로 제거됩니다. 그룹 종속성의 대상인 거부된 패키지는 무시 목록에 배치됩니다. 무시 목록에 대한 자세한 내용은 [47 페이지 “패키지를 무시할 패키지로 표시”](#)를 참조하십시오.

```
# pkg install -nv --reject cvs developer-gnu
```

## 패키지 업데이트

`install` 또는 `update` 하위 명령을 사용하여 설치된 패키지를 나머지 이미지와 호환되는 패키지의 최신 버전으로 업데이트할 수 있습니다. 이때 현재 설치된 버전을 제공한 게시자를 사용하여 업데이트가 이루어집니다. `pkg update` 명령을 사용하여 패키지를 업데이트하면 아직 설치되지 않은 패키지를 실수로 설치하는 것을 방지할 수 있습니다.

이미지에 대해 사용으로 설정된 게시자가 둘 이상인 경우 게시자 고착성과 검색 순서를 설정하거나 패키지 FMRI에 게시자를 지정하여 패키지를 제공하는 게시자를 제어할 수 있습니다. 설치할 버전도 패키지 FMRI에 지정할 수 있습니다. 패키지 FMRI에 대한 설명은 [13 페이지 “FMRI\(Fault Management Resource Identifier\)”](#)를 참조하십시오. 게시자 고착성 및 검색 순서 설정에 대한 자세한 내용은 [43 페이지 “게시자 구성”](#)을 참조하십시오.

```
/usr/bin/pkg update [-fnvq] [-g path_or_uri ...] [--accept] [--licenses] [--no-index] [--no-refresh] [--no-be-act]
```

패키지의 최신 버전을 명시적으로 요청하려면 `pkg_fmri_pattern`의 버전 부분에 `latest`를 사용합니다.

```
# pkg update vim@latest
```

현재 설치된 버전보다 이전 버전을 지정하여 인플레이스(**in-place**) 다운그레이드를 수행할 수 있습니다. 다운그레이드할 패키지의 일부이며 원래 버전이 설치된 이후 변경된, 보존된 구성 파일은 `.update`라는 확장자가 추가되어 이름이 바뀝니다. 패키지 시스템에서 보존할 파일을 확인하는 방법 및 패키지 업그레이드 중 이러한 파일이 보존되는 방법에 대한 자세한 내용은 [pkg\(5\)](#) 매뉴얼 페이지에서 "File Actions"를 참조하십시오.

지정한 패키지 저장소나 패키지 아카이브를 이미지의 임시로 패키지 데이터를 검색할 소스 목록에 추가하려면 `-g` 옵션을 사용합니다. `install` 또는 `update` 뒤에 게시자가 제공했지만 이미지에 없는 패키지를 지정하면 해당 패키지가 원본 없이 이미지 구성에 추가됩니다.

업데이트된 패키지의 사용권 계약에 동의함을 나타내려면 `--accept` 옵션을 사용합니다. 모든 패키지 라이선스에는 사용권 동의가 필요하기 때문에 이 옵션을 제공하지 않을 경우 업데이트 작업이 실패합니다. 이 작업의 일부로 업데이트되는 패키지의 모든 라이선스를 표시하려면 `--licenses` 옵션을 사용합니다.

`--no-index` 옵션을 지정하면 작업이 성공적으로 완료된 후 검색 색인은 업데이트되지 않습니다. 패키지를 다수 설치할 경우 이 옵션을 지정하면 상당 시간을 절약할 수 있습니다. 모든 `install`, `update` 및 `uninstall` 작업이 완료되면 `pkg refresh`를 사용하여 지정한 각 게시자의 게시자 메타 데이터 및 사용 가능한 패키지 목록을 업데이트합니다. 게시자를 지정하지 않으면 모든 게시자에 대해 새로 고침 작업이 수행됩니다.

`pkg-fmri`를 지정하지 않고 `pkg update` 명령을 사용하거나 지정한 `pkg-fmri`가 별표 문자(\*)이면 업데이트가 있는 설치된 모든 패키지가 업데이트됩니다. 업데이트가 있는 패키지 목록을 표시하려면 `pkg list -u` 명령을 사용합니다. 설치된 모든 패키지를 업데이트할 때 `-f` 옵션을 지정하면 패키지가 최신인지 확인하는 클라이언트 검사가 실행되지 않습니다.

## 패키지 문제 수정

패키지를 설치한 후 발생할 수 있는 문제로는 패키지에서 제공한 파일이 손상되는 경우를 들 수 있습니다. 이 절에 나온 예에서는

`/usr/share/auto_install/manifest/default.xml` 파일이 삭제되었습니다.

`pkg search` 명령을 사용하여 누락된 파일을 제공한 패키지를 확인할 수 있습니다.

```
$ pkg search -l -Ho pkg.name /usr/share/auto_install/manifest/default.xml
system/install/auto-install/auto-install-common
```

## 패키지 설치 검증

`pkg verify` 명령을 사용하여 현재 이미지의 패키지 설치를 검증할 수 있습니다.

```
/usr/bin/pkg verify [-Hqv] [pkg_fmri_pattern ...]
```

관련 게시자에 대한 현재 서명 정책이 `ignore`가 아니면 정책을 기반으로 각 패키지의 서명이 검증됩니다. 서명 정책을 적용하는 방법에 대한 설명은 [54 페이지 “패키지 서명에 필요한 등록 정보”](#)의 `signature-policy`를 참조하십시오.

검증 출력 결과에서 머리글을 생략하려면 `-H` 옵션을 사용합니다. 치명적 오류가 발견될 경우 아무 것도 출력하지 않고 실패를 반환하려면 `-q` 옵션을 사용합니다. 패키지와 관련된 정보 메시지를 포함하려면 `-v` 옵션을 사용합니다.

```
# pkg verify -v system/install/auto-install/auto-install-common
PACKAGE                               STATUS
pkg://solaris/system/install/auto-install/auto-install-common  ERROR
      file: usr/share/auto_install/manifest/default.xml
      Missing: regular file does not exist
```

## 검증 오류 수정

`pkg fix` 명령을 사용하여 `pkg verify` 명령이 보고한 패키지 설치 오류를 수정할 수 있습니다.

```
/usr/bin/pkg fix [--accept] [--licenses] [pkg_fmri_pattern ...]
```

설치된 패키지 콘텐츠에 대한 검증은 사용자 정의 콘텐츠 분석을 기반으로 수행되며 사용자 정의 콘텐츠 분석은 다른 프로그램과는 다른 결과를 반환할 수 있습니다.

