

시스템 관리 설명서: 고급 관리

Copyright © 1998, 2011, Oracle and/or its affiliates. All rights reserved.

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 소프트웨어를 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	15
1 터미널 및 모뎀 관리(개요)	19
터미널 및 모뎀 관리의 새로운 기능	19
SPARC: 코히어런트(Coherent) 콘솔	19
SPARC: 콘솔에 대한 \$TERM 값 설정 방법 변경	20
SMF로 관리되는 시스템 콘솔에 대한 ttymon 호출	20
터미널, 모뎀, 포트 및 서비스	21
터미널 설명	21
모뎀 설명	21
포트 설명	21
서비스 설명	22
포트 모니터	22
터미널 및 모뎀 관리를 위한 도구	23
직렬 포트 도구	23
Service Access Facility 개요	24
2 터미널 및 모뎀 설정(작업)	25
터미널 및 모뎀 설정(작업 맵)	25
직렬 포트 도구를 사용하여 터미널 및 모뎀 설정(개요)	26
터미널 설정	26
모뎀 설정	27
터미널 및 모뎀을 설정하고 포트를 초기화하는 방법(작업)	29
▼ 터미널을 설정하는 방법	29
▼ 모뎀을 설정하는 방법	29
▼ 포트를 초기화하는 방법	30
터미널 및 모뎀 문제 해결	31

3 Service Access Facility를 사용하여 직렬 포트 관리(작업)	33
직렬 포트 관리(작업 맵)	34
Service Access Facility 사용	34
전반적인 SAF 관리(sacadm)	35
Service Access Controller(SAC 프로그램)	35
SAC 초기화 프로세스	36
포트 모니터 서비스 관리(pmadm)	36
ttymon 포트 모니터	36
포트 초기화 프로세스	37
양방향 서비스	37
TTY 모니터 및 네트워크 Listener 포트 모니터	38
TTY 포트 모니터(ttymon)	38
ttymon 및 콘솔 포트	38
ttymon 관련 관리 명령(ttyadm)	39
네트워크 Listener 서비스(listen)	39
특수 listen 관련 관리 명령(nlsadmin)	39
ttymon 포트 모니터 관리	40
▼ ttymon 콘솔 터미널 유형을 설정하는 방법	40
▼ ttymon 콘솔 터미널에 대한 변조 속도를 설정하는 방법	41
▼ ttymon 포트 모니터를 추가하는 방법	42
▼ ttymon 포트 모니터 상태를 확인하는 방법	42
▼ ttymon 포트 모니터를 중지하는 방법	43
▼ ttymon 포트 모니터를 시작하는 방법	43
▼ ttymon 포트 모니터를 사용 안함으로 설정하는 방법	44
▼ ttymon 포트 모니터를 사용으로 설정하는 방법	44
▼ ttymon 포트 모니터를 제거하는 방법	44
ttymon 서비스 관리(작업 맵)	45
ttymon 서비스 관리	45
▼ 서비스를 추가하는 방법	46
▼ TTY 포트 서비스의 상태를 확인하는 방법	46
▼ 포트 모니터 서비스를 사용으로 설정하는 방법	49
▼ 포트 모니터 서비스를 사용 안함으로 설정하는 방법	49
Service Access Facility 관리(참조)	49
SAF와 연관된 파일	49
/etc/saf/_sactab 파일	50
/etc/saf/pmtab/_pmtab 파일	51

서비스 상태	52
포트 모니터 상태	52
포트 상태	53
4 시스템 리소스 관리(개요)	55
시스템 리소스 관리의 새로운 기능	55
제품 이름을 표시하는 새로운 prtconf 옵션	55
칩 다중 스레드 기능을 확인하는 psrinfo 명령 옵션	56
새로운 localeadm 명령	56
시스템 리소스 관리(로드맵)	57
5 시스템 정보 표시 및 변경(작업)	59
시스템 정보 표시(작업 맵)	59
시스템 정보 표시	60
▼시스템의 물리적 프로세서 유형 표시 방법	66
▼시스템의 논리적 프로세서 유형 표시 방법	66
▼시스템에 설치된 로케일 표시 방법	67
▼시스템에 로케일이 설치되었는지 여부 확인 방법	68
시스템 정보 변경(작업 맵)	68
시스템 정보 변경	69
▼시스템의 날짜 및 시간을 수동으로 설정하는 방법	69
▼오늘의 메시지 설정 방법	70
▼시스템의 호스트 이름 변경 방법	71
▼시스템에 로케일을 추가하는 방법	72
▼시스템에서 로케일을 제거하는 방법	72
6 디스크 사용 관리(작업)	75
디스크 사용 관리(작업 맵)	75
파일 및 디스크 공간 정보 표시	76
▼파일 및 디스크 공간 정보 표시 방법	77
파일 크기 확인	79
▼파일 크기 표시 방법	79
▼큰 파일을 찾는 방법	80
▼지정된 크기 제한을 초과하는 파일을 찾는 방법	81

디렉토리 크기 확인	82
▼ 디렉토리, 하위 디렉토리 및 파일의 크기 표시 방법	82
▼ 로컬 UFS 파일 시스템의 사용자 소유권 표시 방법	83
오래된 파일 또는 비활성 파일 찾기 및 제거	84
▼ 최신 파일 나열 방법	85
▼ 오래된 파일 또는 비활성 파일 찾기 및 제거 방법	85
▼ 임시 디렉토리를 지우는 방법	86
▼ 코어 파일 찾기 및 삭제 방법	87
▼ 충돌 덤프 파일 삭제 방법	88
7 UFS 쿼터 관리(작업)	89
UFS 쿼터란?	89
UFS 쿼터 사용	89
UFS 쿼터에 대한 소프트 제한 및 하드 제한 설정	90
디스크 블록과 파일 제한의 차이	90
UFS 쿼터 설정	90
UFS 쿼터 설정 지침	91
UFS 쿼터 설정(작업 맵)	92
▼ 파일 시스템에서 UFS 쿼터 구성 방법	92
▼ 사용자에게 대한 UFS 쿼터 설정 방법	93
▼ 다중 사용자에게 대한 UFS 쿼터 설정 방법	94
▼ UFS 쿼터 일관성 확인 방법	94
▼ UFS 쿼터를 켜는 방법	95
UFS 쿼터 유지 관리(작업 맵)	96
UFS 쿼터 확인	97
▼ UFS 쿼터 초과 확인 방법	97
▼ 파일 시스템에 대한 UFS 쿼터 확인 방법	98
UFS 쿼터 변경 및 제거	99
▼ 소프트 제한 기본값 변경 방법	99
▼ 사용자에게 대한 UFS 쿼터 변경 방법	100
▼ 사용자에게 대한 UFS 쿼터를 사용 안함으로 설정하는 방법	102
▼ UFS 쿼터를 끄는 방법	103
8 시스템 작업 예약(작업)	105
crontab 파일 만들기 및 편집(작업 맵)	105

시스템 작업을 자동으로 실행하는 방법	106
반복적 작업 예약:crontab	106
단일 작업 예약:at	107
반복적 시스템 작업 예약(cron)	108
crontab 파일 내부	108
cron 데몬이 예약을 처리하는 방법	109
crontab 파일 항목의 구문	109
crontab 파일 만들기 및 편집	110
▼ crontab 파일을 만들거나 편집하는 방법	111
▼ crontab 파일이 존재하는지 확인하는 방법	112
crontab 파일 표시	112
▼ crontab 파일을 표시하는 방법	112
crontab 파일 제거	113
▼ crontab 파일을 제거하는 방법	113
crontab 명령에 대한 액세스 제어	114
▼ crontab 명령 액세스를 거부하는 방법	115
▼ crontab 명령 액세스를 지정된 사용자로 제한하는 방법	116
제한된 crontab 명령 액세스를 확인하는 방법	117
at 명령 사용(작업 맵)	117
단일 시스템 작업 예약(at)	118
at 명령의 설명	118
at 명령에 대한 액세스 제어	119
▼ at 작업을 만드는 방법	119
▼ at 대기열을 표시하는 방법	120
▼ at 작업을 확인하는 방법	120
▼ at 작업을 표시하는 방법	120
▼ at 작업을 제거하는 방법	121
▼ at 명령에 대한 액세스를 거부하는 방법	122
▼ at 명령 액세스가 거부되었는지 확인하는 방법	123
9 시스템 계산 관리(작업)	125
시스템 계산의 새로운 기능	125
Oracle Solaris 프로세스 계산 및 통계 개선	125
시스템 계산이란?	126
시스템 계산의 작동 원리	126

시스템 계산 구성 요소	127
시스템 계산(작업 맵)	130
시스템 계산 설정	131
▼ 시스템 계산을 설정하는 방법	132
사용자 요금 청구	133
▼ 사용자 요금을 청구하는 방법	134
계산 정보 유지 관리	134
훼손된 파일 및 wtmpx 오류 수정	134
▼ 훼손된 wtmpx 파일을 수정하는 방법	135
tacct 오류 수정	135
▼ tacct 오류를 수정하는 방법	135
runacct 스크립트 다시 시작	136
▼ runacct 스크립트를 다시 시작하는 방법	136
시스템 계산 중지 및 사용 안함	137
▼ 시스템 계산을 일시적으로 중지하는 방법	137
▼ 시스템 계산을 영구적으로 사용 안함으로 설정하는 방법	138
10 시스템 계산(참조)	139
runacct 스크립트	139
일별 계산 보고서	141
일별 보고서	142
일별 사용량 보고서	143
일별 명령 요약	144
월별 명령 요약	145
마지막 로그인 보고서	146
acctcom으로 pacct 파일 검사	146
시스템 계산 파일	148
runacct 스크립트로 생성된 파일	150
11 시스템 성능 관리(개요)	153
시스템 성능 관리의 새로운 기능	153
향상된 pfiles 도구	153
CPU 성능 카운터	154
시스템 성능 작업의 위치	154
시스템 성능 및 시스템 리소스	155

프로세스 및 시스템 성능	155
시스템 성능 모니터링 정보	157
모니터링 도구	157
12 시스템 프로세스 관리(작업)	159
시스템 프로세스 관리의 새로운 기능	159
의사 커널 프로세스	159
시스템 프로세스 관리(작업 맵)	160
시스템 프로세스 관리 명령	161
ps 명령 사용	161
/proc 파일 시스템 및 명령 사용	163
프로세스 명령을 사용하여 프로세스 관리(/proc)	164
▼ 프로세스를 나열하는 방법	164
▼ 프로세스에 대한 정보를 표시하는 방법	165
▼ 프로세스를 제어하는 방법	166
프로세스 종료(pkill, kill)	167
▼ 프로세스를 종료하는 방법(pkill)	167
▼ 프로세스를 종료하는 방법(kill)	168
프로세스 디버깅(pargs, preap)	169
프로세스 클래스 정보 관리(작업 맵)	170
프로세스 클래스 정보 관리	171
프로세스의 예약 우선 순위 변경(priocntl)	171
▼ 프로세스 클래스에 대한 기본 정보를 표시하는 방법(priocntl)	172
▼ 프로세스의 전역 우선 순위를 표시하는 방법	172
▼ 프로세스 우선 순위를 지정하는 방법(priocntl)	173
▼ 시간 공유 프로세스의 예약 매개변수를 변경하는 방법(priocntl)	173
▼ 프로세스의 클래스를 변경하는 방법(priocntl)	174
시간 공유 프로세스의 우선 순위 변경(nice)	175
▼ 프로세스의 우선 순위를 변경하는 방법(nice)	176
시스템 프로세스의 문제 해결	177
13 시스템 성능 모니터링(작업)	179
시스템 성능 정보 표시(작업 맵)	179
가상 메모리 통계 표시(vmstat)	180
▼ 가상 메모리 통계를 표시하는 방법(vmstat)	181

▼ 시스템 이벤트 정보를 표시하는 방법(vmstat -s)	182
▼ 스와핑 통계를 표시하는 방법(vmstat -S)	182
▼ 장치당 인터럽트를 표시하는 방법(vmstat -i)	183
디스크 사용률 정보 표시(iostat)	183
▼ 디스크 사용률 정보를 표시하는 방법(iostat)	183
▼ 확장 디스크 통계를 표시하는 방법(iostat -xtc)	185
디스크 공간 통계 표시(df)	185
▼ 디스크 공간 정보를 표시하는 방법(df -k)	186
시스템 작업 모니터링(작업 맵)	187
시스템 작업 모니터링(sar)	188
▼ 파일 액세스를 확인하는 방법(sar -a)	188
▼ 버퍼 작업을 확인하는 방법(sar -b)	189
▼ 시스템 호출 통계를 확인하는 방법(sar -c)	191
▼ 디스크 작업을 확인하는 방법(sar -d)	192
▼ 페이지 아웃 및 메모리를 확인하는 방법(sar -g)	193
커널 메모리 할당 확인	194
▼ 커널 메모리 할당을 확인하는 방법(sar -k)	195
▼ 프로세스간 통신을 확인하는 방법(sar -m)	196
▼ 페이지 인 작업을 확인하는 방법(sar -p)	197
▼ 대기열 작업을 확인하는 방법(sar -q)	198
▼ 사용되지 않은 메모리를 확인하는 방법(sar -r)	199
▼ CPU 사용률을 확인하는 방법(sar -u)	200
▼ 시스템 테이블 상태를 확인하는 방법(sar -v)	201
▼ 스와핑 작업을 확인하는 방법(sar -w)	202
▼ 터미널 작업을 확인하는 방법(sar -y)	203
▼ 전체 시스템 성능을 확인하는 방법(sar -A)	204
자동으로 시스템 작업 데이터 수집(sar)	204
부트할 때 sadc 명령 실행	205
sa1 스크립트를 사용하여 정기적으로 sadc 명령 실행	205
sa2 셸 스크립트를 사용하여 보고서 생성	206
자동 데이터 수집 설정(sar)	206
▼ 자동 데이터 수집을 설정하는 방법	207
14 소프트웨어 문제 해결(개요)	209
문제 해결의 새로운 정보	209

공통 에이전트 컨테이너 문제	209
x86: 시스템 재부트 중 SMF 부트 아카이브 서비스가 실패할 수 있음	210
동적 추적 기능	210
표준 Solaris 커널 디버거로 kmdb가 kadb를 대체함	210
소프트웨어 문제 해결 작업 검색 위치	211
시스템 충돌 문제 해결	211
시스템이 충돌할 경우 수행할 작업	211
문제 해결 데이터 수집	212
시스템 충돌 문제 해결 점검 목록	213
15 시스템 메시지 관리	215
시스템 메시지 확인	215
▼ 시스템 메시지 확인 방법	216
시스템 로그 교체	217
시스템 메시지 로깅 사용자 정의	218
▼ 시스템 메시지 로깅 사용자 정의 방법	219
원격 콘솔 메시지를 사용으로 설정	220
실행 레벨 전환 중 보조 콘솔 메시지 사용	221
대화형 로그인 세션 중 consadm 명령 사용	222
▼ 보조(원격) 콘솔을 사용으로 설정하는 방법	222
▼ 보조 콘솔 목록 표시 방법	223
▼ 시스템 재부트 시 보조(원격) 콘솔을 사용으로 설정하는 방법	223
▼ 보조(원격) 콘솔을 사용 안함으로 설정하는 방법	224
16 코어 파일 관리(작업)	225
코어 파일 관리(작업 맵)	225
코어 파일 관리 개요	226
구성 가능한 코어 파일 경로	226
확장된 코어 파일 이름	226
코어 파일 이름 패턴 설정	227
코어 파일을 생성하도록 setuid 프로그램을 사용으로 설정	228
현재 코어 덤프 구성 표시 방법	228
▼ 코어 파일 이름 패턴 설정 방법	229
▼ 프로세스별 코어 파일 경로를 사용으로 설정하는 방법	229
▼ 전역 코어 파일 경로를 사용으로 설정하는 방법	229

코어 파일 문제 해결	230
코어 파일 검사	230
17 시스템 충돌 정보 관리(작업)	231
시스템 충돌 정보 관리의 새로운 기능	231
빠른 충돌 덤프 기능	231
시스템 충돌 정보 관리(작업 맵)	232
시스템 충돌(개요)	232
스왑 영역 및 덤프 장치에 대한 Oracle Solaris ZFS 지원	233
x86: GRUB 부트 환경의 시스템 충돌	233
시스템 충돌 덤프 파일	233
충돌 덤프 저장	234
dumpadm 명령	234
dumpadm 명령 작동 방식	235
덤프 장치 및 볼륨 관리자	235
시스템 충돌 덤프 정보 관리	235
▼ 현재 충돌 덤프 구성 표시 방법	236
▼ 충돌 덤프 구성 수정 방법	236
▼ 충돌 덤프 검사 방법	238
▼ 전체 충돌 덤프 디렉토리에 복구 방법(선택 사항)	239
▼ 충돌 덤프 저장을 사용/사용 안함으로 설정하는 방법	239
18 기타 소프트웨어 문제 해결(작업)	241
재부트를 실패할 경우 수행할 작업	241
루트 암호를 잊은 경우 수행할 작업	242
x86: 시스템 재부트 중 SMF 부트 아카이브 서비스가 실패할 경우 수행할 작업	245
시스템이 정지될 경우 수행할 작업	246
파일 시스템이 가득 찬 경우 수행할 작업	247
큰 파일 또는 디렉토리가 만들어져 파일 시스템이 가득 참	247
시스템 메모리 부족으로 인해 TMPFS 파일 시스템이 가득 참	248
복사 또는 복원 후 파일 ACL이 손실된 경우 수행할 작업	248
백업 문제 해결	248
파일 시스템 백업 후 루트(/) 파일 시스템이 가득 참	248
백업 명령과 복원 명령은 동일해야 함	249
적합한 현재 디렉토리를 사용하는지 확인	249

대화형 명령	249
Oracle Solaris OS에서 공통 에이전트 컨테이너 문제 해결	249
포트 번호 충돌	250
▼ 포트 번호 확인 방법	250
수퍼유저 암호 보안 손상	250
▼ Oracle Solaris OS용 보안 키 생성 방법	251
19 파일 액세스 문제 해결(작업)	253
검색 경로 문제 해결(Command not found)	253
▼ 검색 경로 문제 진단 및 해결 방법	254
파일 액세스 문제 해결	255
파일 및 그룹 소유권 변경	256
네트워크 액세스 문제 인식	256
20 UFS 파일 시스템 불일치 해결(작업)	257
fsck 오류 메시지	258
일반 fsck 오류 메시지	259
초기화 단계 fsck 메시지	261
1단계: 블록 및 크기 메시지 확인	264
Oracle Solaris 10: 1B단계: 추가 DUPS 메시지 재검색	268
1B단계: 추가 DUPS 메시지 재검색	269
2단계: 경로 이름 메시지 확인	269
3단계: 연결 메시지 확인	276
4단계: 참조 수 메시지 확인	278
5단계: 실린더 그룹 메시지 확인	281
5단계: 실린더 그룹 메시지 확인	282
fsck 요약 메시지	283
정리 단계 메시지	284
21 소프트웨어 패키지 문제 해결(작업)	285
소프트웨어 패키지 심볼릭 링크 문제 해결	285
특정 소프트웨어 패키지 설치 오류	286
일반 소프트웨어 패키지 설치 문제	286

색인 289

머리말

시스템 관리 설명서: 고급 관리는 Oracle Solaris 시스템 관리 정보의 중요한 부분을 다루는 설명서 모음의 일부입니다. 본 설명서에서는 SPARC 기반 시스템과 x86 기반 시스템에 대한 정보를 모두 다룹니다.