업데이트되거나 설치된 패키지의 사용권 계약에 동의함을 나타내려면 `--accept` 옵션을 사용합니다. 모든 패키지 라이선스에는 사용권 동의가 필요하기 때문에 이 옵션을 제공하지 않을 경우 수정 작업이 실패합니다. 이 작업의 일부로 업데이트되는 패키지의 모든 라이선스를 표시하려면 `--licenses` 옵션을 사용합니다.

```
# pkg fix --accept system/install/auto-install/auto-install-common
Verifying: pkg://solaris/system/install/auto-install/auto-install-common  ERROR
      file: usr/share/auto_install/manifest/default.xml
      Missing: regular file does not exist
Created ZFS snapshot: 2011-09-28-05:34:02
Repairing: pkg://solaris/system/install/auto-install/auto-install-common

DOWNLOAD                PKGS      FILES    XFER (MB)
Completed                1/1       1/1      0.0/0.0

PHASE                    ACTIONS
Update Phase              1/1

PHASE                    ITEMS
Image State Update Phase 2/2
```

## 파일 복원

`pkg revert` 명령을 사용하여 파일을 원래 제공된 상태로 복원할 수 있습니다.

```
/usr/bin/pkg revert [-nv] [--no-be-activate] [--no-backup-be | --require-backup-be] [--backup-be-name name] [--d
```

특정 *tag-name*이 지정된 모든 파일을 되돌리거나 개별 파일을 되돌릴 수 있습니다. 파일 소유권과 보호도 함께 복원됩니다.



---

주의 - 몇몇 편집 가능한 파일을 해당 기본값으로 되돌리면 시스템을 부트할 수 없게 되거나 다른 오작동이 발생할 수 있습니다.

---

## 패키지 제거

`pkg uninstall` 명령을 사용하여 설치된 패키지를 제거할 수 있습니다.

```
/usr/bin/pkg uninstall [-nvq] [--no-index] [--no-be-activate] [--no-backup-be | --require-backup-be] [--backup-be-name
```

패키지가 그룹 종속성의 주체일 경우 패키지를 제거하면 패키지가 무시 목록에 놓입니다. 무시 목록에 대한 자세한 내용은 [47 페이지](#) “패키지를 무시할 패키지로 표시”를 참조하십시오.

`--no-index` 옵션을 지정하면 작업이 성공적으로 완료된 후 검색 색인은 업데이트되지 않습니다. 패키지를 다수 설치할 경우 이 옵션을 지정하면 상당 시간을 절약할 수 있습니다. 모든 `install`, `update` 및 `uninstall` 작업이 완료되면 `pkg refresh`를 사용하여 지정된 각 게시자의 게시자 메타 데이터 및 사용 가능한 패키지 목록을 업데이트합니다. 게시자를 지정하지 않으면 모든 게시자에 대해 새로 고침 작업이 수행됩니다.

## 설치된 이미지 구성

---

이 장에서는 패키지 게시자 구성, 설치 가능한 패키지 제한, 패키지 서명 정책 설정, BE 정책 구성 등 전체 이미지에 적용되는 특성을 구성하는 방법을 보여줍니다.

### 게시자 구성

소프트웨어를 설치하고 업데이트하려면 패키지 저장소에 연결할 수 있어야 합니다.

### 게시자 정보 표시

`pkg publisher` 명령을 사용하여 이 이미지에 대해 구성된 패키지 게시자에 대한 정보를 표시할 수 있습니다. 패키지 FMRI에 게시자가 지정되지 않은 경우에는 패키지를 찾기 위해 게시자가 검색되는 순서로 게시자가 나열됩니다.

```
/usr/bin/pkg publisher [-HPn] [publisher ...]
```

새로 설치한 Oracle Solaris 11 시스템에는 기본적으로 `solaris` 게시자가 구성되어 있습니다. `pkg publisher` 명령을 사용하여 게시자 원본을 확인할 수 있습니다.

```
$ pkg publisher
PUBLISHER          TYPE   STATUS  URI
solaris            origin online  http://pkg.oracle.com/solaris11/release/
isv.com            (non-sticky) origin online  file:/export/isv-repo/
example.com        (disabled) origin online  http://pkg.example.com/
```

해당 게시자에 대한 자세한 구성 정보를 표시하려면 게시자 이름을 지정합니다.

```
$ pkg publisher solaris
Publisher: solaris
Alias:
Origin URI: http://pkg.oracle.com/solaris11/release/
SSL Key: None
SSL Cert: None
```

Client UUID: 00000000-3db4-fcc2-0111-000000000000  
 Catalog Updated: Thu Sep 22 21:06:03 2011  
 Enabled: Yes  
 Signature Policy: verify

게시자 검색 순서의 첫번째 게시자만 표시하려면 `-p` 옵션을 사용하고 사용으로 설정된 게시자만 표시하려면 `-n` 옵션을 사용합니다. `-H` 옵션은 출력 결과에서 머리글을 생략합니다.

## 패키지 게시자 추가, 수정 또는 제거

`pkg set-publisher` 명령을 사용하여 다음 작업을 수행할 수 있습니다.

- 새 게시자를 구성합니다.
- 게시자 원본 및 미러를 설정합니다.
- 게시자를 사용으로 설정하거나 사용 안함으로 설정합니다. 새로 추가한 게시자는 기본적으로 사용으로 설정됩니다. 사용 안함으로 설정된 게시자는 패키지 목록을 채울 때나 패키지 설치, 제거 또는 업데이트 작업 중에 사용되지 않습니다. 사용 안함으로 설정된 게시자에 대해 등록 정보를 설정하거나 볼 수는 있습니다. 사용으로 설정된 게시자가 하나뿐인 경우 해당 게시자를 사용 안함으로 설정할 수 없습니다.
- 게시자 고정성을 설정합니다. 새로 추가한 게시자는 기본적으로 고정되어 있습니다. 게시자가 비고정인 경우 이 게시자에서 설치된 패키지를 다른 게시자에서 업데이트할 수 있습니다.
- 게시자 검색 순서를 설정합니다. 새로 추가한 게시자는 기본적으로 검색 순서에서 마지막 게시자가 됩니다. 게시자 검색 순서는 설치할 패키지를 찾는 데 사용됩니다. 패키지 설치에 사용된 원본 게시자가 비고정인 경우에는 업데이트할 패키지를 찾는 데 게시자 검색 순서가 사용됩니다.
- 게시자의 SSL 키와 인증서를 지정합니다.
- 게시자 등록 정보를 설정 및 해제하고, 게시자 등록 정보 값을 추가 및 제거합니다. [55 페이지 “패키지 서명 등록 정보 구성”](#)을 참조하십시오.

```
/usr/bin/pkg set-publisher [-Ped] [-k ssl_key] [-c ssl_cert] [-g origin_to_add | --add-origin origin_to_add ...] [-G origin_
```

다음 명령은 `-g` 옵션을 사용하여 원본 URI가 지정된 `data.com`이라는 새 게시자를 추가하고 이 게시자를 검색 순서의 첫번째 게시자로 설정합니다. 지정한 게시자를 검색 순서의 첫번째 게시자로 설정하려면 `-P` 옵션이나 `--search-first` 옵션을 사용합니다.

```
# pkg set-publisher -P -g http://pkg.data.com/release/ data.com
```

다음 명령은 `example.com` 게시자를 사용으로 설정하고 검색 순서에서 `isv.com`의 앞에 나오도록 설정합니다.

```
# pkg set-publisher --enable --search-before isv.com example.com
```

지정한 저장소 URI에서 게시자 구성 정보를 검색하려면 `-p` 옵션을 사용합니다. 게시자를 지정하면 일치하는 게시자만 추가되거나 업데이트됩니다. 게시자를 지정하지 않으면 모든 게시자가 추가되거나 업데이트됩니다. `-p` 옵션은 `-g`, `--add-origin`, `-G`, `--remove-origin`, `-m`, `--add-mirror`, `-M`, `--remove-mirror`, `--disable`, `--enable`, `--no-refresh` 또는 `--reset-uuid` 옵션과 함께 사용할 수 없습니다.

게시자의 원본 URI를 변경하려면 새 URI를 추가하고 기존 URI를 제거합니다. 새 원본 URI를 추가하려면 `-g` 옵션을 사용합니다. 기존 원본 URI를 제거하려면 `-G` 옵션을 사용합니다.

```
# pkg set-publisher -G '*' -g http://pkg.example.com/support/ example.com
```

URI를 지정한 게시자의 미러로 추가하려면 `-m` 옵션을 사용합니다. 원본과 미러의 차이점에 대한 설명은 [14 페이지 “저장소 원본 및 미러”](#)를 참조하십시오. URI를 지정한 게시자의 미러로 제거하려면 `-M` 옵션을 사용합니다.

```
# pkg set-publisher -m http://pkg.data.com/release2/ data.com
# pkg publisher
PUBLISHER          TYPE    STATUS  URI
data.com           origin online  http://pkg.data.com/release/
data.com           mirror  online  http://pkg.data.com/release2/
```

클라이언트 SSL 키를 지정하려면 `-k` 옵션을 사용합니다. 클라이언트 SSL 인증서를 지정하려면 `-c` 옵션을 사용합니다. 지정한 인증서를 인증된 CA 인증서로 추가하려면 `--approve-ca-cert` 옵션을 사용합니다. 사용자가 승인한 CA 인증서의 해시는 이 게시자에 대한 `pkg publisher` 명령의 출력 결과에 나열됩니다. [43 페이지 “게시자 정보 표시”](#)를 참조하십시오.

```
# pkg set-publisher -k /root/creds/example.key -c /root/creds/example.cert \
--approve-ca-cert /tmp/example_file.pem example.com
```

지정한 인증서를 취소된 것으로 처리하려면 `--revoked-ca-cert` 옵션을 사용합니다. 사용자가 취소한 CA 인증서의 해시는 이 게시자에 대한 `pkg publisher` 명령의 출력 결과에 나열됩니다.

지정한 인증서를 인증된 인증서 목록 및 취소된 인증서 목록에서 제거하려면 `--unset-ca-cert` 옵션을 사용합니다.

`-no-refresh` 옵션을 지정하면 사용 가능한 패키지 및 기타 메타 데이터의 최신 목록을 검색하기 위해 이미지 게시자 저장소에 연결하지 않습니다.

이 이미지가 해당 게시자에게 식별되도록 하는 새로운 고유 식별자를 선택하려면 `--reset-uuid`를 사용합니다.

게시자를 제거하려면 `pkg unset-publisher` 명령을 사용합니다.

```
# pkg unset-publisher isv.com
```

## 지정한 버전으로 패키지 잠금

패키지 버전을 제한하려면 `pkg freeze` 명령을 사용합니다. 예를 들어, 전역 영역이 업데이트될 때 비전역 영역의 패키지를 업데이트하지 않으려는 경우 패키지를 고정하면 유용합니다.

```
/usr/bin/pkg freeze [-n] [-c reason] [pkg_fmri_pattern] ...
```

`pkg_fmri_pattern`에 버전을 제공하지 않은 경우 지정된 패키지는 반드시 설치되어야 하며 시스템에 설치된 버전으로 제한됩니다. `pkg_fmri_pattern`에 버전을 제공한 경우 이 제약 조건이나 고정의 동작은 `incorporate` 종속성이 설치되었을 때와 같습니다(`fmri` 속성 값이 제공된 패키지 버전일 경우).

고정된 패키지가 설치되거나 업데이트되는 경우 고정된 시점의 버전과 일치하는 버전으로 설치 또는 업데이트되어야 합니다. 예를 들어, 패키지가 1.2로 고정된 경우 1.2.1, 1.2.9, 1.2.0.0.1 등으로는 업데이트될 수 있지만 1.3이나 1.1은 될 수 없습니다.

일치하는 패키지를 찾는 데는 `pkg_fmri_pattern`에 지정된 게시자가 사용됩니다. 그러나 게시자 정보는 고정의 일부로 기록되지 않습니다. 패키지는 게시자가 아니라 해당 버전만 고려하여 고정됩니다.

이미 고정된 패키지를 고정하면 고정된 버전은 새로 지정한 버전으로 교체됩니다.

패키지를 지정하지 않으면 패키지 이름, 버전, 패키지가 고정된 시간 및 관련 이유 등 현재 고정된 패키지에 대한 정보가 표시됩니다.

패키지를 고정해도 패키지를 제거할 수 있습니다. 패키지를 제거하면 별도의 경고가 표시되지 않습니다.

패키지가 고정된 이유를 기록하려면 `-c` 옵션을 사용합니다. 고정으로 인해 설치나 업데이트가 실패할 경우 이유가 표시됩니다.

작업을 시험 실행하여 패키지를 고정하지 않아도 고정되는 패키지 목록을 표시하려면 `-n` 옵션을 사용합니다.

고정은 패키징 시스템에 의해 자동으로 시작되지 않습니다. 제약 조건을 해제하려면 `pkg unfreeze` 명령을 사용합니다.

```
/usr/bin/pkg unfreeze [-n] [pkg_name_pattern] ...
```

지정한 패키지에서 고정에 따른 제약 조건을 제거합니다. 제공한 버전은 모두 무시됩니다.

고정 해제 작업을 시험 실행하여 패키지 고정을 해제하지 않아도 고정이 해제되는 패키지 목록을 표시하려면 `-n` 옵션을 사용합니다.

## 패키지를 무시할 패키지로 표시

지정한 패키지가 그룹 종속성의 대상인 경우 해당 패키지를 무시하려면 `pkg avoid` 명령을 사용합니다.

```
/usr/bin/pkg avoid [pkg_fmri_pattern ...]
```

인수를 지정하지 않으면 `pkg avoid` 명령은 무시된 각 패키지를 해당 패키지에 대해 그룹 종속성을 갖는 패키지와 함께 표시합니다.

`pkg_fmri_pattern`을 지정하면 `pkg avoid` 명령은 지정한 패턴과 일치하는 패키지 이름을 무시 목록에 놓습니다. 현재 설치되어 있지 않은 패키지만 무시할 수 있습니다. 패키지가 그룹 종속성의 대상일 경우 패키지를 제거하면 무시 목록에 놓입니다.

패키지가 무시 목록에 있을 경우 패키지를 설치하면 해당 목록에서 패키지가 제거됩니다. 필요한 경우 필수 종속성을 충족하기 위해 무시 목록에 있는 패키지가 설치됩니다. 해당 종속성이 제거되면 패키지가 제거됩니다.