본 설명서에서는 Oracle Solaris OS(운영 체제)가 설치된 것으로 가정합니다. 또한 사용하려는 네트워킹 소프트웨어가 설정된 것으로 가정합니다.

Oracle Solaris 릴리스의 경우 시스템 관리자를 대상으로 한 새로운 기능이 해당 장의 **...의 새로운 기능**이라는 단원에서 설명됩니다.

주 - 본 Oracle Solaris 릴리스는 프로세서 아키텍처의 SPARC 및 x86 제품군을 사용하는 시스템을 지원합니다. 지원되는 시스템은 **Oracle Solaris OS: 하드웨어 호환성 목록**을 참조하십시오. 이 설명서에서는 플랫폼 유형에 따른 구현 차이가 있는 경우 이에 대하여 설명합니다.

이 문서에서 사용되는 x86 관련 용어의 의미는 다음과 같습니다.

- x86은 64비트 및 32비트 x86 호환 제품을 아우르는 큰 제품군을 의미합니다.
- x64는 특히 64비트 x86 호환 CPU와 관련됩니다.
- "32비트 x86"은 x86 기반 시스템에 대한 특정 32비트 정보를 나타냅니다.

지원되는 시스템은 **Oracle Solaris OS: 하드웨어 호환성 목록**을 참조하십시오.

이 설명서의 대상

본 설명서는 Oracle Solaris 10이 실행되고 있는 한 대 이상의 시스템을 관리하는 사용자를 대상으로 작성되었습니다. 본 설명서를 사용하려면 1-2년 정도의 UNIX 시스템 관리 경험이 있어야 합니다. UNIX 시스템 관리 교육 과정에 참석하는 것도 도움이 될 수 있습니다.

시스템 관리 설명서의 구성

시스템 관리 설명서에서 설명하는 항목 목록은 다음과 같습니다.

설명서 제목	내용
시스템 관리 설명서: 기본 관리	사용자 계정 및 그룹, 서버 및 클라이언트 지원, 시스템 종료 및 부팅, 서비스 관리, 소프트웨어 관리(패키지 및 패치)
시스템 관리 설명서: 고급 관리	터미널 및 모뎀, 시스템 리소스(디스크 쿼터, 계정, 크론탐), 시스템 프로세스, Oracle Solaris 소프트웨어 문제 해결
System Administration Guide: Devices and File Systems	이동식 매체, 디스크 및 장치, 파일 시스템, 데이터 백업 및 복원
System Administration Guide: IP Services	TCP/IP 네트워크 관리, IPv4 및 IPv6 주소 관리, DHCP, IPsec, IKE, Solaris IP 필터, 이동 IP, IPMP(IP Network Multipathing), IPQoS
System Administration Guide: Naming and Directory Services (DNS, NIS, and LDAP)	DNS, NIS 및 LDAP 명명 규칙 및 디렉토리 서비스(NIS에서 LDAP으로의 전환, NIS+에서 LDAP으로의 전환 포함)
System Administration Guide: Naming and Directory Services (NIS+)	NIS+ 이름 지정 및 디렉토리 서비스
System Administration Guide: Network Services	웹 캐시 서버, 시간 관련 서비스, 네트워크 파일 시스템(NFS 및 Autofs), 메일, SLP, PPP
System Administration Guide: Printing	인쇄 항목 및 작업, 서비스, 도구, 프로토콜 및 기술을 사용하여 인쇄 서비스와 프린터 설정 및 관리
System Administration Guide: Security Services	감사, 장치 관리, 파일 보안, BART, Kerberos 서비스, PAM, Solaris Cryptographic Framework, 권한, RBAC, SASL, Solaris Secure Shell
시스템 관리 설명서: Oracle Solaris Containers-리소스 관리 및 Oracle Solaris 영역	리소스 관리 항목 프로젝트 및 작업, 확장 계정, 리소스 제어, FSS(Fair Share Scheduler), rcapd(Resource Capping Daemon)를 통한 물리적 메모리 제어, 리소스 풀, Solaris Zones 소프트웨어 분할 기술 및 ix 브랜드 영역을 통한 가상화
Oracle Solaris ZFS 관리 설명서	ZFS 저장소 풀 및 파일 시스템 만들기/관리, 스냅샷, 복제, 백업, 액세스 제어 목록(ACL)을 통한 ZFS 파일 보호, 영역이 설치된 Oracle Solaris 시스템에서 ZFS 사용, 애플리케이션 불륨, 문제 해결 및 데이터 복구
Oracle Solaris Trusted Extensions 관리자 절차	Oracle Solaris의 Trusted Extensions 기능에만 적용되는 시스템 관리
Oracle Solaris Trusted Extensions 구성 설명서	Solaris 10 5/08 릴리스부터 Oracle Solaris의 Trusted Extensions 기능을 계획, 사용으로 설정 및 처음 구성하는 방법에 대해 설명합니다.

타사 웹 사이트

주 - Oracle은 본 설명서에서 언급된 타사 웹 사이트의 가용성에 대해 책임을 지지 않습니다. 또한 해당 사이트나 리소스를 통해 제공되는 내용, 알림, 제품 및 기타 자료에 대해 어떠한 보증도 하지 않으며 그에 대한 책임도 지지 않습니다. 따라서 타사 웹 사이트나 자원을 통해 제공되는 내용, 제품 또는 서비스의 사용으로 인해 발생한 실제 또는 주장된 손상이나 피해에 대해서도 책임을 지지 않습니다.

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	의미	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	의미	예
AaBbCc123	책 제목, 장, 절	<p>사용자 설명서의 6장을 읽으십시오.</p> <p>캐시는 로컬로 저장된 복사본입니다.</p> <p>파일을 저장하면 안 됩니다.</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#

일반 규칙

본 설명서에서 사용되는 다음 규칙을 이해해야 합니다.

- 단계를 따르거나 예제를 사용할 때는 큰 따옴표("), 왼쪽 작은 따옴표(') 및 오른쪽 작은 따옴표(')를 표시된 대로 정확히 입력해야 합니다.
- Return으로 표시된 키는 키보드에 따라 Enter로 표시될 수 있습니다.
- 루트 경로에는 /sbin, /usr/sbin, /usr/bin 및 /etc 디렉토리가 포함된 것으로 가정되므로 본 설명서의 단계에서는 해당 디렉토리의 명령이 절대 경로 이름 없이 표시됩니다. 다른 특수한 디렉토리에서 명령을 사용하는 단계의 경우 예제에 절대 경로가 표시됩니다.

터미널 및 모뎀 관리(개요)

이 장은 터미널 및 모뎀 관리에 대한 개요 정보를 제공합니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 19 페이지 “터미널 및 모뎀 관리의 새로운 기능”
- 21 페이지 “터미널, 모뎀, 포트 및 서비스”
- 23 페이지 “터미널 및 모뎀 관리를 위한 도구”
- 23 페이지 “직렬 포트 도구”
- 24 페이지 “Service Access Facility 개요”

직렬 포트 도구를 사용하여 터미널 및 모뎀을 설정하는 방법에 대한 단계별 지침은 2 장, “터미널 및 모뎀 설정(작업)”을 참조하십시오.

SAF(Service Access Facility)를 사용하여 터미널 및 모뎀을 설정하는 방법에 대한 단계별 지침은 3 장, “Service Access Facility를 사용하여 직렬 포트 관리(작업)”를 참조하십시오.

터미널 및 모뎀 관리의 새로운 기능

이 단원에서는 Oracle Solaris 릴리스에서 터미널 및 모뎀을 관리하기 위해 새로 추가되거나 변경된 기능에 대해 설명합니다. 새로운 기능의 전체 목록 및 Oracle Solaris 릴리스에 대한 설명은 [Oracle Solaris 10 8/11 새로운 기능](#)을 참조하십시오.

SPARC: 코히어런트(Coherent) 콘솔

Solaris 10 8/07: 코히어런트 콘솔 부속 시스템 기능은 콘솔 출력을 간편하게 렌더링하기 위해 커널 콘솔 부속 시스템의 일부를 구현합니다. 코히어런트 콘솔은 Oracle Solaris 커널 방식을 사용하여 PROM(Programmable Read-Only Memory) 인터페이스가 아닌 콘솔 출력을 렌더링합니다. 따라서 콘솔 렌더링 시 OBP(OpenBoot PROM)에 대한 의존도가 줄어듭니다. 코히어런트 콘솔은 커널에 상주하는 프레임 버퍼 드라이버를 사용하여 콘솔 출력을 생성합니다. 이렇게 생성된 콘솔 출력은 OBP 렌더링을 사용하는 경우보다

더 효율적입니다. 코히어런트 콘솔은 또한 SPARC 콘솔 출력 중 CPU가 유힬 상태로 되는 것을 방지하며 사용자 환경을 향상시킵니다.

SPARC: 콘솔에 대한 \$TERM 값 설정 방법 변경

Solaris 10 8/07: \$TERM 값이 이제 동적으로 파생되며 콘솔에서 사용하는 터미널 에뮬레이터에 따라 달라집니다. x86 기반 시스템의 경우에는 항상 커널 터미널 에뮬레이터가 사용되므로 \$TERM 값이 sun-color입니다.

SPARC 기반 시스템에서 \$TERM 값은 다음과 같습니다.

sun-color 시스템에서 커널의 터미널 에뮬레이터를 사용하는 경우 이 값이 \$TERM에 사용됩니다.

sun 시스템에서 PROM의 터미널 에뮬레이터를 사용하는 경우 이 값이 \$TERM에 사용됩니다.

이 변경으로 인해 직렬 포트에 대해 터미널 유형이 설정되는 방법이 달라지지는 않습니다. 다음 예에 표시된 것과 같이 svccfg 명령을 사용해서도 \$TERM 값을 수정할 수 있습니다.

```
# svccfg
svc:> select system/console-login
svc:/system/console-login> setprop ttymon/terminal_type = "xterm"
svc:/system/console-login> exit
```

SMF로 관리되는 시스템 콘솔에 대한 ttymon 호출

Oracle Solaris 10: 시스템 콘솔에 대한 ttymon 호출은 SMF에 의해 관리됩니다. svc:/system/console-login:default 서비스에 속성을 추가하면 ttymon 명령 인수를 svccfg 명령과 함께 지정할 수 있습니다. 이러한 속성은 일반 SMF 속성이 아닌 ttymon에만 적용됩니다.

주 - ttymon 호출은 /etc/inittab 파일에서 더 이상 사용자 정의할 수 없습니다.

SMF로 ttymon 명령 인수를 지정하는 방법에 대한 단계별 지침은 40 페이지 “ttymon 콘솔 터미널 유형을 설정하는 방법”을 참조하십시오.

SMF에 대한 전체 개요는 **시스템 관리 설명서: 기본 관리의 18 장, “서비스 관리(개요)”**를 참조하십시오. SMF와 연관된 단계별 절차에 대한 자세한 내용은 **시스템 관리 설명서: 기본 관리의 19 장, “서비스 관리(작업)”**를 참조하십시오.

터미널, 모뎀, 포트 및 서비스

터미널 및 모뎀은 시스템 및 네트워크 리소스에 대한 로컬 및 원격 액세스를 제공합니다. 터미널 및 모뎀 액세스 권한을 설정하는 것은 시스템 관리자의 중요한 책임입니다. 이 단원에서는 Oracle Solaris 운영 체제에서 모뎀 및 터미널 관리에 대한 몇 가지 개념에 대해 설명합니다.

터미널 설명

사용자 시스템의 비트맵 그래픽 디스플레이는 영숫자 터미널과 같지 않습니다. 영숫자 터미널은 직렬 포트에만 연결하며 텍스트만 표시합니다. 그래픽 디스플레이를 관리하기 위해 특별한 단계를 수행할 필요는 없습니다.

모뎀 설명

모뎀은 다음과 같은 세 가지 기본 구성으로 설정할 수 있습니다.

- 다이얼 아웃
- 다이얼 인
- 양방향

가정용 컴퓨터에 연결된 모뎀은 **다이얼 아웃** 서비스를 제공하도록 설정되었을 수 있습니다. 다이얼 아웃 서비스를 사용할 경우, 사용자의 가정에서 다른 컴퓨터에 액세스할 수 있습니다. 그러나 외부에 있는 사람은 사용자의 시스템에 액세스할 수 없습니다.

다이얼 인 서비스는 그 반대입니다. 다이얼 인 서비스를 사용할 경우 원격 사이트에서 시스템에 액세스할 수 있습니다. 그러나 외부로 호출할 수는 없습니다.

양방향 액세스는 이름에서 알 수 있듯이 다이얼 인 기능과 다이얼 아웃 기능을 모두 제공합니다.

포트 설명

포트는 장치가 운영 체제와 통신하는 데 사용되는 채널입니다. 하드웨어 관점에서 포트는 터미널 또는 모뎀 케이블을 물리적으로 연결할 수 있는 "콘센트"입니다.

그러나 엄격하게 포트는 물리적인 콘센트가 아니라, 하드웨어(핀 및 커넥터) 및 소프트웨어(장치 드라이버) 구성 요소를 포함하는 엔티티입니다. 한 개의 물리적 콘센트는 대개 둘 이상의 장치에 연결할 수 있도록 여러 개의 포트를 제공합니다.

일반적인 포트 유형으로는 직렬, 병렬, SCSI(Small Computer Systems Interface) 및 이더넷이 있습니다.

직렬 포트는 표준 통신 프로토콜을 사용하며, 단일 회선을 통해 바이트 단위의 정보를 비트 단위로 전송합니다.

RS-232-C 또는 RS-423 표준에 따라 설계된 장치는 대부분 모뎀, 영숫자 터미널, 플로터 및 몇 개의 프린터를 포함합니다. 이러한 장치는 표준 케이블을 사용하여 유사하게 설계된 컴퓨터의 직렬 포트에 서로 연결할 수 있습니다.

한 대의 컴퓨터에 직렬 포트 장치를 여러 개 연결해야 하는 경우 시스템에 **어댑터 보드**를 추가해야 할 수 있습니다. 어댑터 보드는 자체 드라이버 소프트웨어를 포함하고 있으며, 수용 가능한 것보다 많은 장치를 연결할 수 있는 추가 직렬 포트를 제공합니다.

서비스 설명

모뎀 및 터미널은 직렬 포트 소프트웨어를 사용하여 컴퓨팅 리소스에 액세스할 수 있습니다. 직렬 포트 소프트웨어는 포트에 연결된 장치에 알맞은 특정 "서비스"를 제공하도록 설정되어야 합니다. 예를 들어 모뎀에 대해 양방향 서비스를 제공하도록 직렬 포트를 설정할 수 있습니다.

포트 모니터

서비스에 액세스하기 위한 기본 방식은 **포트 모니터**를 통해 액세스하는 것입니다. 포트 모니터는 프린터 또는 파일에 액세스 또는 로그인하기 위한 요청을 끊임없이 모니터링하는 프로그램입니다.

포트 모니터가 요청을 감지하면 운영 체제와 서비스를 요청하는 장치 간 통신을 설정하는 데 필요한 매개변수를 설정합니다. 그런 다음 필요한 서비스를 제공하는 다른 서비스에 제어 권한을 전송합니다.

다음 표는 Oracle Solaris 릴리스에 포함된 포트 모니터의 두 가지 유형을 보여줍니다.

표 1-1 포트 모니터 유형

매뉴얼 페이지	포트 모니터	설명
listen(1M)	listen	Solaris 2.6 이전 릴리스에서 원격 인쇄 요청 처리와 같은 네트워크 서비스에 대한 액세스를 제어합니다. 기본 Oracle Solaris OS에서는 더 이상 이 포트 모니터 유형이 사용되지 않습니다.

표 1-1 포트 모니터 유형 (계속)

매뉴얼 페이지	포트 모니터	설명
ttymon(1M)	<code>ttymon</code>	모뎀 및 영숫자 터미널에 필요한 로그인 서비스에 대한 액세스를 제공합니다. 직렬 포트 도구가 해당 장치의 로그인 요청을 처리하도록 자동으로 <code>ttymon</code> 포트 모니터를 설정합니다.

이전의 포트 모니터인 `getty`에 더 익숙할 수도 있습니다. 새 `ttymon` 포트 모니터는 더 강력합니다. 한 개의 `ttymon` 포트 모니터가 여러 개의 `getty`를 대체할 수 있습니다. 그렇지 않은 경우 이 두 프로그램이 같은 기능을 수행합니다. 자세한 내용은 [getty\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

터미널 및 모뎀 관리를 위한 도구

다음 표는 터미널 및 모뎀 관리를 위한 도구를 보여줍니다.

표 1-2 터미널 및 모뎀 관리를 위한 도구

터미널 및 모뎀 관리 방법	도구	자세한 정보
가장 포괄적	SAF(Service Access Facility) 명령	24 페이지 “Service Access Facility 개요”
가장 신속한 설정	Solaris Management Console의 직렬 포트 도구	2 장, “터미널 및 모뎀 설정(작업)” 및 Solaris Management Console 온라인 도움말

직렬 포트 도구

직렬 포트 도구는 `pmadm` 명령을 적절한 정보와 함께 호출하여 터미널 및 모뎀과 함께 작동할 직렬 포트 소프트웨어를 설정합니다.