지정한 패키지를 무시 목록에서 제거하려면 `pkg unavoid` 명령을 사용합니다.

```
/usr/bin/pkg unavoid [pkg_fmri_pattern ...]
```

설치된 패키지의 그룹 종속성과 일치하는 무시 목록의 패키지는 이 하위 명령을 사용하여 무시 목록에서 제거할 수 없습니다. 그룹 종속성과 일치하는 패키지를 무시 목록에서 제거하려면 패키지를 설치하십시오.

## 선택적 구성 요소의 설치 제어

소프트웨어에는 선택적 구성 요소와 상호 배타적인 구성 요소가 포함될 수 있습니다. 선택적 구성 요소의 예로는 로켈과 설명서가 있습니다. 상호 배타적 구성 요소의 예로는 SPARC 또는 x86 및 디버그 또는 비디버그 이진이 있습니다. IPS에서는 선택적 구성 요소를 **페이킷**이라 하고 상호 배타적 구성 요소를 **변형**이라고 합니다.

페이킷과 변형은 특수한 이미지 등록 정보로 개별 패키지에 설정할 수 없습니다. 이미지에 설정된 페이킷과 변형의 현재 값을 보려면 `pkg facet` 및 `pkg variant` 명령을 사용합니다. 이미지에 설정된 페이킷과 변형의 값을 수정하려면 `pkg change-facet` 및 `pkg change-variant` 명령을 사용합니다. [pkg\(1\)](#) 매뉴얼 페이지와 아래 예제를 참조하십시오.

페이킷과 변형은 패키지 작업에서 태그로 지정됩니다. 각 페이킷 및 변형 태그에는 이름과 값이 있습니다. 단일 작업에 페이킷 및 변형 태그를 여러 개 지정할 수 있습니다. 페이킷 및 변형 태그를 여러 개 가진 구성 요소의 예로는 개발자가 사용하는 아키텍처 관련 헤더 파일이나 SPARC 전역 영역 전용 구성 요소가 있습니다.

변형 태그의 예는 `variant.arch=sparc`입니다. 페이킷 태그의 예는 `facet.devel=true`입니다. 페이킷과 변형은 종종 앞에 `facet.` 및 `variant.` 없이 지정되기도 합니다.

페이킷은 부울이므로 true(사용) 또는 false(사용 안함)로만 설정할 수 있습니다. 기본적으로 모든 페이킷은 이미지에 true로 설정된 것으로 간주됩니다. 작업의 페이킷 태그는 true 값만 가져야 합니다. 다른 값은 정의되지 않은 동작을 일으킵니다. 이미지에 설정된 페이킷은 doc.man과 같은 전체 페이킷이거나 locale.\*와 같은 패턴일 수 있습니다. 이는 페이킷 이름 공간의 일부분을 사용 안함으로 설정하고 그 안의 개별 페이킷만 사용으로 설정하려는 경우 유용합니다. 예를 들어, 다음 예제에 표시된 대로 모든 로케일을 사용 안함으로 설정한 후 한두 개의 특정 로케일만 사용으로 설정할 수 있습니다.

```
# pkg change-facet locale.*=false
[output about packages being updated]
# pkg change-facet locale.en_US=true
[output about packages being updated]
```

대부분의 변형은 원하는 수의 값을 가질 수 있습니다. 예를 들어, arch 변형을 i386, sparc, ppc, arm 또는 배포판이 지원하는 어떤 구조로든 설정할 수 있습니다. (i386 및 sparc만 Oracle Solaris에 사용됩니다.) 한가지 예외는 debug 변형입니다. debug 변형은 true 또는 false로만 설정할 수 있습니다. 다른 값은 정의되지 않은 동작을 일으킵니다. 파일 작업에 디버깅 및 디버그 버전이 모두 있는 경우, 다음 예제에 표시된 대로 양쪽 버전에 적용 가능한 debug 변형을 명시적으로 설정해야 합니다.

```
file group=sys mode=0644 overlay=allow owner=root \
  path=etc/motd pkg.csize=115 pkg.size=103 preserve=true \
  variant.debug.osnet=true

file group=sys mode=0644 overlay=allow owner=root \
  path=etc/motd pkg.csize=68 pkg.size=48 preserve=true \
  variant.debug.osnet=false
```

변형을 사용하여 패키지가 설치되도록 하려면 변형 값을 이미지에 설정해야 합니다. arch 및 zone 변형은 이미지를 만들어 초기 콘텐츠를 설치하는 프로그램에 의해 설정됩니다. debug.\* 변형은 기본적으로 이미지에서 false입니다.

이미지에 설정된 페이킷과 변형은 특정 작업이 설치되는지 여부에 영향을 줍니다.

- 페이킷 또는 변형 태그가 없는 작업은 항상 설치됩니다.
- 페이킷 태그가 있는 작업은 태그와 일치하는 모든 페이킷 또는 페이킷 패턴이 이미지에 false로 설정되지 않는 한 설치됩니다. 어떤 페이킷이 true로 설정되거나 명시적으로 설정되지 않은 경우(true가 기본값) 작업이 설치됩니다.
- 변형 태그가 있는 작업은 모든 변형 태그의 값이 이미지에 설정된 값과 동일한 경우에만 설치됩니다.
- 페이킷 및 변형 태그가 모두 있는 작업은 페이킷과 변형이 모두 작업 설치를 허용하는 경우 설치됩니다.

고유의 페이킷 및 변형 태그를 만들 수 있습니다. 다음은 Oracle Solaris에서 흔히 사용되는 태그입니다.

변형 이름	가능한 값
variant.arch	sparc, i386
variant.opensolaris.zone	global, nonglobal
variant.debug.*	true, false

다음 목록은 Oracle Solaris에서 사용되는 페이스릿 태그의 작은 샘플을 보여줍니다.

```
facet.devel          facet.doc
facet.doc.html      facet.doc.info
facet.doc.man       facet.doc.pdf
facet.locale.de     facet.locale.en_GB
facet.locale.en_US  facet.locale.fr
facet.locale.ja_JP  facet.locale.zh_CN
```

현재 이미지에 설정된 변형 및 페이스릿의 값을 표시할 수 있으며 현재 이미지의 변형 및 페이스릿을 변경할 수 있습니다. 변형과 페이스릿을 변경하면 다수의 패키지가 업데이트될 수 있으며 이 경우 새로운 BE가 필요할 수도 있습니다. 변경하기 전에 어떻게 변경될지 검토하려면 `-nv`를 사용합니다.

## 변형 표시

설정된 변형의 값을 표시하려면 `pkg variant` 명령을 사용합니다.

```
/usr/bin/pkg variant [-H] [variant_spec ...]
```

```
$ pkg variant
VARIANT          VALUE
variant.opensolaris.zone global
variant.arch     i386
$ pkg variant -H variant.arch
variant.arch i386
```

## 변형 변경

`pkg change-variant` 명령을 사용하여 변형의 값을 변경할 수 있습니다.

```
/usr/bin/pkg change-variant [-nvq] [-g path_or_uri ...] [--accept] [--licenses] [--no-be-activate] [--no-backup-be]
```

다음 명령은 대량의 출력 결과를 생성하는데 이는 영향을 받는 패키지가 너무 많기 때문입니다. 새 BE는 기본적으로 생성되지 않지만 대신 백업 BE가 생성됩니다. BE 생성 시점에 대한 자세한 내용은 52 페이지 “부트 환경 정책 이미지 등록 정보”를 참조하십시오.

`-n`을 사용하지 않고 작업을 수행할 경우 어떻게 변경될지 확인하려면 `-n` 옵션을 사용합니다. 실제로 변경하는 것은 없습니다.

```
# pkg change-variant -nv --accept variant.debug.*=true
  Packages to update:      831
  Variants/Facets to change: 4
  Estimated space available: 112.19 GB
  Estimated space to be consumed: 220.76 MB
  Create boot environment: No
  Create backup boot environment: Yes
  Rebuild boot archive:    No
Changed variants/facets:
  variant variant.debug.*: true
  facet facet.locale.en_US: None
  facet facet.locale.en: None
  facet facet.locale.*: None
Changed packages:
solaris
...
```

## 페이킷 표시

pkg facet 명령을 사용하여 설정된 페이킷의 값을 표시할 수 있습니다.

```
/usr/bin/pkg facet [-H] [facet_spec ...]
```

```
$ pkg facet
FACETS          VALUE
facet.locale.en_US True
facet.locale.en  True
facet.locale.*   False
$ pkg facet -H facet.locale.*
facet.locale.* False
```

## 페이킷 변경

pkg change-facet 명령을 사용하여 페이킷의 값을 변경할 수 있습니다.

```
/usr/bin/pkg change-facet [-nvq] [-g path_or_uri ...] [--accept] [--licenses] [--no-be-activate] [--no-backup-be | --re
```

-n을 사용하지 않고 작업을 수행할 경우 어떻게 변경될지 확인하려면 -n 옵션을 사용합니다. 실제로 변경하는 것은 없습니다.

페이킷 값이 None으로 설정되어 있으면 페이킷 사양이 현재 이미지에서 제거됩니다.

다음 명령은 대량의 출력 결과를 생성하는데 이는 영향을 받는 패키지가 너무 많기 때문입니다. 사용될 추가 공간의 양이 메가바이트가 아니라 기가바이트 단위라는 점에 주목하십시오. 이 작업은 완료되는데 많은 시간이 필요할 수 있으며 이 이미지와 패키지 저장소 간에 대량의 네트워크 전송이 발생할 수 있습니다. 새 BE는 기본적으로 생성되지 않지만 대신 백업 BE가 생성됩니다. BE 생성 시점에 대한 자세한 내용은 [52 페이지 “부트 환경 정책 이미지 등록 정보”](#)를 참조하십시오.

```
# pkg change-facet -nv facet.locale.*=true
  Packages to update:      831
  Variants/Facets to change: 1
  Estimated space available: 112.19 GB
  Estimated space to be consumed: 2.96 GB
  Create boot environment: No
  Create backup boot environment: Yes
  Rebuild boot archive:    No
  Changed variants/facets:
    facet facet.locale.*: True
  Changed packages:
  solaris
  ...
```

## 이미지 업데이트

`pkg-fmri`를 지정하지 않고 또는 `pkg-fmri` 대신 별표(\*)와 함께 `pkg update` 명령을 사용하면 업데이트가 있는 설치된 모든 패키지를 설치된 패키지 및 게시자 구성에 따라 시스템에 적용된 제약 조건에서 허용하는 최신 버전으로 업데이트할 수 있습니다. 현재 이미지에 비전역 영역이 구성되어 있으면 이러한 영역도 업데이트됩니다. **Oracle Solaris 관리: Oracle Solaris Zones, Oracle Solaris 10 Zones 및 리소스 관리의 “패키지 및 영역 정보”**를 참조하십시오.

```
/usr/bin/pkg update [-fnvq] [-g path_or_uri ...] [--accept] [--licenses] [--no-index] [--no-refresh] [--no-be-act]
```

업데이트가 있는 패키지 목록을 표시하려면 `pkg list -u` 명령을 사용합니다. `pkg update` 작업을 실행하면 대개 새 BE가 자동으로 생성됩니다. 현재 이미지나 새 BE에서 어떤 변경이 이루어질지 검토하려면 `pkg update -nv`를 사용합니다. 새 BE가 강제로 생성되도록 설정하거나 새 BE에 의미 있는 이름을 지정하려면 `--be-name` 옵션을 사용합니다.

---

**참고** - 가장 안전한 설치 또는 업데이트 방법은 새 BE를 명시적으로 지정하는 것입니다. BE 생성 시점에 대한 자세한 내용은 [52 페이지 “부트 환경 정책 이미지 등록 정보”](#)를 참조하십시오.

---

기본적으로 각 패키지는 현재 설치된 버전을 제공한 게시자에서 업데이트됩니다. 게시자 고착성 및 검색 순서를 지정하여 패키지를 제공하는 게시자를 제어할 수 있습니다. [44 페이지 “패키지 게시자 추가, 수정 또는 제거”](#)를 참조하십시오.

설치된 모든 패키지를 업데이트할 때 `-f` 옵션을 지정하면 패키지가 최신인지 확인하는 클라이언트 검사가 실행되지 않습니다.

지정한 패키지 저장소나 패키지 아카이브를 이미지의 임시로 패키지 데이터를 검색할 소스 목록에 추가하려면 `-g` 옵션을 사용합니다. `update`를 실행한 후 게시자가 제공했지만 이미지에 없는 모든 패키지는 원본 없이 이미지 구성에 추가됩니다.

업데이트된 패키지의 사용권 계약에 동의함을 나타내려면 `--accept` 옵션을 사용합니다. 모든 패키지 라이선스에는 사용권 동의가 필요하기 때문에 이 옵션을 제공하지 않을 경우 업데이트 작업이 실패합니다. 이 작업의 일부로 업데이트되는 패키지의 모든 라이선스를 표시하려면 `--licenses` 옵션을 사용합니다.

`--no-refresh` 옵션을 지정하면 사용 가능한 패키지 및 기타 메타 데이터의 최신 목록을 검색하기 위해 이미지 게시자 저장소에 연결하지 않습니다.

`--no-index` 옵션을 지정하면 작업이 성공적으로 완료된 후 검색 색인은 업데이트되지 않습니다. 패키지를 다수 설치할 경우 이 옵션을 지정하면 상당 시간을 절약할 수 있습니다. 업데이트 작업이 완료되면 `pkg refresh`를 사용하여 사용 가능한 패키지 목록 및 지정한 각 패키지의 게시자 메타 데이터를 업데이트할 수 있습니다. 게시자를 지정하지 않으면 모든 게시자에 대해 새로 고침 작업이 수행됩니다. 새 BE가 생성될 경우 새 BE에서 이 게시자 새로 고침 작업을 수행하십시오.

## 이미지 및 게시자 등록 정보 구성

이미지 정책을 구현하려면 이미지 등록 정보를 설정하면 됩니다. 이 절에서는 이미지 및 게시자 등록 정보와 이러한 등록 정보를 설정하는 방법에 대해 설명합니다. 이미지 등록 정보에 대한 설명은 [pkg\(1\)](#) 매뉴얼 페이지에서 "Image Properties"를 참조하십시오.