다음과 같은 정보도 제공합니다.

- 일반 터미널 및 모뎀 구성에 대한 템플릿
- 다중 포트 설정, 수정 또는 삭제
- 각 포트에 대한 신속한 시각적 상태

Service Access Facility 개요

SAF는 터미널, 모뎀 및 기타 네트워크 장치를 관리하는 데 사용되는 도구입니다.

SAF를 사용하면 특히 다음을 설정할 수 있습니다.

- `ttymon` 및 `listen` 포트 모니터 - `sacadm` 명령 사용
- `ttymon` 포트 모니터 서비스 - `pmadm` 및 `ttyadm` 명령 사용
- `listen` 포트 모니터 서비스 - `pmadm` 및 `nlsadmin` 명령 사용
- `tty` 장치 문제 해결
- 인쇄 서비스에 대한 수신 네트워크 요청 문제 해결
- `sacadm` 명령을 사용하여 Service Access Controller 문제 해결

SAF는 `tty` 장치 및 근거리 통신망(LAN)을 통해 시스템 및 네트워크 리소스에 대한 액세스를 제어하는 개방형 시스템 솔루션입니다. SAF는 프로그램이 아니라 백그라운드 프로세스 및 관리 명령 계층입니다.

터미널 및 모뎀 설정(작업)

이 장은 Solaris Management Console의 직렬 포트 도구를 사용하여 터미널 및 모뎀을 설정하기 위한 단계별 지침을 제공합니다.

터미널 및 모뎀에 대한 개요 정보는 1 장, “터미널 및 모뎀 관리(개요)”를 참조하십시오. 시스템 리소스 관리에 대한 개요 정보는 4 장, “시스템 리소스 관리(개요)”를 참조하십시오.

Solaris Management Console의 직렬 포트 도구를 사용하여 터미널 및 모뎀을 설정하는 것과 관련된 절차에 대한 정보는 25 페이지 “터미널 및 모뎀 설정(작업 맵)”을 참조하십시오.

터미널 및 모뎀 설정(작업 맵)

작업	설명	수행 방법
터미널 설정	Solaris Management Console 직렬 포트 도구를 사용하여 터미널을 설정합니다. 터미널을 구성하려면 Action(작업) 메뉴에서 적절한 옵션을 선택합니다.	29 페이지 “터미널을 설정하는 방법”
모뎀 설정	Solaris Management Console 직렬 포트 도구를 사용하여 모뎀을 설정합니다. 모뎀을 구성하려면 Action(작업) 메뉴에서 적절한 옵션을 선택합니다.	29 페이지 “모뎀을 설정하는 방법”

작업	설명	수행 방법
포트 초기화	포트를 초기화하려면 Solaris Management Console 직렬 포트 도구를 사용합니다. Action(작업) 메뉴에서 적절한 옵션을 선택합니다.	30 페이지 “포트를 초기화하는 방법”

직렬 포트 도구를 사용하여 터미널 및 모뎀 설정(개요)

Solaris Management Console의 직렬 포트 도구를 사용하여 직렬 포트를 설정할 수 있습니다.

Serial Ports(직렬 포트) 창에서 직렬 포트를 선택한 다음 Action(작업) 메뉴에서 Configure(구성) 옵션을 선택하여 다음을 구성하십시오.

- 터미널
- 모뎀 - 다이얼 인
- 모뎀 - 다이얼 아웃
- 모뎀 - 다이얼 인/다이얼 아웃
- 초기화만 - 연결 안함

Configure(구성) 옵션을 통해 이러한 서비스를 구성할 수 있는 템플릿에 액세스할 수 있습니다. 직렬 포트 각각에 대해 기본 및 고급이라는 두 가지 레벨의 세부 정보를 확인할 수 있습니다. 직렬 포트를 선택한 다음 Action(작업) 메뉴에서 Properties(속성) 옵션을 선택하여 직렬 포트를 구성한 후에는 각 직렬 포트에 대한 고급 레벨의 세부 정보에 액세스할 수 있습니다. 직렬 포트가 구성되면 SAF 명령으로 포트를 사용/사용 안함으로 설정할 수 있습니다. SAF 명령 사용에 대한 자세한 내용은 3 장, “Service Access Facility를 사용하여 직렬 포트 관리(작업)”를 참조하십시오.

직렬 포트 명령줄 인터페이스 사용에 대한 자세한 내용은 `smserialport(1M)` 매뉴얼 페이지를 참조하십시오.

터미널 설정

다음 표는 직렬 포트 도구를 사용하여 터미널을 설정할 때 메뉴 항목 및 기본값을 보여줍니다.

표 2-1 터미널 기본값

세부 정보	항목	기본값
기본	포트	
	설명	터미널

표 2-1 터미널 기본값 (계속)

세부 정보	항목	기본값
고급	서비스 상태	사용
	변조 속도	9600
	터미널 유형	vi925
	로그인 프롬프트	ttyn login:
	캐리어 감지	소프트웨어
	옵션: 캐리어에 대한 연결	사용할 수 없음
	옵션: 양방향	사용 가능
	옵션: 초기화만	사용할 수 없음
	시간 초과(초)	안함
	포트 모니터	zsmon
	서비스 프로그램	/usr/bin/login

모뎀 설정

다음 표는 직렬 포트 도구를 사용하여 설정할 때 사용할 수 있는 세 개의 모뎀 템플리트를 보여줍니다.

표 2-2 모뎀 템플리트

모뎀 구성	설명
다이얼 인만	사용자가 모뎀으로 다이얼 인할 수는 있지만 다이얼 아웃할 수는 없습니다.
다이얼 아웃만	사용자가 모뎀에서 다이얼 아웃할 수는 있지만 다이얼 인할 수는 없습니다.
다이얼 인 및 다이얼 아웃(양방향)	사용자가 모뎀에서 다이얼 인 또는 다이얼 아웃할 수 있습니다.

다음 표는 각 템플리트의 기본값을 보여줍니다.

표 2-3 모뎀 템플리트 기본값

세부 정보	항목	모뎀 - 다이얼 인만	모뎀 - 다이얼 아웃만	모뎀 - 다이얼 인 및 다이얼 아웃
기본	포트 이름			
	설명	모뎀 - 다이얼 인만	모뎀 - 다이얼 아웃만	모뎀 - 다이얼 인 및 다이얼 아웃

표 2-3 모뎀 템플리트 기본값 (계속)

세부 정보	항목	모뎀-다이얼 인만	모뎀-다이얼 아웃만	모뎀-다이얼 인 및 다이얼 아웃
	서비스 상태	사용	사용	사용
	변조 속도	9600	9600	9600
	로그인 프롬프트	ttyn login:	ttyn login:	ttyn login:
고급	캐리어 감지	소프트웨어	소프트웨어	소프트웨어
	옵션: 캐리어에 대한 연결	사용할 수 없음	사용할 수 없음	사용할 수 없음
	옵션: 양방향	사용할 수 없음	사용할 수 없음	사용 가능
	옵션: 초기화만	사용할 수 없음	사용 가능	사용할 수 없음
	시간 초과(초)	안함	안함	안함
	포트 모니터	zsmon	zsmon	zsmon
	서비스 프로그램	/usr/bin/login	/usr/bin/login	/usr/bin/login

다음 표는 초기화만 템플리트의 기본값을 보여줍니다.

표 2-4 초기화만-연결 안함 기본값

세부 정보	항목	기본값
기본	포트 이름	-
	설명	초기화만-연결 안함
	서비스 상태	사용
	변조 속도	9600
	로그인 프롬프트	ttyn login:
고급	캐리어 감지	소프트웨어
	옵션: 캐리어에 대한 연결	사용할 수 없음
	옵션: 양방향	사용 가능
	옵션: 초기화만	사용 가능
	시간 초과(초)	안함
	포트 모니터	zsmon
	서비스 프로그램	/usr/bin/login

터미널 및 모뎀을 설정하고 포트를 초기화하는 방법(작업)

▼ 터미널을 설정하는 방법

- 1 아직 실행 중이지 않은 경우 Solaris Management Console을 시작합니다.

```
% /usr/sadm/bin/smc &
```

Solaris Management Console 시작에 대한 자세한 내용은 [시스템 관리 설명서: 기본 관리의 “Solaris Management Console 시작”](#)을 참조하십시오.

- 2 탐색 창에서 **This Computer**(이 컴퓨터) 아이콘을 누릅니다.
- 3 **Devices and Hardware**(장치 및 하드웨어) -> **Serial Ports**(직렬 포트)를 누릅니다.
Serial Ports(직렬 포트) 메뉴가 표시됩니다.
- 4 터미널에 사용할 포트를 선택합니다.
- 5 **Action**(작업) 메뉴에서 **Configure**(구성) —> **Terminal**(터미널)을 선택합니다.
Configure Serial Port(직렬 포트 구성) 창이 Basic Detail(기본 세부 정보) 모드로 표시됩니다.
터미널 메뉴 항목에 대한 설명은 [표 2-1](#)을 참조하십시오.
- 6 **OK**(확인)를 누릅니다.
- 7 고급 항목을 구성하려면 터미널로 구성된 포트를 선택합니다.
- 8 **Action**(작업) 메뉴에서 **Properties**(속성)을 선택합니다.
- 9 원하는 경우 템플릿 항목의 값을 변경합니다.
- 10 **OK**(확인)을 눌러 포트를 구성합니다.
- 11 터미널 서비스가 추가되었는지 확인합니다.

```
$ pmadm -l -s tty"
```

▼ 모뎀을 설정하는 방법

- 1 아직 실행 중이지 않은 경우 Solaris Management Console을 시작합니다.

```
% /usr/sadm/bin/smc &
```

Solaris Management Console 시작에 대한 자세한 내용은 [시스템 관리 설명서: 기본 관리의 “Solaris Management Console 시작”](#)을 참조하십시오.

- 2 탐색 창에서 **This Computer**(이 컴퓨터) 아이콘을 누릅니다.
 - 3 **Devices and Hardware**(장치 및 하드웨어) —> **Serial Ports**(직렬 포트)를 누릅니다.
Serial Ports(직렬 포트) 메뉴가 표시됩니다.
 - 4 모뎀에 사용할 포트를 선택합니다.
 - 5 **Action**(작업) 메뉴에서 다음 **Configure**(구성) 옵션 중 하나를 선택합니다.
 - a. **Configure**(구성) —> **Modem (Dial In)**(모뎀(다이얼 인))
 - b. **Configure**(구성) —> **Modem (Dial Out)**(모뎀(다이얼 아웃))
 - c. **Configure**(구성) —> **Modem (Dial In/Out)**(모뎀(다이얼 인/아웃))
- Configure Serial Port(직렬 포트 구성) 창이 Basic Detail(기본 세부 정보) 모드로 표시됩니다.
- 모뎀 메뉴 항목에 대한 설명은 [표 2-3](#)을 참조하십시오.
- 6 **OK**(확인)를 누릅니다.
 - 7 고급 항목을 구성하려면 모뎀으로 구성된 포트를 선택합니다.
 - 8 **Action**(작업) 메뉴에서 **Properties**(속성)을 선택합니다.
 - 9 원하는 경우 템플릿 항목의 값을 변경합니다.
 - 10 **OK**(확인)을 눌러 포트를 구성합니다.
 - 11 모뎀 서비스가 구성되었는지 확인합니다.

```
$ pmadm -l -s ttyn
```

▼ 포트를 초기화하는 방법

- 1 아직 실행 중이지 않은 경우 Solaris Management Console을 시작합니다.

```
% /usr/sadm/bin/smc &
```

Solaris Management Console 시작에 대한 자세한 내용은 [시스템 관리 설명서: 기본 관리의 “Solaris Management Console 시작”](#)을 참조하십시오.

- 2 탐색 창에서 **This Computer**(이 컴퓨터) 아이콘을 누릅니다.
- 3 **Devices and Hardware**(장치 및 하드웨어) —> **Serial Ports**(직렬 포트)를 누릅니다.
Serial Ports(직렬 포트) 메뉴가 표시됩니다.
- 4 초기화할 포트를 선택합니다.
- 5 **Configure**(구성) —> **Initialize Only**(초기화만) – **No Connection**(연결 안함)을 선택합니다.
Serial Port(직렬 포트) 창이 **Basic Detail**(기본 세부 정보) 모드로 표시됩니다.
초기화만 메뉴 항목에 대한 설명은 [표 2-4](#)를 참조하십시오.
- 6 **OK**(확인)를 누릅니다.
- 7 고급 항목을 구성하려면 초기화만으로 구성된 포트를 선택합니다 그런 다음 **Action**(작업) 메뉴에서 **Properties**(속성)을 선택합니다.
- 8 원하는 경우 템플릿 항목의 값을 변경합니다.
- 9 **OK**(확인)을 눌러 포트를 구성합니다.
- 10 모뎀 서비스가 초기화되었는지 확인합니다.

```
$ pmadm -l -s ttyn
```

터미널 및 모뎀 문제 해결

터미널 또는 모뎀을 추가하고 적절한 서비스를 설정한 후 직렬 포트 회선을 통해 로그인할 수 없는 경우, 다음과 같은 가능한 실패 원인을 고려하십시오.

- 사용자와 확인합니다.
터미널 및 모뎀 사용의 오작동은 보통 로그인 또는 다이얼 인에 실패한 사용자가 보고합니다. 따라서 문제 해결은 데스크탑에 문제가 있는지 확인하는 것에서부터 시작하십시오.

로그인 실패의 몇 가지 일반적인 원인은 다음과 같습니다.
 - 로그인 ID 또는 암호가 올바르지 않습니다.
 - 터미널이 X-ON 플로우 Ctrl 키(Ctrl-Q)를 기다리고 있습니다.
 - 직렬 케이블이 느슨하거나 플러그가 뽑혀 있습니다.
 - 터미널 구성이 올바르지 않습니다.
 - 터미널이 차단되었거나 전원이 공급되지 않았습니다.
- 터미널을 확인합니다.

이어서 터미널 또는 모뎀의 구성을 확인하여 문제를 해결합니다. 터미널 또는 모뎀과 통신하는 데 적절한 *ttylabel*을 확인합니다. 터미널 또는 모뎀 설정이 *ttylabel* 설정과 일치하는지 확인합니다.

- 터미널 서버를 확인합니다.

터미널을 확인한 경우 이어서 터미널 또는 모뎀 서버에서 문제의 원인을 검색합니다. `pmadm` 명령을 사용하여 포트 모니터가 터미널 또는 모뎀을 서비스하도록 구성되었는지와 올바른 *ttylabel*이 포트 모니터와 연관되었는지 확인합니다. 예를 들어, 다음과 같습니다.

```
$ pmadm -l -t ttymon
```

`/etc/ttydefs` 파일을 확인한 다음 터미널 구성과 비교하여 레이블 정의를 다시 확인합니다. `sacadm` 명령을 사용하여 포트 모니터의 상태를 확인합니다. `pmadm`을 사용하여 터미널이 사용하는 포트와 연관된 서비스를 확인합니다.

- 직렬 연결을 확인합니다.

Service Access Controller가 TTY 포트 모니터를 시작하는 중이면서 다음이 True인 경우

- `pmadm` 명령이 터미널 포트에 대한 서비스가 **사용**으로 설정된 것으로 보고합니다.
- 터미널 구성이 포트 모니터의 구성과 일치합니다.

이어서 직렬 연결을 확인하여 문제를 검색합니다. 직렬 연결은 직렬 포트, 케이블 및 터미널로 구성됩니다. 한 파트를 안전하다고 알려진 다른 두 파트와 함께 사용하여 이들 파트를 각각 테스트합니다.

다음은 모두 테스트합니다.

- 직렬 포트
- 모뎀
- 케이블
- 커넥터
- 직렬 포트를 콘솔로 사용하고 있는 경우 직렬 포트 도구를 사용하여 직렬 포트 설정을 수정하지 마십시오. Oracle Solaris 10부터 콘솔에 대한 `ttymon` 호출은 SMF에 의해 관리됩니다. 콘솔 터미널 유형을 변경하는 방법에 대한 단계별 지침은 [40 페이지 “ttymon 콘솔 터미널 유형을 설정하는 방법”](#)을 참조하십시오.

`ttymon` 및 SMF에 대한 자세한 내용은 [19 페이지 “터미널 및 모뎀 관리의 새로운 기능”](#)을 참조하십시오.

Service Access Facility를 사용하여 직렬 포트 관리(작업)

이 장은 SAF(Service Access Facility)를 사용하여 직렬 포트 서비스를 관리하는 방법에 대해 설명합니다.

또한 SMF(서비스 관리 기능)를 사용하여 콘솔 관리를 수행하는 방법도 설명합니다.

주 - SAF 및 SMF는 Oracle Solaris OS의 서로 다른 두 도구입니다. Oracle Solaris 10부터 시스템 콘솔에 대한 `ttymon` 호출은 SMF에 의해 관리됩니다. SAF 도구는 터미널, 모뎀 및 기타 네트워크 장치를 관리하는 데 계속 사용됩니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 34 페이지 “Service Access Facility 사용”
- 35 페이지 “전반적인 SAF 관리(`sacadm`)”
- 36 페이지 “포트 모니터 서비스 관리(`pmadm`)”
- 38 페이지 “TTY 모니터 및 네트워크 Listener 포트 모니터”

직렬 포트 관리와 연관된 단계별 절차에 대한 자세한 내용은 다음을 참조하십시오.

- 34 페이지 “직렬 포트 관리(작업 맵)”
- 45 페이지 “`ttymon` 서비스 관리(작업 맵)”

SAF에 대한 참조 정보는 49 페이지 “Service Access Facility 관리(참조)”를 참조하십시오.