## 부트 환경 정책 이미지 등록 정보

이미지는 IPS 패키지를 설치할 수 있으며 다른 IPS 작업을 수행할 수 있는 위치입니다.

BE(부트 환경)는 이미지의 부트 가능한 인스턴스입니다. 시스템에서 여러 BE를 유지 관리할 수 있으며 각 BE에서 서로 다른 소프트웨어 버전을 설치할 수 있습니다. 시스템을 부트할 때 시스템에 있는 BE 중 하나로 부트하는 옵션이 제공됩니다. 패키지 작업의 결과로 새 BE가 자동으로 생성될 수 있습니다. 새 BE를 명시적으로 만들 수도 있습니다. 새 BE가 생성되는지 여부는 이 절에 설명된 대로 이미지 정책에 따라 달라집니다.

기본적으로 다음 작업 중 하나를 수행하면 새 BE가 자동으로 생성됩니다.

- 일부 드라이버 및 기타 커널 구성 요소 같은 특정한 주요 시스템 패키지를 업데이트하는 경우 이와 같은 작업은 변형을 설치, 제거, 업데이트, 변경하거나 패치셋을 변경할 때 수행될 수 있습니다.  
대개 새 BE는 업데이트가 있는 모든 패키지를 업데이트하기 위해 `pkg update` 명령을 실행할 때 생성됩니다.
- `--be-name`, `--require-new-be`, `--backup-be-name`, `--require-backup-be` 등의 옵션을 지정하는 경우
- `be-policy` 이미지 정책을 `always-new`로 설정하는 경우. 이 정책에서는 모든 패키지 작업이 다음 부트 시 활성화되는 새 BE에서 수행됩니다.

새 BE가 생성되면 시스템에서 다음 단계를 수행합니다.

1. 현재 BE의 복제본을 만듭니다.

복제 BE에는 원본 BE의 주 루트 데이터 집합 아래에 있는 계층 구조 내 모든 항목이 포함됩니다. 공유된 파일 시스템은 루트 데이터 집합 아래에 없으므로 복제되지 않습니다. 대신 새 BE에서는 원래 공유 파일 시스템에 액세스합니다.

2. 복제본 BE에서 패키지를 업데이트하되, 현재 BE의 패키지는 업데이트하지 않습니다.

현재 BE에 비전역 영역이 구성되어 있는 경우 새 BE에서 이러한 기존 영역이 구성됩니다.

3. --no-be-activate를 지정하지 않을 경우 다음에 시스템을 부트할 때 새 BE를 기본 부트 옵션으로 설정합니다. 현재 BE는 대체 부트 옵션으로 유지됩니다.

새 BE가 필요하지만 새 BE를 만드는 데 필요한 공간이 부족하면 필요 없는 기존 BE를 삭제하면 됩니다. BE에 대한 자세한 내용은 [Oracle Solaris 11 부트 환경 만들기 및 관리](#)를 참조하십시오.

아래에 설명된 이미지 등록 정보 설정에 대한 자세한 내용은 [57 페이지 “이미지 등록 정보 설정”](#)을 참조하십시오.

**be-policy**

패키지 작업 중 부트 환경이 생성되는 시점을 지정합니다. 사용할 수 있는 값은 다음과 같습니다.

**default** 기본 BE 생성 정책(`create-backup`)을 적용합니다.

**always-new** 모든 패키지 작업을 다음 부트 시 활성화되는 새 BE에서 수행하므로 재부트가 필요합니다. 백업 BE는 명시적으로 요청하지 않으면 생성되지 않습니다.

이 정책은 가장 안전하지만 재부트하지 않으면 패키지를 추가할 수 없기 때문에 대부분의 사이트 요구 사항에 비해 다소 엄격합니다.

**create-backup**

재부트가 필요한 패키지 작업의 경우 이 정책은 다음 부트 시 활성화되는 새 BE를 만듭니다. 패키지가 수정되거나 커널에 영향을 줄 수 있는 콘텐츠가 설치되고 작업이 라이브 BE에 영향을 주게 되면 백업 BE가 생성되지만 활성 BE로 설정되지는 않습니다. 백업 BE를 명시적으로 요청할 수도 있습니다.

이 정책은 새로 설치한 소프트웨어가 시스템 불안정을 야기하는 경우에 한해 위험을 가져올 수 있습니다. 이와 같은 경우는 거의 발생하지 않지만 가능성은 있습니다.

**when-required**

재부트가 필요한 패키지 작업의 경우 이 정책은 다음 부트 시 활성화되는 새 BE를 만듭니다. 백업 BE는 명시적으로 요청하지 않으면 생성되지 않습니다.

이 정책은 많은 위험을 야기할 수 있습니다. 패키징 작업으로 인해 라이브 BE가 변경되어 추가 변경이 불가능하게 되었는데 최신 풀백 BE가 존재하지 않을 수도 있기 때문입니다.

## 패키지 서명에 필요한 등록 정보

서명된 패키지를 설치하는 경우 이 절에 설명된 이미지 등록 정보 및 게시자 등록 정보를 설정하여 패키지 서명을 확인합니다.

### 서명된 패키지에 필요한 이미지 등록 정보

서명된 패키지를 사용하려면 다음 이미지 등록 정보를 구성합니다.

#### signature-policy

이 등록 정보 값은 이미지에 패키지를 설치 및 업데이트하거나 이미지의 패키지를 수정 또는 확인할 때 매니페스트에 대해 수행될 검사를 결정합니다. 패키지에 적용되는 최종 정책은 이미지 정책과 게시자 정책의 조합에 따라 달라집니다. 두 정책을 조합할 경우 수준은 둘 중 더 엄격한 정책을 적용했을 때와 같습니다. 기본적으로 패키지 클라이언트는 인증서가 취소되었는지 여부를 확인하지 않습니다. 이러한 검사를 사용으로 설정하려면 `check-certificate-revocation` 이미지 등록 정보를 `true`로 설정합니다. 이 경우 클라이언트가 외부 웹 사이트에 연결해야 할 수도 있습니다. 사용할 수 있는 값은 다음과 같습니다.

<code>ignore</code>	모든 매니페스트에 대해 서명을 무시합니다.
<code>verify</code>	서명이 있는 모든 매니페스트가 유효하게 서명되었는지 확인하지만 설치된 모든 패키지에 대해 서명을 요구하지는 않습니다.  이것이 기본값입니다.
<code>require-signatures</code>	새로 설치된 모든 패키지가 유효한 서명을 적어도 하나 이상 포함하도록 요구합니다. 또한 <code>pkg fix</code> 및 <code>pkg verify</code> 명령은 설치된 패키지에 유효한 서명이 없을 경우 경고 메시지를 표시합니다.
<code>require-names</code>	<code>require-signatures</code> 와 동일한 요구 사항을 따르지만 <code>signature-required-names</code> 이미지 등록 정보에 나열되는 문자열이 서명의 트러스트 체인을 확인하는 데 사용되는 인증서의 공통 이름으로 나타나도록 요구합니다.

#### signature-required-names

이 등록 정보의 값은 패키지 서명을 검증하는 동안 인증서의 공통 이름으로 나타나야 할 이름 목록입니다.

#### trust-anchor-directory

이 등록 정보의 값은 이미지의 트러스트 앵커가 포함된 디렉토리의 경로 이름입니다. 이 경로는 이미지의 상대 경로입니다.

기본값은 `ignore`입니다.

## 서명된 패키지에 대한 게시자 등록 정보

특정 게시자의 서명된 패키지를 사용하려면 다음 게시자 등록 정보를 구성합니다.

### signature-policy

이 등록 정보는 지정한 게시자의 패키지에만 적용되는 점을 제외하면 signature-policy 이미지 등록 정보와 기능이 같습니다.

### signature-required-names

이 등록 정보는 지정한 게시자의 패키지에만 적용되는 점을 제외하면 signature-required-names 이미지 등록 정보와 기능이 같습니다.

## 패키지 서명 등록 정보 구성

이 이미지에 대해 패키지 서명 등록 정보를 구성하려면 set-property, add-property-value, remove-property-value 및 unset-property 하위 명령을 사용합니다.

set-publisher 하위 명령의 --set-property, --add-property-value, --remove-property-value 및 --unset-property 옵션을 사용하여 서명 정책과 특정 게시자에 필요한 이름을 지정할 수 있습니다.

다음 예에서는 모든 패키지에 서명을 요구하도록 이 이미지를 구성합니다. 또한 이 예에서는 "oracle.com"이라는 문자열이 트러스트 체인의 인증서 중 하나에 대해 공통 이름으로 나타나도록 요구합니다.

```
# pkg set-property signature-policy require-names oracle.com
```

다음 예에서는 서명된 모든 패키지를 확인하도록 이 이미지를 구성합니다.

```
# pkg set-property signature-policy verify
```

다음 예에서는 example.com이라는 게시자에서 설치된 모든 패키지에 대해 서명을 요구하도록 이 이미지를 구성합니다.

```
# pkg set-publisher --set-property signature-policy=require-signatures example.com
```

다음 예에서는 필요한 서명 이름을 추가합니다. 이 예에서는 서명의 트러스트 체인에서 유효한 것으로 나타나야 할 이미지의 공통 이름 목록에 trustedname이라는 문자열을 추가합니다.

```
# pkg add-property-value signature-require-names trustedname
```

다음 예에서는 필요한 서명 이름을 제거합니다. 이 예에서는 서명의 트러스트 체인에서 유효한 것으로 나타나야 할 이미지의 공통 이름 목록에서 trustedname이라는 문자열을 제거합니다.

```
# pkg remove-property-value signature-require-names trustedname
```

다음 예에서는 지정한 게시자에 대해 필요한 서명 이름을 추가합니다. 이 예에서는 서명의 트러스트 체인에서 유효한 것으로 나타나야 `example.com` 게시자의 공통 이름 목록에 `trustedname`이라는 문자열을 추가합니다.

```
# pkg set-publisher --add-property-value \
signature-require-names=trustedname example.com
```

## 추가 이미지 등록 정보

### ca-path

SSL 작업을 위해 CA 인증서가 보관된 디렉토리를 가리키는 경로 이름을 지정합니다. 이 디렉토리의 형식은 기본 SSL 구현에 따라 다릅니다. 인증된 CA 인증서를 보관하는 데 다른 위치를 사용하려면 다른 디렉토리를 가리키도록 이 값을 변경합니다. CA 디렉토리 요구 사항에 대한 자세한 내용은 `SSL_CTX_load_verify_locations(3openssl)`의 `CApath` 부분을 참조하십시오.

기본값은 `/etc/openssl/certs`입니다.

### check-certificate-revocation

이 등록 정보가 `True`로 설정되면 패키지 클라이언트는 인증서 발급 후 인증서가 취소되었는지 확인하기 위해 서명 확인에 사용되는 인증서의 CRL 배포 지점에 연결합니다.

기본값은 `False`입니다.

### flush-content-cache-on-success

이 등록 정보가 `True`로 설정되면 패키지 클라이언트는 설치 또는 업데이트 작업이 완료될 때 해당 콘텐츠 캐시에서 파일을 제거합니다. 업데이트 작업의 경우 소스 BE에서만 콘텐츠가 제거됩니다. 이 옵션을 변경하지 않을 경우 다음에 대상 BE에서 패키징 작업이 수행될 때 패키지 클라이언트는 해당 콘텐츠 캐시를 비웁니다.

이 등록 정보는 디스크 공간이 제한된 시스템에서 콘텐츠 캐시 크기를 작게 유지하는데 사용할 수 있습니다. 이 등록 정보는 작업이 완료되는 데 걸리는 시간을 늘릴 수 있습니다.

기본값은 `True`입니다.

### mirror-discovery

이 등록 정보는 mDNS 및 DNS-SD를 사용하여 링크-로컬 콘텐츠 미러를 검색하도록 클라이언트에 알려줍니다. 이 등록 정보가 `True`로 설정되면 클라이언트는 동적으로 검색하는 미러에서 패키지 콘텐츠를 다운로드합니다. mDNS를 통해 콘텐츠를 보급하는 미러를 실행하려면 `pkg.depotd(1m)`을 참조하십시오.

기본값은 `False`입니다.

### send-uuid

네트워크 작업을 수행할 때 이미지의 UUID(Universally Unique Identifier)를 보냅니다. 사용자가 이 옵션을 사용 안함으로 설정할 수는 있지만 일부 네트워크 저장소의 경우 UUID를 제공하지 않는 클라이언트와의 통신을 거부할 수도 있습니다.

기본값은 True입니다.

### use-system-repo

이 등록 정보는 이미지가 이미지 및 게시자 구성의 소스로, 그리고 제공된 게시자와의 통신을 위한 프록시로 시스템 저장소를 사용해야 하는지 여부를 나타냅니다. 시스템 저장소에 대한 자세한 내용은 `pkg.sysrepo(1m)`을 참조하십시오.

기본값은 ignore입니다.

## 이미지 등록 정보 설정

이 이미지의 등록 정보를 구성하려면 `set-property`, `add-property-value`, `remove-property-value` 및 `unset-property` 하위 명령을 사용합니다.

```
/usr/bin/pkg property [-H] [propname ...] /usr/bin/pkg set-property propname propvalue /usr/bin/pkg add-property
```

## 이미지 등록 정보 값 표시

`pkg property` 명령을 사용하여 이미지 등록 정보를 볼 수 있습니다.

```
$ pkg property
PROPERTY                                VALUE
be-policy                                default
ca-path                                  /etc/openssl/certs
check-certificate-revocation             False
display-copyrights                       True
flush-content-cache-on-success           False
mirror-discovery                          False
preferred-authority                       solaris
publisher-search-order                   ['solaris', 'opensolaris.org', 'extra']
pursue-latest                             True
send-uuid                                 True
signature-policy                          verify
signature-required-names                  []
trust-anchor-directory                    etc/certs/CA
use-system-repo                           False
```

`preferred-authority` 및 `publisher-search-order` 등록 정보는 `pkg set-publisher` 명령 옵션을 사용하여 설정할 수 있습니다. 44 페이지 “패키지 게시자 추가, 수정 또는 제거”를 참조하십시오.