직렬 포트 관리(작업 맵)

작업	설명	수행 방법
콘솔 관리 수행	다음과 같은 콘솔 관리 작업을 수행해야 합니다. <ul style="list-style-type: none"> ■ ttymon 콘솔 터미널 유형 설정 Oracle Solaris 10부터는 svccfg 명령을 사용하여 ttymon 콘솔 터미널 유형을 지정해야 합니다. ■ ttymon 콘솔 터미널 변조 속도 설정 	40 페이지 “ttymon 콘솔 터미널 유형을 설정하는 방법” 41 페이지 “ttymon 콘솔 터미널에 대한 변조 속도를 설정하는 방법”
ttymon 포트 모니터 추가	sacadm 명령을 사용하여 ttymon 포트 모니터를 추가합니다.	42 페이지 “ttymon 포트 모니터를 추가하는 방법”
ttymon 포트 모니터 상태 확인	sacadm 명령을 사용하여 ttymon 포트 모니터 상태를 확인합니다.	42 페이지 “ttymon 포트 모니터 상태를 확인하는 방법”
ttymon 포트 모니터 중지	sacadm 명령을 사용하여 ttymon 포트 모니터를 중지합니다.	43 페이지 “ttymon 포트 모니터를 중지하는 방법”
ttymon 포트 모니터 시작	sacadm 명령을 사용하여 ttymon 포트 모니터를 시작합니다.	43 페이지 “ttymon 포트 모니터를 시작하는 방법”
ttymon 포트 모니터를 사용 안함으로 설정	sacadm 명령을 사용하여 ttymon 포트 모니터를 사용 안함으로 설정합니다.	44 페이지 “ttymon 포트 모니터를 사용 안함으로 설정하는 방법”
ttymon 포트 모니터를 사용으로 설정	sacadm 명령을 사용하여 ttymon 포트 모니터를 사용으로 설정합니다.	44 페이지 “ttymon 포트 모니터를 사용으로 설정하는 방법”
ttymon 포트 모니터 제거	sacadm 명령을 사용하여 ttymon 포트 모니터를 제거합니다.	44 페이지 “ttymon 포트 모니터를 제거하는 방법”

Service Access Facility 사용

Solaris Management Console의 직렬 포트 도구 또는 SAF 명령을 사용하여 터미널 및 모뎀을 설정할 수 있습니다.

SAF는 터미널, 모뎀 및 기타 네트워크 장치를 관리하는 데 사용되는 도구입니다. 최상위 SAF 프로그램은 SAC(Service Access Controller)입니다. SAC는 sacadm 명령을 통해 관리되는 포트 모니터를 제어합니다. 각 포트 모니터는 하나 이상의 포트를 관리할 수 있습니다.

포트와 연관된 서비스는 `pmadm` 명령을 통해 관리됩니다. SAC를 통해 제공되는 서비스는 네트워크마다 다를 수 있지만 SAC 및 해당 관리 명령 `sacadm`과 `pmadm`은 네트워크 독립적입니다.

다음 표는 SAF 제어 계층을 보여줍니다. `sacadm` 명령은 `ttymon` 및 `listen` 포트 모니터를 제어하는 SAC를 관리하는 데 사용됩니다.

`ttymon` 및 `listen`의 서비스는 `pmadm` 명령으로 제어됩니다. `ttymon`의 한 인스턴스는 여러 개의 포트를 서비스할 수 있습니다. `listen`의 한 인스턴스는 네트워크 인터페이스에서 여러 개의 서비스를 제공할 수 있습니다.

표 3-1 SAF 제어 계층

기능	프로그램	설명
전반적인 관리	<code>sacadm</code>	포트 모니터를 추가 및 제거하는 명령입니다.
Service Access Controller	<code>sac</code>	SAF의 마스터 프로그램입니다.
포트 모니터	<code>ttymon</code>	직렬 포트 로그인 요청을 모니터링합니다.
	<code>listen</code>	네트워크 서비스에 대한 요청을 모니터링합니다.
포트 모니터 서비스 관리자	<code>pmadm</code>	포트 모니터 서비스를 제어하는 명령입니다.
서비스	로그인, 원격 프로시저 호출	SAF가 액세스를 제공하는 서비스입니다.
콘솔 관리	콘솔 로그인	콘솔 서비스는 SMF 서비스 <code>svc:/system/console-login:default</code> 에 의해 관리됩니다. 이 서비스는 <code>ttymon</code> 포트 모니터를 호출합니다. 콘솔을 관리하는 데 <code>pmadm</code> 또는 <code>sacadm</code> 명령을 사용하지 마십시오. 자세한 내용은 38 페이지 “ <code>ttymon</code> 및 콘솔 포트”, 40 페이지 “ <code>ttymon</code> 콘솔 터미널 유형을 설정하는 방법” 및 41 페이지 “ <code>ttymon</code> 콘솔 터미널에 대한 변조 속도를 설정하는 방법”을 참조하십시오.

전반적인 SAF 관리(sacadm)

`sacadm` 명령은 최상위 SAF입니다. `sacadm` 명령은 주로 `ttymon` 및 `listen`과 같은 포트 모니터를 추가 및 제거하는 데 사용됩니다. 기타 `sacadm` 기능에는 포트 모니터의 현재 상태 나열 및 포트 모니터 구성 스크립트 관리 기능이 포함됩니다.

Service Access Controller(SAC 프로그램)

SAC(Service Access Controller) 프로그램은 모든 포트 모니터를 검사합니다. 다중 사용자 모드가 시작되면 시스템에서 자동으로 SAC가 시작됩니다.

SAC 프로그램은 호출되면 먼저 각 시스템의 구성 스크립트를 찾아 해석합니다. 사용자는 구성 스크립트를 사용하여 SAC 프로그램 환경을 사용자 정의할 수 있습니다. 이 스크립트는 기본적으로 비어 있습니다. SAC 환경에 대해 수정한 내용은 SAC의 모든 "자식"이 상속합니다. 이 상속된 환경은 자식이 수정할 수 있습니다.

SAC 프로그램은 시스템별 구성 스크립트를 해석한 후 해당 관리 파일을 읽고 지정된 포트 모니터를 시작합니다. SAC 프로그램은 포트 모니터별로 복사본을 실행하여 자식 프로세스를 포크합니다. 그러면 포트별 모니터 구성 스크립트가 있는 경우 각 자식 프로세스에서 이 스크립트를 해석합니다.

포트별 모니터 구성 스크립트에 지정된 환경에 대해 수정된 내용은 포트 모니터에 영향을 미치며 모든 자식이 상속하게 됩니다. 끝으로, 자식 프로세스는 SAC 프로그램 관리 파일에서 발견된 명령을 사용하여 포트 모니터 프로그램을 실행합니다.

SAC 초기화 프로세스

다음은 SAC를 처음 시작할 때 수행되는 프로세스를 요약한 단계입니다.

1. SAC 프로그램이 SMF 서비스 svc:/system/sac:default에 의해 시작됩니다.
2. SAC 프로그램이 시스템별 구성 스크립트 /etc/saf/_sysconfig를 읽습니다.
3. SAC 프로그램이 SAC 관리 파일 /etc/saf/_sactab을 읽습니다.
4. SAC 프로그램이 시작되는 포트 모니터 각각에 대해 자식 프로세스를 포크합니다.
5. 각 포트 모니터가 포트별 모니터 구성 스크립트 /etc/saf/pmtag/_config를 읽습니다.

포트 모니터 서비스 관리(pmadm)

pmadm 명령을 사용하여 포트 모니터 서비스를 관리할 수 있습니다. 특히 pmadm 명령은 서비스를 추가 또는 제거하고 서비스를 사용/사용 안함으로 설정하는 데 사용됩니다. 또한 서비스별 구성 스크립트를 설치 또는 대체하거나 서비스에 대한 정보를 인쇄할 수도 있습니다.

서비스의 각 인스턴스는 포트 모니터 및 포트에 고유하게 식별되어야 합니다. pmadm 명령을 사용하여 서비스를 관리하는 경우 특정 포트 모니터는 pmtag 인수와 함께 지정하고, 특정 포트는 svctag 인수와 함께 지정하십시오.

각 포트 모니터 유형에 대해 SAF에는 포트 모니터별 구성 데이터의 형식을 지정하는 특수화된 명령이 필요합니다. 이 데이터는 pmadm 명령에 사용됩니다. ttymon 및 listen 유형의 포트 모니터의 경우 이러한 특수화된 명령은 각각 ttyadm 및 nlsadmin입니다.

ttymon 포트 모니터

직접 연결된 모뎀 또는 영숫자 터미널을 사용하여 로그인하려고 시도하려 할 때마다 ttymon이 작동합니다. 먼저 SAC 프로세스가 SMF에 의해 시작됩니다. 그러면 SAC가 해당

관리 파일/etc/saf/_sactab에 지정된 포트 모니터를 자동으로 시작합니다. ttymon 포트 모니터가 시작되면 직렬 포트 회선에서 서비스 요청을 모니터링합니다.

어떤 사용자가 영숫자 터미널 또는 모뎀을 사용하여 로그인하려고 하면 직렬 포트 드라이버가 작업을 운영 체제에 전달합니다. ttymon 포트 모니터는 직렬 포트 작업을 메모하고 통신 링크를 설정하려고 시도합니다. ttymon 포트 모니터는 장치와 통신하는데 필요한 데이터 전송 속도, 회선 규칙 및 핸드셰이크 프로토콜을 결정합니다.

모뎀 또는 터미널과 통신하기 위한 적절한 매개변수가 설정되면 ttymon 포트 모니터가 이 매개변수를 로그인 프로그램에 전달하고 제어 권한을 이 프로그램에 전송합니다.

포트 초기화 프로세스

ttymon 포트 모니터의 인스턴스가 SAC에 의해 호출되면 ttymon이 포트 모니터를 시작합니다. 각 포트에 대해 ttymon 포트 모니터가 먼저 회선 규칙(지정된 경우)과 속도 및 터미널 설정을 초기화합니다. 초기화에 사용되는 값은 /etc/ttydefs 파일의 해당 항목에서 가져옵니다.

그런 다음 ttymon 포트 모니터는 프롬프트를 기록하고 사용자 입력을 기다립니다. 사용자가 Break 키를 눌러 속도가 부적절하다고 표시하면 ttymon 포트 모니터가 다음 속도를 시도하고 프롬프트를 다시 기록합니다.

포트에 대해 autobaud를 사용할 수 있는 경우 ttymon 포트 모니터가 자동으로 포트에 대한 변조 속도를 확인하려고 시도합니다. 사용자가 먼저 Return 키를 눌러야 ttymon 포트 모니터가 변조 속도를 인식하고 프롬프트를 인쇄할 수 있습니다.

유효한 입력을 수신한 경우 ttymon 포트 모니터는 다음과 같은 작업을 수행합니다.

- 포트에 대한 서비스별 구성 파일 해석
- /etc/utmpx 항목 만들기(필요한 경우)
- 서비스 환경 설정
- 포트와 연관된 서비스 호출

서비스가 종료되면 ttymon 포트 모니터는 /etc/utmpx 항목이 있을 경우 이 항목을 정리하고 포트를 초기 상태로 되돌립니다.

양방향 서비스

양방향 서비스용 포트가 구성된 경우 ttymon 포트 모니터는 다음을 수행합니다.

- 사용자가 서비스에 연결할 수 있도록 해줍니다.
- uucico, cu 또는 ct 명령에 다이얼 아웃용 포트를 사용할 수 있도록 해줍니다(포트가 사용 가능한 경우).
- 프롬프트를 인쇄하기 전에 문자를 읽을 때까지 기다립니다.

- 연결이 요청되면 프롬프트 메시지를 보내지 않고 포트의 연관된 서비스를 호출합니다(connect-on-carrier 플래그가 설정된 경우).

TTY 모니터 및 네트워크 Listener 포트 모니터

SAF는 향후 또는 타사 포트 모니터를 관리하기 위한 일반적인 방법을 제공하지만 Oracle Solaris 릴리스에서는 `ttymon` 및 `listen`이라는 두 개의 포트 모니터만 구현되었습니다.

TTY 포트 모니터(ttymon)

`ttymon` 포트 모니터는 STREAMS를 기반으로 하며 다음을 수행합니다.

- 포트 모니터
- 터미널 모드, 변조 속도 및 회선 규칙 설정
- 로그인 프로세스 호출

`ttymon` 포트 모니터는 이전 버전의 SunOS 4.1 소프트웨어에서 `getty` 포트 모니터가 수행한 것과 동일한 서비스를 사용자에게 제공합니다.

`ttymon` 포트 모니터는 SAC 프로그램 하에서 실행되며 `sacadm` 명령을 사용하여 구성됩니다. `ttymon`의 각 인스턴스는 여러 개의 포트를 모니터할 수 있습니다. 이러한 포트는 포트 모니터의 관리 파일에 지정되어 있습니다. 관리 파일은 `pmadm` 및 `ttyadm` 명령을 사용하여 구성됩니다.

ttymon 및 콘솔 포트

콘솔 서비스는 SAC(Service Access Controller)와 명시적 `ttymon` 관리 파일에 의해 관리되지 않습니다. `ttymon` 호출은 SMF에 의해 관리됩니다. 따라서 `/etc/inittab` 파일에 항목을 추가하는 방식으로는 더 이상 `ttymon`을 호출할 수 없습니다. 유형이 `application`이고 이름이 `ttymon`인 등록 정보 그룹이 SMF 서비스 `svc:/system/console-login:default`에 추가되었습니다. 이 등록 정보 그룹 내의 등록 정보는 메소드 스크립트 `/lib/svc/method/console-login`에서 사용됩니다. 이 스크립트에서는 등록 정보 값이 `ttymon` 호출에 대한 인수로 사용됩니다. 보통 값이 비어 있거나 어떠한 속성에 대해서도 정의되지 않은 경우 `ttymon`에 값이 사용되지 않습니다. 그러나 `ttymon` 장치 값이 비어 있거나 설정되지 않은 경우 `/dev/console`이 기본값으로 사용되어 `ttymon`이 실행되도록 합니다.

SMF 서비스 `svc:/system/console-login:default`에서 사용할 수 있는 속성은 다음과 같습니다.

<code>ttymon/nohangup</code>	<code>nohangup</code> 등록 정보를 지정합니다. <code>true</code> 로 설정된 경우, 기본값 또는 지정된 속도를 설정하기 전에 회선 속도를 0으로 설정하여 회선을 강제로 행업하지 않도록 합니다.
------------------------------	--

<code>ttymon/prompt</code>	콘솔 포트에 대한 프롬프트 문자열을 지정합니다.
<code>ttymon/terminal_type</code>	콘솔에 대한 기본 터미널 유형을 지정합니다.
<code>ttymon/device</code>	콘솔 장치를 지정합니다.
<code>ttymon/label</code>	<code>/etc/ttydefs</code> 라인에 TTY 레이블을 지정합니다.

ttymon 관련 관리 명령(ttyadm)

`ttymon` 관리 파일은 `sacadm` 및 `pmadm` 명령뿐 아니라 `ttyadm` 명령으로도 업데이트됩니다. `ttyadm` 명령은 `ttymon` 관련 정보의 형식을 지정한 다음 이 정보를 표준 출력에 기록하므로, 형식이 지정된 `ttymon` 관련 데이터를 `sacadm` 및 `pmadm` 명령에 표시할 수 있습니다.

따라서 `ttyadm` 명령은 `ttymon`을 직접 관리하지 않습니다. `ttyadm` 명령은 일반 관리 명령인 `sacadm` 및 `pmadm`을 보완합니다. 자세한 내용은 `ttyadm(1M)` 매뉴얼 페이지를 참조하십시오.

네트워크 Listener 서비스(listen)

`listen` 포트 모니터는 SAC하에서 실행되며 다음을 수행합니다.

- 네트워크에서 서비스 요청 모니터
- 요청 도착 시 수락
- 서비스 요청에 대한 응답으로 서버 호출

`listen` 포트 모니터는 `sacadm` 명령을 사용하여 구성됩니다. `listen`의 각 인스턴스는 여러 개의 서비스를 제공할 수 있습니다. 이러한 서비스는 포트 모니터의 관리 파일에 지정되어 있습니다. 이 관리 파일은 `pmadm` 및 `nlsadmin` 명령을 사용하여 구성됩니다.

네트워크 Listener 프로세스는 TLI(Transport Layer Interface)를 준수하는 연결 지향 전송 공급자와 함께 사용할 수 있습니다. Oracle Solaris OS에서 `listen` 포트 모니터는 `inetd` 서비스에서 제공하지 않는 추가 네트워크 서비스를 제공할 수 있습니다.

특수 listen 관련 관리 명령(nlsadmin)

`listen` 포트 모니터 관리 파일은 `sacadm` 및 `pmadm` 명령뿐 아니라 `nlsadmin` 명령으로도 업데이트됩니다. `nlsadmin` 명령은 `listen` 관련 정보의 형식을 지정한 다음 이 정보를 표준 출력에 기록하므로, 형식이 지정된 `listen` 관련 데이터를 `sacadm` 및 `pmadm` 명령에 표시할 수 있습니다.

따라서 `nlsadmin` 명령은 `listen`을 직접 관리하지 않습니다. 이 명령은 일반 관리 명령인 `sacadm` 및 `pmadm`을 보완합니다.

개별적으로 구성되는 각 네트워크에는 연관된 네트워크 `listener` 프로세스의 인스턴스가 하나 이상 있을 수 있습니다. `nlsadmin` 명령은 `listen` 포트 모니터의 작동 상태를 제어합니다.

`nlsadmin` 명령은 지정된 네트워크에 대해 `listen` 포트 모니터를 설정하고, 해당 포트 모니터의 특정 속성을 구성하며, 모니터를 시작 및 강제 종료합니다. `nlsadmin` 명령은 시스템에 대한 `listen` 포트 모니터를 보고할 수도 있습니다.

자세한 내용은 `nlsadmin(1M)` 매뉴얼 페이지를 참조하십시오.

ttymon 포트 모니터 관리

ttymon에 대한 콘솔 관리는 이제 SMF에 의해 관리됩니다. `svccfg` 명령을 사용하여 `ttymon` 시스템 콘솔 속성을 설정하십시오. 계속 SAF 명령 `sacadm`을 사용하여 `ttymon` 포트 모니터를 추가, 나열, 제거, 강제 종료, 시작, 사용/사용 안함으로 설정 및 제거할 수 있습니다.

▼ ttymon 콘솔 터미널 유형을 설정하는 방법

이 절차는 `svccfg` 명령을 사용하여 콘솔 터미널 유형을 변경하는 방법을 보여줍니다.

- 1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 `svccfg` 명령을 실행하여 변경할 서비스 인스턴스의 등록 정보를 설정합니다.

```
# svccfg -s console-login setprop ttymon/terminal_type = "xterm"
```

여기서 `xterm`은 사용하려는 터미널 유형의 예입니다.
- 3 (선택 사항) 서비스 인스턴스를 다시 시작합니다.

```
# svcadm restart svc:/system/console-login:default
```



주의 - 서비스 인스턴스를 즉시 다시 시작하도록 선택할 경우 콘솔에서 로그아웃됩니다. 서비스 인스턴스를 즉시 다시 시작하지 않을 경우 다음에 콘솔에 로그인할 때 등록 정보 변경 사항이 적용됩니다.

▼ ttymon 콘솔 터미널에 대한 변조 속도를 설정하는 방법

이 절차는 ttymon 콘솔 터미널에 대한 변조 속도를 설정하는 방법을 보여줍니다. x86 기반 시스템에서 콘솔 속도에 대한 지원은 플랫폼에 따라 다릅니다.

SPARC 기반 시스템에 대해 지원되는 콘솔 속도는 다음과 같습니다.

- 9600 bps
- 19200 bps
- 38400 bps

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 eeprom 명령을 통해 사용 중인 시스템 유형에 적합한 변조 속도를 설정합니다.

```
# eeprom ttya-mode=baud-rate,8,n,1,-
```

예를 들어 x86 기반 시스템의 콘솔에 대한 변조 속도를 38400으로 변경하려면 다음과 같이 입력합니다.

```
# eeprom ttya-mode=38400,8,n,1,-
```

3 /etc/ttydefs 파일에서 콘솔 라인을 다음과 같이 변경합니다.

```
console baud-rate hupcl opost onlcr:baud-rate::console
```

4 사용 중인 시스템 유형에 대해 다음과 같이 추가로 변경합니다.

이러한 변경 사항은 플랫폼에 따라 다릅니다.

- **SPARC 기반 시스템:** /kernel/drv/options.conf 파일에서 변조 속도를 변경합니다. 변조 속도를 9600으로 변경하려면 다음 명령을 사용합니다.

```
# 9600 :bd:
ttymodes="2502:1805:bd:8a3b:3:1c:7f:15:4:0:0:0:11:13:1a:19:12:f:17:16";
```

변조 속도를 19200으로 변경하려면 다음 명령을 사용합니다.

```
# 19200 :be:
ttymodes="2502:1805:be:8a3b:3:1c:7f:15:4:0:0:0:11:13:1a:19:12:f:17:16";
```

변조 속도를 38400으로 변경하려면 다음 명령을 사용합니다.

```
# 38400 :bf:
ttymodes="2502:1805:bf:8a3b:3:1c:7f:15:4:0:0:0:11:13:1a:19:12:f:17:16";
```

- **x86 기반 시스템:** BIOS 직렬 리디렉션이 사용 가능한 경우 콘솔 속도를 변경합니다. 콘솔 속도를 변경하는 데 사용하는 방법은 플랫폼에 따라 다릅니다.

▼ ttymon 포트 모니터를 추가하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 ttymon 포트 모니터를 추가합니다.

```
# sacadm -a -p mbmon -t ttymon -c /usr/lib/saf/ttymon -v 'ttyadm
-V' -y "TTY Ports a & b"
```

- a 추가 포트 모니터 옵션을 지정합니다.
- p `pmtagmbmon`을 포트 모니터 태그로 지정합니다.
- t 포트 모니터 유형을 `ttymon`으로 지정합니다.
- c 포트 모니터를 시작하는 데 사용된 명령 문자열을 정의합니다.
- v 포트 모니터의 버전 번호를 지정합니다.
- y 포트 모니터의 해당 인스턴스에 대한 설명을 정의합니다.

▼ ttymon 포트 모니터 상태를 확인하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 ttymon 포트 모니터의 상태를 확인합니다.

```
# sacadm -l -p mbmon
```

- l 나열 포트 모니터 상태 플래그를 지정합니다.
- p `pmtagmbmon`을 포트 모니터 태그로 지정합니다.

예 3-1 ttymon 포트 모니터 상태 확인

다음 예는 포트 모니터 `mbmon`을 확인하는 방법을 보여줍니다.

```
# sacadm -l -p mbmon
PMTAG PMTYPE FLGS RCNT STATUS COMMAND
mbmon ttymon - 0 STARTING /usr/lib/saf/ttymon #TTY Ports a & b
```

PMTAG 포트 모니터 이름 mbmon을 식별합니다.

PMTYPE 포트 모니터 유형 ttymon을 식별합니다.

FLGS 다음 플래그가 설정되어 있는지 여부를 나타냅니다.

d 새 포트 모니터를 사용으로 설정하지 않습니다.

x 새 포트 모니터를 시작하지 않습니다.

대시(-) 플래그가 설정되어 있지 않습니다.

RCNT 리턴 카운트 값을 나타냅니다. 리턴 카운트 0은 포트 모니터가 실패할 경우 포트 모니터를 다시 시작하지 않음을 나타냅니다.

STATUS 포트 모니터의 현재 상태를 나타냅니다.

COMMAND 포트 모니터를 시작하는 데 사용된 명령을 식별합니다.

#TTY Ports a & b 포트 모니터를 설명하는 데 사용된 설명을 식별합니다.

▼ ttymon 포트 모니터를 중지하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 ttymon 포트 모니터를 중지합니다.

```
# sacadm -k -p mbmon
-k 강제 종료 포트 모니터 상태 플래그를 지정합니다.
-p pmtag mbmon을 포트 모니터 태그로 지정합니다.
```

▼ ttymon 포트 모니터를 시작하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 강제 종료된 ttymon 포트 모니터를 시작합니다.

```
# sacadm -s -p mbmon
-s   시작 포트 모니터 상태 플래그를 지정합니다.
-p   pmtagmbmon을 포트 모니터 태그로 지정합니다.
```

▼ ttymon 포트 모니터를 사용 안함으로 설정하는 방법

포트 모니터를 사용 안함으로 설정하면 기존 서비스에 영향을 주지 않으면서 새 서비스가 시작되지 않도록 합니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 ttymon 포트 모니터를 사용 안함으로 설정

```
# sacadm -d -p mbmon
-d   사용 안함 포트 모니터 상태 플래그를 지정합니다.
-p   pmtagmbmon을 포트 모니터 태그로 지정합니다.
```

▼ ttymon 포트 모니터를 사용으로 설정하는 방법

ttymon 포트 모니터를 사용으로 설정하면 새 요청을 서비스할 수 있습니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 ttymon 포트 모니터를 사용으로 설정합니다.

```
# sacadm -e -p mbmon
-e   사용 포트 모니터 상태 플래그를 지정합니다.
-p   pmtagmbmon을 포트 모니터 태그로 지정합니다.
```

▼ ttymon 포트 모니터를 제거하는 방법

포트 모니터를 제거하면 연관된 구성 파일이 모두 삭제됩니다.

주 - 포트 모니터 구성 파일은 `sacadm` 명령을 사용하여 업데이트하거나 변경할 수 없습니다. 포트 모니터를 다시 구성하려면 포트 모니터를 제거한 다음 새로 추가하십시오.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 ttymon 포트 모니터 제거

```
# sacadm -r -p mbmon
```

-r 제거 포트 모니터 상태 플래그를 지정합니다.

-p `pmtag mbmon`을 포트 모니터 태그로 지정합니다.

ttymon 서비스 관리(작업 맵)

작업	설명	수행 방법
ttymon 서비스 추가	<code>pmadm</code> 명령을 사용하여 서비스를 추가합니다.	46 페이지 “서비스를 추가하는 방법”
TTY 포트 서비스의 상태 확인	<code>pmadm</code> 명령을 사용하여 TTY 포트의 상태를 확인합니다.	46 페이지 “TTY 포트 서비스의 상태를 확인하는 방법”
포트 모니터 서비스를 사용으로 설정	<code>pmadm</code> 명령을 <code>-e</code> 옵션과 함께 사용하여 포트 모니터를 사용으로 설정합니다.	49 페이지 “포트 모니터 서비스를 사용으로 설정하는 방법”
포트 모니터 서비스를 사용 안함으로 설정	<code>pmadm</code> 명령을 <code>-d</code> 옵션과 함께 사용하여 포트 모니터를 사용 안함으로 설정합니다.	49 페이지 “포트 모니터 서비스를 사용 안함으로 설정하는 방법”

ttymon 서비스 관리

`pmadm` 명령을 사용하여 서비스를 추가하고, 포트 모니터와 연관된 하나 이상의 포트 서비스를 나열하며, 서비스를 사용/사용 안함으로 설정할 수 있습니다.

▼ 서비스를 추가하는 방법

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 mbmon 포트 모니터에 표준 터미널 서비스를 추가합니다.

```
# pmadm -a -p mbmon -s a -i root -v 'ttyadm -V' -m "'ttyadm -i 'Terminal
  disabled' -l conTTY -m ldterm,ttcompat -S y -d /dev/term/a
-s /usr/bin/login'"
```

주 _ 이 예에서 입력 내용은 다음 라인으로 자동 줄 바꿈되어 있습니다. Return 키 또는 줄 바꿈을 사용하지 마십시오.

-a 추가 포트 모니터 상태 플래그를 지정합니다.

-p *pmtag* mbmon을 포트 모니터 태그로 지정합니다.

-s *svctag* a를 포트 모니터 서비스 태그로 지정합니다.

-i 서비스가 실행될 때 *svctag*에 지정할 ID를 지정합니다.

-v 포트 모니터의 버전 번호를 지정합니다.

-m *ttyadm*에 의해 형식이 지정된 *ttymon* 관련 구성 데이터를 지정합니다.

선행 *pmadm* 명령에는 내장 *ttyadm* 명령이 포함되어 있습니다. 이 내장 명령의 옵션은 다음과 같습니다.

-b 양방향 포트 플래그를 지정합니다.

-i 비활성(사용 안함) 응답 메시지를 지정합니다.

-l 사용할 */etc/ttydefs* 파일에 TTY 레이블을 지정합니다.

-m 이 서비스를 호출하기 전에 푸시할 STREAMS 모듈을 지정합니다.

-d TTY 포트에 사용할 장치에 대한 전체 경로 이름을 지정합니다.

-s 연결 요청을 수신할 때 호출할 서비스의 전체 경로 이름을 지정합니다. 인수가 필요한 경우 명령과 해당 인수를 따옴표(")로 묶으십시오.

▼ TTY 포트 서비스의 상태를 확인하는 방법

이 절차에 표시된 *pmadm* 명령을 사용하여 포트 모니터와 연관된 모든 포트 또는 TTY 포트의 상태를 나열할 수 있습니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 **System Administration Guide: Security Services**의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 포트 모니터의 한 서비스를 나열합니다.

```
# pmadm -l -p mbmon -s a
```

-l 시스템에 대한 서비스 정보를 나열합니다.

-p *pmtag*mbmon을 포트 모니터 태그로 지정합니다.

-s *svctag*a를 포트 모니터 서비스 태그로 지정합니다.

예 3-2 TTY 포트 모니터 서비스의 상태 확인

다음 예는 포트 모니터의 모든 서비스를 나열합니다.

```
# pmadm -l -p mbmon
PMTAG PMTYPE SVCTAG FLGS ID <PMSPECIFIC>
mbmon ttymon a - root /dev/term/a - /usr/bin/login - contty
ldterm,ttcompat login: Terminal disabled tvi925 y #
```

PMTAG pmadm -p 명령을 사용하여 설정된 포트 모니터 이름 mbmon을 식별합니다.

PMTYPE 포트 모니터 유형 ttymon을 식별합니다.

SVCTAG pmadm -s 명령을 사용하여 설정된 서비스 태그 값을 나타냅니다.

FLAGS 다음 플래그가 pmadm -f 명령을 사용하여 설정되었는지 여부를 식별합니다.

- x - 서비스를 사용으로 설정하지 않습니다.
- u - 서비스에 대해 utmpx 항목을 만듭니다.
- 대시(-) - 플래그가 설정되어 있지 않습니다.

ID 서비스가 시작될 때 서비스에 지정되는 ID를 나타냅니다. 이 값은 pmadm -i 명령을 사용하여 설정됩니다.

<PMSPECIFIC> 정보

/dev/term/a ttyadm -d 명령을 사용하여 설정된 TTY 포트 경로 이름을 나타냅니다.

- 다음 플래그가 ttyadm -c -b -h -I -r 명령을 사용하여 설정되었는지 여부를 나타냅니다.

- c - 포트의 캐리어 플래그에 대한 연결을 설정합니다.

	<ul style="list-style-type: none"> ■ b - 수신 및 송신 트래픽을 모두 허용하도록 포트를 양방향으로 설정합니다. ■ h - 수신 호출을 수신하는 즉시 자동 행업을 억제합니다. ■ I - 포트를 초기화합니다. ■ r - login: 메시지를 인쇄하기 전에 포트로부터 문자를 수신할 때까지 ttymon을 강제로 대기시킵니다. ■ 대시(-) - 플래그가 설정되어 있지 않습니다.
-	ttyadm -r count 옵션을 사용하여 설정된 값을 나타냅니다. 이 옵션은 ttymon이 포트로부터 데이터를 수신한 후 프롬프트를 표시할 때 결정됩니다. count가 0일 경우, ttymon은 문자를 수신할 때까지 기다립니다. count가 0보다 클 경우, ttymon은 count 새 라인을 수신할 때까지 기다립니다. 이 예에서는 값이 설정되어 있지 않습니다.
/usr/bin/login	연결을 수신할 때 호출할 서비스의 전체 경로 이름을 식별합니다. 이 값은 ttyadm -s 명령을 사용하여 설정됩니다.
-	ttyadm -t 명령의 시간 초과 값을 식별합니다. 이 옵션은 포트 열기를 성공할 경우 ttymon이 포트를 닫도록 지정합니다. timeout 초 내에는 입력 데이터가 수신되지 않습니다. 이 예에는 시간 초과 값이 없습니다.
contty	/etc/ttydefs 파일에서 TTY 레이블을 식별합니다. 이 값은 ttyadm -l 명령을 사용하여 설정됩니다.
ldterm, ttcompat	푸시할 STREAMS 모듈을 식별합니다. 이 모듈은 ttyadmin -m 명령을 사용하여 설정됩니다.
login: Terminal disabled	포트가 사용 안함으로 설정되었을 때 표시될 비활성 메시지를 식별합니다. 이 메시지는 ttyadm -i 명령을 사용하여 설정됩니다.
tvi925	ttyadm -T 명령을 사용하여 터미널 유형(설정된 경우)을 식별합니다. 이 예에서는 터미널 유형이 tvi925입니다.
y	ttyadm -s 명령을 사용하여 설정된 소프트웨어 캐리어 값을 식별합니다. n은 소프트웨어 캐리어를 해제합니다. y는 소프트웨어 캐리어를 설정합니다. 이 예에서는 소프트웨어 캐리어가 설정되어 있습니다.
#	pmadm -y 명령을 사용하여 지정된 주석을 식별합니다. 이 예에는 주석이 없습니다.

▼ 포트 모니터 서비스를 사용으로 설정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 사용 안함으로 설정된 포트 모니터 서비스를 사용으로 설정합니다.

```
# pmadm -e -p mbmon -s a
```

 - e 사용 플래그를 지정합니다.
 - p `pmtagmbmon`을 포트 모니터 태그로 지정합니다.
 - s `svctag a`를 포트 모니터 서비스 태그로 지정합니다.

▼ 포트 모니터 서비스를 사용 안함으로 설정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 포트 모니터 서비스를 사용 안함으로 설정합니다.

```
# pmadm -d -p mbmon -s a
```

 - d 사용 안함 플래그를 지정합니다.
 - p `pmtagmbmon`을 포트 모니터 태그로 지정합니다.
 - s `svctag a`를 포트 모니터 서비스 태그로 지정합니다.

Service Access Facility 관리(참조)

이 단원에는 Service Access Facility 관리를 위한 참조 정보가 포함되어 있습니다.

SAF와 연관된 파일

SAF는 `sacadm` 및 `pmadm` 명령을 사용하여 수정할 수 있는 구성 파일을 사용합니다. 구성 파일을 수동으로 편집할 필요는 없습니다.

파일 이름	설명
/etc/saf/_sysconfig	시스템별 구성 스크립트입니다.
/etc/saf/_sactab	SAC가 제어하는 포트 모니터에 대한 구성 데이터를 포함하는 SAC 관리 파일입니다.
/etc/saf/pmtag	포트 모니터 <i>pmtag</i> 의 홈 디렉토리입니다.
/etc/saf/pmtag/_config	포트 모니터 <i>pmtag</i> (있는 경우)에 대한 포트 모니터별 구성 스크립트입니다.
/etc/saf/pmtag/_pmtab	<i>pmtag</i> 가 제공하는 서비스에 대한 포트 모니터별 구성 데이터를 포함하는 포트 모니터 <i>pmtag</i> 의 관리 파일입니다.
/etc/saf/pmtag/svctag	<i>svctag</i> 서비스에 대한 서비스별 구성 스크립트입니다.
/var/saf/log	SAC 로그 파일입니다.
/var/saf/pmtag	<i>pmtag</i> 에 의해 생성된 파일(예: 로그 파일)의 디렉토리입니다.