## 이미지 등록 정보 값 설정

`pkg set-property` 명령을 사용하여 이미지 등록 정보 값을 설정하거나 등록 정보를 추가 및 설정할 수 있습니다.

다음 예에서는 `mirror-discovery` 등록 정보의 값을 설정합니다.

```
# pkg set-property mirror-discovery True
# pkg property -H mirror-discovery
mirror-discovery True
```

## 이미지 등록 정보 값 재설정

`pkg unset-property` 명령을 사용하여 지정한 등록 정보의 값을 기본값으로 재설정할 수 있습니다.

```
# pkg unset-property mirror-discovery
$ pkg property -H mirror-discovery
mirror-discovery False
```

# 이미지 만들기

이미지는 IPS 패키지 및 관련 파일, 디렉토리, 링크 및 종속성을 설치할 수 있으며 다른 IPS 작업을 수행할 수 있는 위치입니다.

이미지는 다음 세 유형 중 하나일 수 있습니다.

- 전체 시스템을 제공할 수 있는 전체 이미지. 전체 이미지에서는 모든 종속성이 이미지 자체 내에서 확인되고 IPS가 일관된 방식으로 종속성을 유지 관리합니다. Oracle Solaris OS 설치를 완료하면 루트 파일 시스템과 해당 콘텐츠가 전체 이미지에 포함됩니다.
- 전체 이미지(부모 이미지)에 연결되어 있지만 자체 내에서 전체 시스템을 제공하지 않는 부분 이미지 비전역 영역은 부분 이미지입니다. `-z` 또는 `--zone` 옵션을 사용하여 적절한 변형을 설정합니다. 영역 이미지에서는 IPS가 패키지의 종속성에 정의된 대로, 비전역 영역이 해당 전역 영역과 일치하도록 유지 관리합니다. 비전역 영역에 대한 자세한 내용은 [Oracle Solaris 관리: Oracle Solaris Zones, Oracle Solaris 10 Zones 및 리소스 관리의 제II부](#), “Oracle Solaris Zones”을 참조하십시오.
- 위치 조정이 가능한 패키지만 포함하는 사용자 이미지

```
/usr/bin/pkg image-create [-FPufz] [--force] [--full | --partial | --user] [--zone] [-k ssl_key] [-c ssl_cert] [--no-r
```

`dir`에 지정한 위치에 패키지 작업에 적합한 이미지를 만듭니다. 기본 이미지 유형은 사용자(`-U` 또는 `-user`)입니다. 이미지 유형은 전체 이미지(`-F` 또는 `--full`)로 설정하거나, 지정한 `dir` 경로를 포함하는 전체 이미지에 연결된 부분 이미지(`-P` 또는 `--partial`)로 설정할 수 있습니다.

새 이미지를 비전역 영역 컨텍스트에서 실행하려면 `-z` 또는 `--zone` 옵션을 사용하여 적절한 변형을 설정합니다.

패키지 저장소 URI는 `-p` 또는 `--publisher` 옵션을 사용하여 제공해야 합니다. 게시자 이름도 제공한 경우 이미지가 생성될 때 해당 게시자만 추가됩니다. 게시자 이름을 제공하지 않은 경우 지정한 저장소에서 인식하는 모든 게시자가 이미지에 추가됩니다. 초기 생성 작업이 끝나면 이 게시자와 관련된 카탈로그가 검색됩니다.

추가 원본을 지정하려면 `-g` 옵션을 사용합니다. 미러를 지정하려면 `-m` 옵션을 사용합니다.

클라이언트 SSL 인증을 사용하는 게시자의 경우 `-c` 또는 `-k` 옵션을 사용하여 클라이언트 키와 클라이언트 인증서를 등록합니다. 이 키와 인증서는 이미지 생성 중 추가된 모든 게시자에 대해 사용됩니다.

기존 이미지를 대체하는 이미지를 강제로 만들려면 `-f` 옵션을 사용합니다. 이 옵션은 주의해서 사용해야 합니다.

`-no-refresh` 옵션을 지정하면 사용 가능한 패키지 및 기타 메타 데이터의 최신 목록을 검색하기 위해 이미지 게시자 저장소에 연결하지 않습니다.

지정한 변형을 표시된 값으로 설정하려면 `--variant` 옵션을 사용합니다. 지정한 페이스를 표시된 값으로 설정하려면 `--facet` 옵션을 사용합니다.

## 작업 기록 보기

`pkg history` 명령을 사용하여 현재 이미지의 명령 기록을 볼 수 있습니다.

```
/usr/bin/pkg history [-HL] [-t [time | time-time],...] [-o column,...] [-n number]
```

명령의 결과, 명령이 완료된 시간, 사용된 클라이언트의 버전과 이름, 작업을 수행한 사용자의 이름, 명령 실행 중 발생한 오류 등의 추가 정보를 표시하려면 `-l` 옵션을 사용합니다.

지정한 개수만큼의 최근 작업만 표시하려면 `-n` 옵션을 사용합니다.

```
$ pkg history -n4
```

START	OPERATION	CLIENT	OUTCOME
2011-09-07T12:15:52	update	pkg	Succeeded
2011-09-26T18:53:12	refresh-publishers	pkg	Succeeded
2011-09-26T18:53:50	rebuild-image-catalogs	pkg	Succeeded
2011-09-27T09:05:34	update	pkg	Succeeded

지정한 선택포로 구분된 열 이름 목록을 사용하여 출력 결과를 표시하려면 `-o` 옵션을 사용합니다. `pkg(1)`의 열 이름 목록을 참조하십시오.

```
# pkg history -o start,time,operation,outcome -n4
```

START	TIME	OPERATION	OUTCOME
2011-09-07T12:15:52	0:13:56	update	Succeeded
2011-09-26T18:53:12	0:01:22	refresh-publishers	Succeeded
2011-09-26T18:53:50	0:00:44	rebuild-image-catalogs	Succeeded
2011-09-27T09:05:34	0:20:08	update	Succeeded

%Y-%m-%dT%H:%M:%S 형식의 쉘표로 구분된 시간 기록 목록(strftime(3C) 참조)에 대한 레코드를 기록하려면 **-t** 옵션을 사용합니다. 시간 범위를 지정하려면 시작 시간 기록과 끝 시간 기록 사이에 하이픈(-)을 사용합니다. 현재 시간에 대한 별칭으로 키워드 **now**를 사용할 수 있습니다. 지정한 시간 기록에 중복되는 시간 기록이나 날짜 범위가 포함되어 있으면 각 중복 기록 이벤트의 한 인스턴스만 출력됩니다.

명령 기록 정보를 모두 삭제하려면 **pkg purge-history** 명령을 사용합니다.

**# pkg purge-history**