/etc/saf/_sactab 파일

/etc/saf/_sactab 파일에 포함된 정보는 다음과 같습니다.

```
# VERSION=1
zsmon: ttymon::0:/usr/lib/saf/ttymon
#
# VERSION=1           Service Access Facility 버전 번호를 나타냅니다.
zsmon                포트 모니터의 이름입니다.
ttymon               포트 모니터의 유형입니다.
::                  다음 두 플래그가 설정되어 있는지 여부를 나타냅니다.
d   포트 모니터를 사용으로 설정하지 않습니다.
x   포트 모니터를 시작하지 않습니다. 이 예에서는 플래그가
    설정되어 있지 않습니다.
0   리턴 코드 값을 나타냅니다. 리턴 카운트 0은 포트 모니터가
    실패할 경우 포트 모니터를 다시 시작하지 않음을 나타냅니다.
/usr/lib/saf/ttymon  포트 모니터 경로 이름을 나타냅니다.
```

/etc/saf/pmtab/_pmtab 파일

/etc/saf/pmtab/_pmtab 파일(예: /etc/saf/zsmon/_pmtab)은 다음과 비슷합니다.

```
# VERSION=1
ttya:u:root:reserved:reserved:/dev/term/a:I:./usr/bin/login::9600:
ldterm,ttcompat:ttya login\: ::tvi925:y:#

# VERSION=1      Service Access Facility 버전 번호를 나타냅니다.

ttya             서비스 태그를 나타냅니다.

x,u             다음 플래그가 설정되어 있는지 여부를 식별합니다.
x               서비스를 사용으로 설정하지 않습니다.
u               서비스에 대해 utmpx 항목을 만듭니다.

root            서비스 태그에 지정되는 ID를 나타냅니다.

reserved        이 필드는 향후 사용을 위해 예약되었습니다.
reserved        이 필드는 향후 사용을 위해 예약되었습니다.
reserved        이 필드는 향후 사용을 위해 예약되었습니다.

/dev/term/a     TTY 포트 경로 이름을 나타냅니다.

/usr/bin/login  연결을 수신할 때 호출할 서비스의 전체 경로 이름을 식별합니다.

:c,b,h,I,r:     다음 플래그가 설정되어 있는지 여부를 나타냅니다.
c               포트의 캐리어 플래그에 대한 연결을 설정합니다.
b               수신 및 송신 트래픽을 모두 허용하도록 포트를 양방향으로
                설정합니다.
h               수신 호출을 수신하는 즉시 자동 행업을 억제합니다.
I               포트를 초기화합니다.
r               ttymon이 login: 메시지를 인쇄하기 전에 포트로부터 문자를
                수신할 때까지 ttymon을 강제로 대기시킵니다.

9600            /etc/ttydefs 파일에 정의된 TTY 레이블을 식별합니다.

ldterm,ttcompat 푸시할 STREAMS 모듈을 식별합니다.

ttya login\:    표시할 프롬프트를 식별합니다.

:y/n:          예 또는 아니오 응답을 나타냅니다.

message        모든 비활성(사용 안함) 응답 메시지를 식별합니다.

tvi925         터미널 유형을 식별합니다.
```

y 소프트웨어 캐리어가 설정되어 있는지 여부를 나타냅니다(y/ n).

서비스 상태

sacadm 명령은 서비스의 상태를 제어합니다. 다음 목록은 가능한 서비스 상태를 보여줍니다.

Enabled 기본 상태. 포트 모니터가 추가되면 서비스가 작동합니다.

Disabled 기본 상태. 포트 모니터가 제거되면 서비스가 중지됩니다.

특정 서비스의 상태를 확인하려면 다음을 사용하십시오.

```
# pmadm -l -p portmon-name -ssvctag
```

포트 모니터 상태

sacadm 명령은 ttymon 및 listen 포트 모니터의 상태를 제어합니다. 다음 표는 가능한 포트 모니터 상태를 보여줍니다.

상태	설명
Started	기본 상태 - 포트 모니터가 추가되면 자동으로 시작됩니다.
Enabled	기본 상태 - 포트 모니터가 추가되면 자동으로 서비스에 대한 요청을 수락할 준비가 됩니다.
Stopped	기본 상태 - 포트 모니터가 제거되면 자동으로 중지됩니다.
Disabled	기본 상태 - 포트 모니터가 제거되면 자동으로 기존 서비스를 계속 진행하되 새 서비스를 더 이상 추가하지는 않습니다.
Starting	중간 상태 - 포트 모니터가 시작되는 중입니다.
Stopping	중간 상태 - 포트 모니터가 수동으로 종료되었지만 종료 프로세스가 완료되지 않았습니다. 포트 모니터를 중지하는 중입니다.
Notrunning	비활성 상태 - 포트 모니터가 강제 종료되었습니다. 이전에 모니터링했던 모든 포트에 액세스할 수 없습니다. 외부 사용자는 포트가 disabled 또는 notrunning 상태인지 확인할 수 없습니다.
Failed	비활성 상태 - 포트 모니터를 시작할 수 없어 실행 중 상태로 남아 있습니다.

특정 포트 모니터의 상태를 확인하려면 다음 명령을 사용하십시오.

```
# sacadm -l -p portmon-name
```

포트 상태

포트를 제어하는 포트 모니터의 상태에 따라 포트를 사용/사용 안함으로 설정할 수 있습니다.

상태	설명
직렬(ttymon) 포트 상태	
Enabled	ttymon 포트 모니터가 프롬프트 메시지를 포트에 보내고 포트에 로그인 서비스를 제공합니다.
Disabled	ttymon이 강제 종료되었거나 사용 안함으로 설정된 경우 모든 포트의 기본 상태입니다. 이 상태를 지정한 경우 ttymon이 연결 요청을 수신하면 disabled 메시지를 보냅니다.

◆◆◆ 4 장

시스템 리소스 관리(개요)

이 장에서는 Oracle Solaris OS에서 사용할 수 있는 시스템 리소스 관리 기능을 간단히 설명하고 시스템 리소스 관리에 유용한 로드맵을 제공합니다.

이러한 기능을 사용하여 일반 시스템 정보를 표시하고, 디스크 공간을 모니터하고, 디스크 쿼터를 설정하고, 계산 프로그램을 사용할 수 있습니다. 또한 루틴 명령이 자동으로 실행되도록 cron 및 at 명령 일정을 잡을 수 있습니다.

이 단원에서는 시스템 리소스를 할당, 모니터, 제어하는 리소스 관리에 대한 정보는 다루지 않습니다.

리소스 관리 없이 시스템 리소스 관리와 연관된 절차에 대한 내용은 57 페이지 “시스템 리소스 관리(로드맵)”를 참조하십시오.

리소스 관리와 함께 시스템 리소스 관리에 대한 내용은 **시스템 관리 설명서: Oracle Solaris Containers-리소스 관리 및 Oracle Solaris 영역의 1 장, “Solaris 10 리소스 관리 소개”**.

시스템 리소스 관리의 새로운 기능

이 단원에서는 Oracle Solaris 릴리스에서 시스템 리소스를 관리하기 위해 새로 추가되거나 변경된 기능에 대해 설명합니다. 새로운 기능의 전체 목록 및 Oracle Solaris 릴리스에 대한 설명은 **Oracle Solaris 10 8/11 새로운 기능**을 참조하십시오.

제품 이름을 표시하는 새로운 prtconf 옵션

Solaris 10 1/06: 새로운 -b 옵션이 시스템의 제품 이름을 표시할 목적으로 prtconf 명령에 추가되었습니다. 이 옵션은 uname -i 명령과 비슷합니다. 하지만 prtconf -b 명령이 제품의 마케팅 이름을 결정하기 위해 특별히 설계되었습니다.

`prtconf` 명령에 `-b` 옵션을 사용하여 표시되는 펌웨어 장치 트리 루트 속성은 다음과 같습니다.

- name
- compatible
- banner-name
- model

사용 가능한 플랫폼 관련 출력을 추가로 표시하려면 `prtconf -vb` 명령을 사용하십시오. 자세한 내용은 [prtconf\(1M\)](#) 매뉴얼 페이지 및 [65 페이지](#) “시스템의 제품 이름 표시 방법”을 참조하십시오.

칩 다중 스레드 기능을 확인하는 `psrinfo` 명령 옵션

Oracle Solaris 10: `psrinfo` 명령이 가상 프로세서에 대한 정보에 더하여 실제 프로세서에 대한 정보 또한 제공하도록 수정되었습니다. 이 개선된 기능은 칩 다중 스레드(CMT) 기능을 식별하기 위하여 추가되었습니다. 새 `-p` 옵션을 사용하면 시스템에 있는 실제 프로세서의 총 수가 보고됩니다. `psrinfo -pv` 명령을 사용하면 시스템에 있는 실제 프로세서뿐 아니라 각 실제 프로세서에 연결된 가상 프로세서가 모두 목록으로 표시됩니다. `psrinfo` 명령의 기본 출력에는 시스템의 가상 프로세서 정보가 계속 표시됩니다.

자세한 내용은 [psrinfo\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

이 기능과 연관된 절차에 대한 내용은 [66 페이지](#) “시스템의 물리적 프로세서 유형 표시 방법”을 참조하십시오.

새로운 `localeadm` 명령

Oracle Solaris 10: 새로운 `localeadm` 명령을 사용하면 OS를 재설치하거나 수동으로 패키지를 추가/제거하지 않고도 시스템의 로케일을 변경할 수 있습니다. 또한 이 명령으로 시스템을 질의하여 설치된 로케일을 확인할 수 있습니다. `localeadm` 명령을 실행하려면 슈퍼유저 권한이 있거나 RBAC(역할 기반 액세스 제어)를 통해 동등한 역할을 획득해야 합니다.

자세한 내용은 [localeadm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

이 설명서의 자세한 내용은 [5 장](#), “시스템 정보 표시 및 변경(작업)”을 참조하십시오.

새로운 기능의 전체 목록 및 Oracle Solaris 릴리스에 대한 설명은 [Oracle Solaris 10 8/11 새로운 기능](#)을 참조하십시오.

시스템 리소스 관리(로드맵)

작업	설명	지침
시스템 정보 표시 및 변경	다양한 명령을 사용하여 일반 시스템 정보, 언어 환경, 날짜 및 시간, 시스템의 호스트 이름과 같은 시스템 정보를 표시하고 변경합니다.	5 장, “시스템 정보 표시 및 변경(작업)”
디스크 사용 관리	디스크 공간 사용 현황을 확인하고 단계에 따라 오래된 사용되지 않은 파일을 제거합니다.	6 장, “디스크 사용 관리(작업)”
쿼터 관리	UFS 파일 시스템 쿼터를 사용하여 사용자별 활용되는 디스크 공간량을 관리합니다.	7 장, “UFS 쿼터 관리(작업)”
시스템 이벤트 일정 잡기	<code>cron</code> 및 <code>at</code> 작업을 사용하여 시스템 루틴 일정을 잡아서 오래된 사용되지 않은 파일을 정리할 수 있습니다.	8 장, “시스템 작업 예약(작업)”
시스템 계산 관리	시스템 계산을 사용하여 시스템 리소스를 사용하는 사용자 및 응용 프로그램을 식별합니다.	9 장, “시스템 계산 관리(작업)”
Solaris Resource Management로 시스템 리소스 관리	리소스 관리자를 사용하여 응용 프로그램이 시스템 리소스를 사용하는 방법을 제어하고 리소스 사용량을 추적하여 요금을 부과합니다.	시스템 관리 설명서: Oracle Solaris Containers-리소스 관리 및 Oracle Solaris 영역의 1 장, “Solaris 10 리소스 관리 소개”

시스템 정보 표시 및 변경(작업)

이 장에서는 가장 일반적인 시스템 정보를 표시하고 변경하는 데 필요한 작업에 대해 설명합니다.

시스템 정보 표시 및 변경과 관련된 절차에 대한 자세한 내용은 다음을 참조하십시오.

- 59 페이지 “시스템 정보 표시(작업 맵)”
- 68 페이지 “시스템 정보 변경(작업 맵)”

시스템 리소스 관리에 대한 개요 정보는 4 장, “시스템 리소스 관리(개요)”를 참조하십시오.

시스템 정보 표시(작업 맵)

작업	설명	수행 방법
시스템에서 32비트 기능을 사용할 수 있는지 아니면 64비트 기능을 사용할 수 있는지 확인합니다.	isainfo 명령을 사용하여 시스템에서 32비트 기능을 사용할 수 있는지 아니면 64비트 기능을 사용할 수 있는지 확인합니다. x86 기반 시스템의 경우 isalist 명령을 사용하여 이 정보를 표시할 수 있습니다.	61 페이지 “시스템에서 32비트 기능을 사용할 수 있는지 아니면 64비트 기능을 사용할 수 있는지 확인하는 방법”
Oracle Solaris 릴리스 정보를 표시합니다.	/etc/release 파일의 콘텐츠를 표시하여 Oracle Solaris 릴리스 버전을 식별합니다.	63 페이지 “Oracle Solaris 릴리스 정보 표시 방법”
일반적인 시스템 정보를 표시합니다.	showrev 명령을 사용하여 일반적인 시스템 정보를 표시합니다.	64 페이지 “일반적인 시스템 정보 표시 방법”

작업	설명	수행 방법
시스템의 호스트 ID 번호를 표시합니다.	hostid 명령을 사용하여 시스템의 호스트 ID를 표시합니다.	64 페이지 “시스템의 호스트 ID 번호 표시 방법”
시스템의 제품 이름을 표시합니다.	Solaris 10 1/06 릴리스부터는 prtconf -b 명령을 사용하여 시스템의 제품 이름을 표시할 수 있습니다.	65 페이지 “시스템의 제품 이름 표시 방법”
시스템에 설치된 메모리를 표시합니다.	prtconf 명령을 사용하여 시스템에 설치된 메모리에 대한 정보를 표시합니다.	65 페이지 “시스템에 설치된 메모리 표시 방법”
시스템의 날짜 및 시간을 표시합니다.	date 명령을 사용하여 시스템의 날짜 및 시간을 표시합니다.	66 페이지 “날짜 및 시간 표시 방법”
시스템의 물리적 프로세서 유형을 표시합니다.	psrinfo -p 명령을 사용하여 시스템의 총 물리적 프로세서 수를 나열합니다. psrinfo -pv 명령을 사용하여 시스템의 모든 물리적 프로세서와 각 물리적 프로세서에 연결된 가상 프로세서를 나열합니다.	66 페이지 “시스템의 물리적 프로세서 유형 표시 방법”
시스템의 논리적 프로세서 유형을 표시합니다.	psrinfo -v 명령을 사용하여 시스템의 논리적 프로세서 유형을 표시합니다.	66 페이지 “시스템의 논리적 프로세서 유형 표시 방법”
시스템에 설치된 로케일을 표시합니다.	localeadm 명령을 사용하여 시스템에 설치된 로케일을 표시합니다.	67 페이지 “시스템에 설치된 로케일 표시 방법”
시스템에 로케일이 설치되었는지 여부를 확인합니다.	localeadm 명령의 -q 옵션 및 로케일을 사용하여 시스템에 로케일이 설치되었는지 여부를 확인합니다.	68 페이지 “시스템에 로케일이 설치되었는지 여부 확인 방법”

시스템 정보 표시

다음 표에서는 일반적인 시스템 정보 표시에 사용할 수 있는 명령에 대해 설명합니다.

표 5-1 시스템 정보 표시 명령

명령	표시되는 시스템 정보	매뉴얼 페이지
date	날짜와 시간	date(1)

표 5-1 시스템 정보 표시 명령 (계속)

명령	표시되는 시스템 정보	매뉴얼 페이지
hostid	호스트 ID 번호	hostid(1)
isainfo	실행 중인 시스템에서 고유 응용 프로그램이 지원하며 스크립트에 토큰으로 전달될 수 있는 비트 수	isainfo(1)
isalist	x86 기반 시스템에 대한 프로세서 유형	psrinfo(1M)
localeadm	시스템에 설치된 로케일	localeadm(1M)
prtconf	시스템 구성 정보, 설치된 메모리 및 제품 이름	prtconf(1M)
psrinfo	프로세서 유형	psrinfo(1M)
showrev	호스트 이름, 호스트 ID, 릴리스, 커널 구조, 응용 프로그램 구조, 하드웨어 공급자, 도메인 및 커널 버전	showrev(1M)
uname	운영 체제 이름, 릴리스, 버전, 노드 이름, 하드웨어 이름 및 프로세서 유형	uname(1)

▼ 시스템에서 32비트 기능을 사용할 수 있는지 아니면 64비트 기능을 사용할 수 있는지 확인하는 방법

- **isainfo** 명령을 사용하여 시스템에서 32비트 기능을 사용할 수 있는지 아니면 64비트 기능을 사용할 수 있는지 확인합니다.

isainfo options

옵션 지정 없이 실행되는 **isainfo** 명령은 현재 OS 버전에서 지원하는 응용 프로그램에 대한 고유 명령 세트의 이름을 표시합니다.

- v 기타 옵션에 대한 자세한 정보를 인쇄합니다.
- b 고유 명령 세트의 주소 공간에 비트 수를 인쇄합니다.
- n 현재 OS 버전에서 지원하는 이동식 응용 프로그램에 사용되는 고유 명령 세트의 이름을 인쇄합니다.
- k OS 커널 구성 요소(예: 장치 드라이버 및 STREAMS 모듈)에서 사용하는 명령 세트의 이름을 인쇄합니다.

주 - x86 기반 시스템의 경우 **isalist** 명령을 통해서도 이 정보를 인쇄할 수 있습니다.

자세한 내용은 **isalist(1)** 매뉴얼 페이지를 참조하십시오.

예 5-1 SPARC: 시스템에서 32비트 기능을 사용할 수 있는지 아니면 64비트 기능을 사용할 수 있는지 확인

32비트 커널을 사용하는 Oracle Solaris OS의 이전 릴리스가 실행되고 있는 UltraSPARC 시스템에 대한 `isainfo` 명령 출력은 다음과 같이 표시됩니다.

```
$ isainfo -v
32-bit sparc applications
```

이 출력은 해당 시스템이 32비트 응용 프로그램만 지원할 수 있음을 나타냅니다.

현재 Oracle Solaris OS 릴리스는 SPARC 기반 시스템에만 64비트 커널을 제공합니다. 64비트 커널이 실행되고 있는 UltraSPARC 시스템에 대한 `isainfo` 명령 출력은 다음과 같이 표시됩니다.

```
$ isainfo -v
64-bit sparcv9 applications
32-bit sparc applications
```

이 출력은 해당 시스템이 32비트 응용 프로그램과 64비트 응용 프로그램을 모두 지원할 수 있음을 나타냅니다.

`isainfo -b` 명령을 사용하여 실행 중인 시스템의 고유 응용 프로그램에서 지원하는 비트 수를 표시할 수 있습니다.

32비트 Oracle Solaris OS가 실행되고 있는 SPARC 기반, x86 기반 또는 UltraSPARC 시스템의 출력은 다음과 같이 표시됩니다.

```
$ isainfo -b
32
```

64비트 Oracle Solaris OS가 실행되고 있는 64비트 UltraSPARC 시스템의 `isainfo` 명령 출력은 다음과 같이 표시됩니다.

```
$ isainfo -b
64
```

명령은 64만 반환합니다. 64비트 UltraSPARC 시스템이 두 유형의 응용 프로그램을 모두 실행할 수 있기는 하지만 64비트 시스템에서는 64비트 응용 프로그램을 실행하는 것이 가장 좋습니다.

예 5-2 x86: 시스템에서 32비트 기능을 사용할 수 있는지 아니면 64비트 기능을 사용할 수 있는지 확인

64비트 커널이 실행되고 있는 x86 기반 시스템에 대한 `isainfo` 명령 출력은 다음과 같이 표시됩니다.

```
$ isainfo
amd64 i386
```

이 출력은 해당 시스템이 64비트 응용 프로그램을 지원할 수 있음을 나타냅니다.

`isainfo -v` 명령을 사용하여 x86 기반 시스템이 32비트 커널을 실행할 수 있는지 여부를 확인할 수 있습니다.

```
$ isainfo -v
64-bit amd64 applications
    fpu tsc cx8 cmov mmx ammx a3dnow a3dnowx fxsr sse sse2
32-bit i386 applications
    fpu tsc cx8 cmov mmx ammx a3dnow a3dnowx fxsr sse sse2
```

이 출력은 해당 시스템이 64비트 응용 프로그램과 32비트 응용 프로그램을 모두 지원할 수 있음을 나타냅니다.

`isainfo -b` 명령을 사용하여 실행 중인 시스템의 고유 응용 프로그램에서 지원하는 비트 수를 표시할 수 있습니다.

32비트 Oracle Solaris OS가 실행되고 있는 x86 기반 시스템의 출력은 다음과 같이 표시됩니다.

```
$ isainfo -b
32
```

64비트 Oracle Solaris OS가 실행되고 있는 x86 기반 시스템의 `isainfo` 명령 출력은 다음과 같이 표시됩니다.

```
$ isainfo -b
64
```

`isalist` 명령을 통해서도 x86 기반 시스템이 32비트 모드로 실행 중인지 아니면 64비트 모드로 실행 중인지 확인할 수 있습니다.

```
$ isalist
amd64 pentium_pro+mmx pentium_pro pentium+mmx pentium i486 i386 i86
```

위 예에서 `amd64`는 시스템에서 64비트 기능을 사용할 수 있음을 나타냅니다.

▼ Oracle Solaris 릴리스 정보 표시 방법

- `/etc/release` 파일의 콘텐츠를 표시하여 릴리스 버전을 식별합니다.

```
$ cat /etc/release
Oracle Solaris 10 s10_51 SPARC
Copyright (c) 1983, 2011, Oracle and/or its affiliates. All Rights Reserved.
Use is subject to license terms.
Assembled 28 February 20011
```

▼ 일반적인 시스템 정보 표시 방법

- 일반적인 시스템 정보를 표시하려면 **showrev** 명령을 사용합니다.

```
$ showrev options
```

- a 사용 가능한 모든 시스템 수정 정보를 인쇄합니다.
- c (command) 명령에 대한 수정 정보를 인쇄합니다.
- p 패치에 대한 수정 정보만 인쇄합니다.
- R (root_path) root_path로 사용되도록 디렉토리의 전체 경로 이름을 정의합니다.
- s (host name) 지정된 호스트 이름에 대해 이 작업을 수행합니다.
- w OpenWindows 수정 정보만 인쇄합니다.

uname 명령을 통해서도 시스템 정보를 표시할 수 있습니다. 다음 예에서는 uname 명령 출력을 보여 줍니다. -a 옵션은 운영 체제 이름, 시스템 노드 이름, 운영 체제 릴리스, 운영 체제 버전, 하드웨어 이름 및 프로세서 유형을 표시합니다.

```
$ uname
SunOS
$ uname -a
SunOS starbug 5.10 Generic sun4u sparc SUNW,Ultra-5_10
$
```

예 5-3 일반적인 시스템 정보 표시

다음 예에서는 showrev 명령 출력을 보여 줍니다. -a 옵션은 사용 가능한 모든 시스템 정보를 표시합니다.

```
$ showrev -a
Hostname: stonetouch
Hostid: 8099dfb9
Release: 5.10
Kernel architecture: sun4u
Application architecture: sparc
Hardware provider:
Domain:
Kernel version: SunOS 5.10 s10_46

OpenWindows version:
Solaris X11 Version 6.6.2 19 November 2010
No patches are installed
```

▼ 시스템의 호스트 ID 번호 표시 방법

- 호스트 ID 번호를 16진수 형식으로 표시하려면 **hostid** 명령을 사용합니다.

예 5-4 시스템의 호스트 ID 번호 표시

다음 예에서는 `hostid` 명령의 샘플 출력을 보여 줍니다.

```
$ hostid
80a5d34c
```

▼ 시스템의 제품 이름 표시 방법

Solaris 10 1/06: `prtconf` 명령에 `-b` 옵션을 지정하여 시스템의 제품 이름을 표시할 수 있습니다. 이 기능에 대한 자세한 내용은 [prtconf\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

- 시스템의 제품 이름을 표시하려면 다음과 같이 `-b` 옵션과 함께 `prtconf` 명령을 사용합니다.

```
% prtconf -b
```

예 5-5 시스템의 제품 이름 표시

다음 예에서는 `prtconf -b` 명령의 샘플 출력을 보여 줍니다.

```
$ prtconf -b
name: SUNW,Ultra-5 10
model: SUNW,375-0066
banner-name: Sun Ultra 5/10 UPA/PCI (UltraSPARC-IIi 333MHz)
```

다음 예에서는 `prtconf -vb` 명령의 샘플 출력을 보여 줍니다.

```
$ prtconf -vb
name: SUNW,Ultra-5 10
model: SUNW,375-0066
banner-name: Sun Ultra 5/10 UPA/PCI (UltraSPARC-IIi 333MHz)
idprom: 01800800.20a6c363.00000000.a6c363a9.00000000.00000000.405555aa.aa555500
openprom model: SUNW,3.15
openprom version: 'OBP 3.15.2 1998/11/10 10:35'
```

▼ 시스템에 설치된 메모리 표시 방법

- 시스템에 설치된 메모리를 표시하려면 `prtconf` 명령을 사용합니다.

예 5-6 시스템에 설치된 메모리 표시

다음 예에서는 `prtconf` 명령의 샘플 출력을 보여 줍니다. `grep Memory` 명령은 메모리 정보만 표시하기 위해 `prtconf` 명령의 출력을 선택합니다.

```
$ prtconf | grep Memory
Memory size: 128 Megabytes
```

▼ 날짜 및 시간 표시 방법

- 시스템 클럭에 따라 현재 날짜 및 시간을 표시하려면 **date** 명령을 사용합니다.

예 5-7 날짜 및 시간 표시

다음 예에서는 date 명령의 샘플 출력을 보여 줍니다.

```
$ date
Wed Jan 21 17:32:59 MST 2004
$
```

▼ 시스템의 물리적 프로세서 유형 표시 방법

- **psrinfo -p** 명령을 사용하여 시스템의 총 물리적 프로세서 수를 표시합니다.

```
$ psrinfo -p
1
```

psrinfo -pv 명령을 사용하여 시스템의 각 물리적 프로세서 및 각 물리적 프로세서에 연결된 가상 프로세서에 대한 정보를 표시합니다.

```
$ psrinfo -pv
The UltraSPARC-IV physical processor has 2 virtual processors (8, 520)
The UltraSPARC-IV physical processor has 2 virtual processors (9, 521)
The UltraSPARC-IV physical processor has 2 virtual processors (10, 522)
The UltraSPARC-IV physical processor has 2 virtual processors (11, 523)
The UltraSPARC-III+ physical processor has 1 virtual processor (16)
The UltraSPARC-III+ physical processor has 1 virtual processor (17)
The UltraSPARC-III+ physical processor has 1 virtual processor (18)
The UltraSPARC-III+ physical processor has 1 virtual processor (19)
```

x86 기반 시스템에서 psrinfo -pv 명령을 사용할 경우 다음 출력이 표시됩니다.

```
$ psrinfo -pv
The i386 physical processor has 2 virtual processors (0, 2)
The i386 physical processor has 2 virtual processors (1, 3)
```

▼ 시스템의 논리적 프로세서 유형 표시 방법

- **psrinfo -v** 명령을 사용하여 시스템의 프로세서 유형에 대한 정보를 표시합니다.

```
$ psrinfo -v
```

x86 기반 시스템에서 isalist 명령을 사용하여 가상 프로세서 유형을 표시합니다.

```
$ isalist
```

예 5-8 SPARC: 시스템의 프로세서 유형 표시

다음 예에서는 SPARC 기반 시스템의 프로세서 유형에 대한 정보를 표시하는 방법을 보여 줍니다.

```
$ psrinfo -v
Status of virtual processor 0 as of: 04/16/2004 10:32:13
  on-line since 03/22/2004 19:18:27.
  The sparcv9 processor operates at 650 MHz,
  and has a sparcv9 floating point processor.
```

예 5-9 x86: 시스템의 프로세서 유형 표시

다음 예에서는 x86 기반 시스템의 프로세서 유형에 대한 정보를 표시하는 방법을 보여 줍니다.

```
$ isalist
pentium_pro+mmx pentium_pro pentium+mmx pentium i486 i386 i86
```

▼ 시스템에 설치된 로케일 표시 방법

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 localeadm 명령을 사용하여 시스템에 현재 설치된 로케일을 표시합니다. -l 옵션은 시스템에 설치된 로케일을 표시합니다. 예를 들어, 다음과 같습니다.

```
# localeadm -l
Checking for installed pkgs. This could take a while.

Checking for Australasia region (aua)
(1of2 pkgs)
|.....|
.
:
.
The following regions are installed on concordance on Wed Dec 17 15:13:00 MST 2003

POSIX (C)

Central Europe (ceu)
[ Austria, Czech Republic, Germany, Hungary, Poland, Slovakia,
Switzerland (German), Switzerland (French) ]

Done.
```

▼ 시스템에 로케일이 설치되었는지 여부 확인 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 `localeadm` 명령을 사용하여 시스템에 로케일이 설치되었는지 여부를 확인합니다. `-q` 옵션 및 로케일은 시스템에서 해당 로케일이 시스템에 설치되었는지 여부를 확인하는 질의를 수행합니다. 예를 들어, 시스템에 중앙 유럽어 영역(ceu)이 설치되었는지 여부를 확인하려면 다음과 같이 합니다.

```
# localeadm -q ceu
locale/region name is ceu
Checking for Central Europe region (ceu)
.
.
.
The Central Europe region (ceu) is installed on this system
```

시스템 정보 변경(작업 맵)

작업	지침	수행 방법
시스템의 날짜 및 시간을 수동으로 설정합니다.	<code>date mmddHHMM[[cc]yy]</code> 명령줄 구문을 사용하여 시스템의 날짜 및 시간을 수동으로 설정합니다.	69 페이지 “시스템의 날짜 및 시간을 수동으로 설정하는 방법”
오늘의 메시지를 설정합니다.	<code>/etc/motd</code> 파일을 편집하여 시스템에서 오늘의 메시지를 설정합니다.	70 페이지 “오늘의 메시지 설정 방법”

작업	지침	수행 방법
시스템의 호스트 이름을 변경합니다.	<p>다음 줄을 편집하여 시스템의 호스트 이름을 변경합니다.</p> <ul style="list-style-type: none"> ■ /etc/nodename ■ /etc/hostname.*host-name ■ /etc/inet/hosts <p>주 - Solaris 3/05, 1/06, 6/06 또는 11/06 릴리스를 실행 중인 경우 /etc/inet/ipnodes 파일도 업데이트해야 합니다. Solaris 10 8/07 릴리스부터는 OS에서 더 이상 두 개의 별도 hosts 파일이 사용되지 않습니다. 대신 /etc/inet/hosts 파일이 IPv4 항목과 IPv6 항목을 모두 포함하는 하나의 hosts 파일로 사용됩니다.</p>	71 페이지 “시스템의 호스트 이름 변경 방법”
시스템에 로케일을 추가합니다.	localeadm 명령을 사용하여 시스템에 로케일을 추가합니다.	시스템에 로케일을 추가하는 방법
시스템에서 로케일을 제거합니다.	localeadm 명령의 -r 옵션 및 로케일을 사용하여 시스템에서 로케일을 제거합니다.	시스템에서 로케일을 제거하는 방법

시스템 정보 변경

이 단원에서는 일반적인 시스템 정보 변경에 사용할 수 있는 명령에 대해 설명합니다.

▼ 시스템의 날짜 및 시간을 수동으로 설정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 새 날짜 및 시간을 입력합니다.

```
# date mmddHHMM[[cc]yy]
```

mm 두 자리의 월입니다.

dd 두 자리의 날짜입니다.

HH 두 자리의 24시간 형식 시간입니다.

MM 두 자리의 분입니다.

cc 두 자리의 세기입니다.

yy 두 자리의 연도입니다.

자세한 내용은 [date\(1\)](#) 매뉴얼 페이지를 참조하십시오.

- 3 옵션 없이 `date` 명령을 사용하여 시스템의 날짜를 제대로 재설정했는지 확인합니다.

예 5-10 시스템의 날짜 및 시간 수동 설정

다음 예에서는 `date` 명령을 사용하여 시스템의 날짜 및 시간을 수동으로 설정하는 방법을 보여 줍니다.

```
# date
Wed Mar 3 14:04:19 MST 2004
# date 0121173404
Thu Jan 21 17:34:34 MST 2004
```

▼ 오늘의 메시지 설정 방법

사용자가 로그인할 때 시스템의 모든 사용자에게 표시될 공지 또는 문의가 포함되도록 오늘의 메시지 파일인 `/etc/motd`를 편집합니다. 이 기능을 자주 사용하지 않을 경우 해당 파일을 정기적으로 편집하여 오래된 메시지를 제거합니다.

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 `/etc/motd` 파일을 편집하고 선택한 메시지를 추가합니다.

사용자 로그인 시 표시될 메시지가 포함되도록 텍스트를 편집합니다. 공백, 탭 및 캐리지 리턴을 포함합니다.

- 3 `/etc/motd` 파일의 콘텐츠를 표시하여 변경 사항을 확인합니다.

```
$ cat /etc/motd
Welcome to the UNIX Universe. Have a nice day.
```

예 5-11 오늘의 메시지 설정

Oracle Solaris 소프트웨어를 설치할 때 기본적으로 제공되는 오늘의 메시지에는 버전 정보가 포함됩니다.

```
$ cat /etc/motd
Oracle Corporation SunOS 5.10 Generic January 2005
```

다음 예에서는 로그인한 각 사용자에게 시스템 가용성에 대한 정보를 제공하는 편집된 `/etc/motd` 파일을 보여 줍니다.

```
$ cat /etc/motd
The system will be down from 7:00 a.m to 2:00 p.m. on
Saturday, July 7, for upgrades and maintenance.
Do not try to access the system during those hours.
Thank you.
```

▼ 시스템의 호스트 이름 변경 방법

시스템의 호스트 이름은 여러 위치에서 지정됩니다.

따라서 새 호스트 이름이 반영되도록 이름 서비스 데이터베이스를 업데이트해야 합니다.

다음 절차에 따라 시스템의 호스트 이름을 변경하거나 이름을 바꿀 수 있습니다.

`sys-unconfig` 명령을 통해서도 호스트 이름 등의 시스템 재구성을 수행할 수 있습니다. 자세한 내용은 [sys-unconfig\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “[Configuring RBAC \(Task Map\)](#)”를 참조하십시오.

2 다음 파일에서 시스템의 호스트 이름을 변경합니다.

- `/etc/nodename`
- `/etc/hostname.*interface`
- `/etc/inet/hosts`
- `/etc/inet/ipnodes`(일부 릴리스에만 적용)

주 - Solaris 10 8/07 릴리스부터는 더 이상 두 개의 별도 `hosts` 파일이 사용되지 않습니다. 대신 `/etc/inet/hosts` 파일이 IPv4 항목과 IPv6 항목을 모두 포함하는 하나의 `hosts` 파일로 사용됩니다. 따라서 항상 동기화해야 하는 두 개의 `hosts` 파일에서 IPv4 항목을 유지 관리하지 않아도 됩니다. 역방향 호환성을 위해 `/etc/inet/ipnodes` 파일은 `/etc/inet/hosts` 파일에 대한 동일한 이름의 심볼릭 링크로 대체됩니다. 자세한 내용은 [hosts\(4\)](#) 매뉴얼 페이지를 참조하십시오.

3 (옵션) 이름 서비스를 사용 중인 경우 `hosts` 파일에서 시스템의 호스트 이름을 변경합니다.

- 4 /var/crash 디렉토리에서 호스트 이름 디렉토리의 이름을 바꿉니다.

```
# cd /var/crash
# mv old-host-name new-host-name
```

- 5 새 호스트 이름이 활성화되도록 시스템을 재부트합니다.

```
# init 6
```

▼ 시스템에 로케일을 추가하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 `localeadm` 명령을 사용하여 시스템에 설치할 로케일에 대한 패키지를 추가합니다.

`-a` 옵션 및 로케일은 추가할 로케일을 식별합니다. `-d` 옵션 및 장치는 추가할 로케일 패키지가 포함된 장치를 식별합니다. 예를 들어, 시스템에 중앙 유럽어 영역(ceu)을 추가하려면 다음과 같이 합니다.

```
# localeadm -a ceu -d /net/install/latest/Solaris/Product
```

```
locale/region name is ceu
```

```
Devices are /net/install/latest/Solaris/Product
```

```
.
.
```

```
One or more locales have been added.
```

```
To update the list of locales available at
```

```
.
.
```

▼ 시스템에서 로케일을 제거하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 localeadm 명령을 사용하여 시스템에 설치된 로케일에 대한 패키지를 제거합니다.

-r 옵션 및 로케일은 시스템에서 제거할 로케일을 식별합니다. 예를 들어, 시스템에서 중앙 유럽어 영역(ceu)을 제거하려면 다음과 같이 합니다.

```
# localeadm -r ceu
locale/region name is ceu
Removing packages for Central Europe (ceu)
.
.
.
One or more locales have been removed.
To update the list of locales available
at the login screen's "Options->Language" menu,
.
.
.
```


◆ ◆ ◆ 6 장

디스크 사용 관리(작업)

이 장에서는 사용되지 않은 파일 및 큰 디렉토리를 찾아 디스크 공간을 최적화하는 방법에 대해 설명합니다.

디스크 사용 관리와 관련된 절차에 대한 자세한 내용은 75 페이지 “디스크 사용 관리(작업 맵)”를 참조하십시오.

디스크 사용 관리(작업 맵)

작업	설명	수행 방법
파일 및 디스크 공간에 대한 정보를 표시합니다.	df 명령을 사용하여 디스크 공간 사용 방식에 대한 정보를 표시합니다.	77 페이지 “파일 및 디스크 공간 정보 표시 방법”
파일 크기를 표시합니다.	-lh 옵션과 함께 ls 명령을 사용하여 파일 크기에 대한 정보를 표시합니다.	79 페이지 “파일 크기 표시 방법”
큰 파일을 찾습니다.	ls -s 명령을 사용하여 파일을 크기에 따라 내림차순으로 정렬할 수 있습니다.	80 페이지 “큰 파일을 찾는 방법”
지정된 크기 제한을 초과하는 파일을 찾습니다.	-size 옵션 및 지정된 크기 제한 값과 함께 find 명령을 사용하여 지정된 크기를 초과하는 파일을 찾아 파일 이름을 표시합니다.	81 페이지 “지정된 크기 제한을 초과하는 파일을 찾는 방법”
디렉토리, 하위 디렉토리 및 파일의 크기를 표시합니다.	du 명령을 사용하여 하나 이상의 디렉토리, 하위 디렉토리 및 파일의 크기를 표시합니다.	82 페이지 “디렉토리, 하위 디렉토리 및 파일의 크기 표시 방법”

작업	설명	수행 방법
로컬 UFS 파일 시스템의 소유권을 표시합니다.	<code>quot -a</code> 명령을 사용하여 파일 소유권을 표시합니다.	83 페이지 “로컬 UFS 파일 시스템의 사용자 소유권 표시 방법”
최신 파일을 나열합니다.	<code>ls -t</code> 명령을 사용하여 가장 최근에 생성되었거나 변경된 파일을 먼저 표시합니다.	85 페이지 “최신 파일 나열 방법”
오래된 파일 또는 비활성 파일을 찾아 제거합니다.	<code>-atime</code> 및 <code>-mtime</code> 옵션과 함께 <code>find</code> 명령을 사용하여 지정된 일 수 동안 액세스되지 않은 파일을 찾습니다. <code>rm 'cat filename'</code> 명령을 사용하여 해당 파일을 제거할 수 있습니다.	85 페이지 “오래된 파일 또는 비활성 파일 찾기 및 제거 방법”
임시 디렉토리를 지웁니다.	임시 디렉토리를 찾은 다음 <code>rm -r *</code> 명령을 사용하여 전체 디렉토리를 제거합니다.	86 페이지 “임시 디렉토리를 지우는 방법”
코어 파일을 찾아 삭제합니다.	<code>find . -name core -exec rm {} \;</code> 명령을 사용하여 코어 파일을 찾은 후 삭제합니다.	87 페이지 “코어 파일 찾기 및 삭제 방법”
충돌 덤프 파일을 삭제합니다.	<code>rm *</code> 명령을 사용하여 <code>/var/crash/</code> 디렉토리에 있는 충돌 덤프 파일을 삭제합니다.	88 페이지 “충돌 덤프 파일 삭제 방법”

파일 및 디스크 공간 정보 표시

다음 표는 파일 크기 및 디스크 공간 정보 표시에 사용할 수 있는 명령을 요약한 것입니다.

명령	설명	매뉴얼 페이지
<code>df</code>	사용 가능한 디스크 블록 및 파일 수를 보고합니다.	df(1M)
<code>du</code>	각 하위 디렉토리에 할당된 디스크 공간을 요약합니다.	du(1)
<code>find -size</code>	<code>-size</code> 옵션과 함께 지정된 크기를 기반으로 디렉토리를 반복적으로 검색합니다.	find(1)
<code>ls -lh</code>	파일 크기를 1024바이트로 나열합니다.	ls(1)

▼ 파일 및 디스크 공간 정보 표시 방법

- **df** 명령을 사용하여 디스크 공간 사용 방식에 대한 정보를 표시합니다.

```
$ df [directory] [-h] [-t]
```

df 옵션 없이 마운트된 모든 파일 시스템과 해당 장치 이름, 사용된 512바이트 블록 수, 파일 수를 나열합니다.

directory 파일 시스템을 확인할 디렉토리를 지정합니다.

-h 디스크 공간을 1024바이트로 표시합니다.

-t 마운트된 모든 파일 시스템에 사용된 블록 수와 함께 총 블록 수를 표시합니다.

예 6-1 파일 크기 및 디스크 공간 정보 표시

다음 예에서는 /usr/dist를 제외하고 로컬에서 마운트된 모든 파일 시스템을 나열합니다.

```
$ df
/ (/dev/dsk/c0t0d0s0 ): 101294 blocks 105480 files
/devices (/devices ): 0 blocks 0 files
/system/contract (ctfs ): 0 blocks 2147483578 files
/proc (proc ): 0 blocks 1871 files
/etc/mnttab (mnttab ): 0 blocks 0 files
/etc/svc/volatile (swap ): 992704 blocks 16964 files
/system/object (objfs ): 0 blocks 2147483530 files
/usr (/dev/dsk/c0t0d0s6 ): 503774 blocks 299189 files
/dev/fd (fd ): 0 blocks 0 files
/var/run (swap ): 992704 blocks 16964 files
/tmp (swap ): 992704 blocks 16964 files
/opt (/dev/dsk/c0t0d0s5 ): 23914 blocks 6947 files
/export/home (/dev/dsk/c0t0d0s7 ): 16810 blocks 7160 files
```

예 6-2 UFS 루트 파일 시스템이 설치된 시스템에 대한 파일 크기 정보를 1024바이트로 표시

다음 예에서는 UFS 루트 파일 시스템이 설치된 시스템에 대한 파일 시스템 정보가 1024바이트로 표시됩니다.

```
$ df -h
Filesystem size used avail capacity Mounted on
/dev/dsk/c0t0d0s0 249M 200M 25M 90% /
/devices 0K 0K 0K 0% /devices
ctfs 0K 0K 0K 0% /system/contract
proc 0K 0K 0K 0% /proc
mnttab 0K 0K 0K 0% /etc/mnttab
swap 485M 376K 485M 1% /etc/svc/volatile
objfs 0K 0K 0K 0% /system/object
```

/dev/dsk/c0t0d0s6	3.2G	2.9G	214M	94%	/usr
fd	0K	0K	0K	0%	/dev/fd
swap	485M	40K	485M	1%	/var/run
swap	485M	40K	485M	1%	/tmp
/dev/dsk/c0t0d0s5	13M	1.7M	10M	15%	/opt
/dev/dsk/c0t0d0s7	9.2M	1.0M	7.3M	13%	/export/home

/proc 및 /tmp는 로컬 파일 시스템이기는 하지만 UFS 파일 시스템이 아닙니다. /proc는 PROCFS 파일 시스템, /var/run 및 /tmp는 TMPFS 파일 시스템, /etc/mnttab는 MNTFS 파일 시스템입니다.

예 6-3 ZFS 루트 파일 시스템이 설치된 시스템에 대한 파일 크기 정보를 1024비트로 표시

다음 예에서는 Oracle Solaris ZFS 루트 파일 시스템이 설치된 시스템에 대한 파일 시스템 정보가 1024비트로 표시됩니다.

Filesystem	size	used	avail	capacity	Mounted on
rpool/ROOT/s1008be	67G	4.6G	58G	8%	/
/devices	0K	0K	0K	0%	/devices
ctfs	0K	0K	0K	0%	/system/contract
proc	0K	0K	0K	0%	/proc
mnttab	0K	0K	0K	0%	/etc/mnttab
swap	1.9G	1.5M	1.9G	1%	/etc/svc/volatile
objfs	0K	0K	0K	0%	/system/object
sharefs	0K	0K	0K	0%	/etc/dfs/sharetab
/platform/sun4u-us3/lib/libc_psr/libc_psr_hwcapl.so.1					
	63G	4.6G	58G	8%	/platform/sun4u-us3/lib/libc_psr.so.1
/platform/sun4u-us3/lib/sparcv9/libc_psr/libc_psr_hwcapl.so.1					
	63G	4.6G	58G	8%	/platform/sun4u-us3/lib/sparcv9/libc_psr.so.1
fd	0K	0K	0K	0%	/dev/fd
rpool/ROOT/s1008be/var					
	67G	73M	58G	1%	/var
swap	1.9G	32K	1.9G	1%	/tmp
swap	1.9G	40K	1.9G	1%	/var/run
rpool/export	67G	20K	58G	1%	/export
rpool/export/home	67G	18K	58G	1%	/export/home

예 6-4 파일 시스템에 대해 할당된 총 블록 및 파일 수 표시

다음 예에서는 마운트된 모든 파일 시스템, 장치 이름, 사용된 총 512바이트 블록 수 및 파일 수를 나열합니다. 각 2라인 항목의 두번째 라인에는 파일 시스템에 대해 할당된 총 블록 및 파일 수가 표시됩니다.

```

$ df -t
/                (/dev/dsk/c0t0d0s0 ): 101294 blocks 105480 files
                  total: 509932 blocks 129024 files
/devices        (/devices ): 0 blocks 0 files
                  total: 0 blocks 113 files
/system/contract (ctfs ): 0 blocks 2147483578 files
                  total: 0 blocks 69 files
/proc           (proc ): 0 blocks 1871 files
    
```

```

total:      0 blocks      1916 files
/etc/mnttab (mnttab      ):      0 blocks      0 files
total:      0 blocks      1 files
/etc/svc/volatile (swap      ): 992608 blocks 16964 files
total:     993360 blocks 17025 files
/system/object (objfs      ):      0 blocks 2147483530 files
total:      0 blocks      117 files
/usr          (/dev/dsk/c0t0d0s6 ): 503774 blocks 299189 files
total:     6650604 blocks 420480 files
/dev/fd      (fd        ):      0 blocks      0 files
total:      0 blocks      31 files
/var/run     (swap      ): 992608 blocks 16964 files
total:     992688 blocks 17025 files
/tmp        (swap      ): 992608 blocks 16964 files
total:     992688 blocks 17025 files
/opt        (/dev/dsk/c0t0d0s5 ): 23914 blocks 6947 files
total:     27404 blocks 7168 files
/export/home (/dev/dsk/c0t0d0s7 ): 16810 blocks 7160 files
total:     18900 blocks 7168 files

```

파일 크기 확인

ls 명령을 사용하여 파일 크기를 확인하고 정렬할 수 있습니다. find 명령을 사용하여 크기 제한을 초과하는 파일을 찾을 수 있습니다. 자세한 내용은 [ls\(1\)](#) 및 [find\(1\)](#) 매뉴얼 페이지를 참조하십시오.

주 - /var 디렉토리의 공간이 부족할 경우 /var 디렉토리와 디스크 공간이 보다 많은 파일 시스템의 디렉토리 간에 심볼릭 링크를 만들지 마십시오. 만들 경우 임시 조치인 경우에도 특정 데몬 프로세스 및 유틸리티에 문제가 발생할 수 있습니다.

▼ 파일 크기 표시 방법

- 1 확인할 파일이 있는 디렉토리로 변경합니다.
- 2 파일 크기를 표시합니다.

```
$ ls [-lh] [-s]
```

- l 크기가 바이트로 나타나도록 파일 및 디렉토리 목록을 긴 형식으로 표시합니다. 아래에 나오는 예를 참조하십시오.
- h 파일 또는 디렉토리 크기가 1024바이트보다 클 경우 파일 크기 및 디렉토리 크기를 KB, MB, GB 또는 TB로 조정합니다. 또한 이 옵션은 파일 또는 디렉토리 크기가 새 형식으로 표시되도록 -o, -n, -@ 및 -g 옵션으로 표시되는 출력을 수정합니다. 자세한 내용은 [ls\(1\)](#) 매뉴얼 페이지를 참조하십시오.
- s 크기가 블록 수로 나타나도록 파일 및 디렉토리 목록을 표시합니다.

예 6-5 파일 크기 표시

다음 예에서는 lastlog 및 messages 파일이 /var/adm 디렉토리의 다른 파일보다 큰 것을 알 수 있습니다.

```
$ cd /var/adm
$ ls -lh
total 148
drwxrwxr-x  5 adm      adm           512 Nov 26 09:39 acct/
-rw-----  1 uucp     bin           0 Nov 26 09:25 aculog
drwxr-xr-x  2 adm      adm           512 Nov 26 09:25 exacct/
-r--r--r--  1 root     other        342K Nov 26 13:56 lastlog
drwxr-xr-x  2 adm      adm           512 Nov 26 09:25 log/
-rw-r--r--  1 root     root         20K Nov 26 13:55 messages
drwxr-xr-x  2 adm      adm           512 Nov 26 09:25 passwd/
drwxrwxr-x  2 adm      sys           512 Nov 26 09:39 sa/
drwxr-xr-x  2 root     sys           512 Nov 26 09:49 sm.bin/
-rw-rw-rw-  1 root     bin           0 Nov 26 09:25 spellhist
drwxr-xr-x  2 root     sys           512 Nov 26 09:25 streams/
-rw-r--r--  1 root     bin          3.3K Nov 26 13:56 utmpx
-rw-r--r--  1 root     root           0 Nov 26 10:17 vold.log
-rw-r--r--  1 adm      adm           19K Nov 26 13:56 wtmpx
```

다음 예에서는 lpsched.1 파일에 두 개의 블록이 사용됨을 알 수 있습니다.

```
$ cd /var/lp/logs
$ ls -s
total 2          0 lpsched          2 lpsched.1
```

▼ 큰 파일을 찾는 방법

1 검색할 디렉토리로 변경합니다.

2 파일 크기를 블록 수로 내림차순으로 표시합니다.

- 파일에 대한 문자 또는 열이 다를 경우 다음 명령을 사용하여 파일 목록을 블록 크기에 따라 내림차순으로 정렬합니다.

```
$ ls -l | sort +4rn | more
```

이 명령은 목록의 파일을 네번째 필드에 있는 왼쪽 문자부터 정렬합니다.

- 파일에 대한 문자 또는 열이 동일할 경우 다음 명령을 사용하여 파일 목록을 블록 크기에 따라 내림차순으로 정렬합니다.

```
$ ls -s | sort -nr | more
```

이 명령은 목록의 파일을 가장 왼쪽 문자부터 정렬합니다.

예 6-6 큰 파일 찾기(네번째 필드의 문자를 기준으로 정렬)

```
$ cd /var/adm
$ ls -l | sort +4rn | more
-r--r--r-- 1 root root 4568368 Oct 17 08:36 lastlog
-rw-r--r-- 1 adm adm 697040 Oct 17 12:30 pacct.9
-rw-r--r-- 1 adm adm 280520 Oct 17 13:05 pacct.2
-rw-r--r-- 1 adm adm 277360 Oct 17 12:55 pacct.4
-rw-r--r-- 1 adm adm 264080 Oct 17 12:45 pacct.6
-rw-r--r-- 1 adm adm 255840 Oct 17 12:40 pacct.7
-rw-r--r-- 1 adm adm 254120 Oct 17 13:10 pacct.1
-rw-r--r-- 1 adm adm 250360 Oct 17 12:25 pacct.10
-rw-r--r-- 1 adm adm 248880 Oct 17 13:00 pacct.3
-rw-r--r-- 1 adm adm 247200 Oct 17 12:35 pacct.8
-rw-r--r-- 1 adm adm 246720 Oct 17 13:15 pacct.0
-rw-r--r-- 1 adm adm 245920 Oct 17 12:50 pacct.5
-rw-r--r-- 1 root root 190229 Oct 5 03:02 messages.1
-rw-r--r-- 1 adm adm 156800 Oct 17 13:17 pacct
-rw-r--r-- 1 adm adm 129084 Oct 17 08:36 wtmpx
```

예 6-7 큰 파일 찾기(가장 왼쪽 문자를 기준으로 정렬)

다음 예에서는 lastlog 및 messages 파일이 /var/adm 디렉토리에서 가장 큰 파일임을 알 수 있습니다.

```
$ cd /var/adm
$ ls -s | sort -nr | more
48 lastlog
30 messages
24 wtmpx
18 pacct
8 utmpx
2 vold.log
2 sulog
2 sm.bin/
2 sa/
2 passwd/
2 pacct1
2 log/
2 acct/
0 spellhist
0 aculog
total 144
```

▼ 지정된 크기 제한을 초과하는 파일을 찾는 방법

- 지정된 크기를 초과하는 파일을 찾아 파일 이름을 표시하려면 find 명령을 사용합니다.

```
$ find directory -size +nnn
```

directory 검색할 디렉토리를 식별합니다.

-size +nnn 512바이트 블록 수입니다. 이 크기를 초과하는 파일이 나열됩니다.

예 6-8 지정된 크기 제한을 초과하는 파일 찾기

다음 예에서는 현재 작업 디렉토리에서 블록 수가 400개를 초과하는 파일을 찾는 방법을 보여 줍니다. `-print` 옵션은 `find` 명령의 출력을 표시합니다.

```
$ find . -size +400 -print
./Howto/howto.doc
./Howto/howto.doc.backup
./Howto/howtotest.doc
./Routine/routineBackupconcepts.doc
./Routine/routineIntro.doc
./Routine/routineTroublefsck.doc
./record
./Mail/pagination
./Config/configPrintadmin.doc
./Config/configPrintsetup.doc
./Config/configMailappx.doc
./Config/configMailconcepts.doc
./snapshot.rs
```

디렉토리 크기 확인

`du` 명령 및 옵션을 사용하여 디렉토리 크기를 표시할 수 있습니다. 또한 `quot` 명령을 사용하여 로컬 UFS 파일 시스템의 사용자 계정에 사용되는 디스크 공간을 찾을 수 있습니다. 해당 명령에 대한 자세한 내용은 `du(1)` 및 `quot(1M)` 매뉴얼 페이지를 참조하십시오.

▼ 디렉토리, 하위 디렉토리 및 파일의 크기 표시 방법

- `du` 명령을 사용하여 하나 이상의 디렉토리, 하위 디렉토리 및 파일의 크기를 표시합니다. 크기가 512바이트 블록 수로 표시됩니다.

```
$ du [-as] [directory...]
```

<code>du</code>	지정하는 각 디렉토리와 해당 디렉토리에 포함된 각 하위 디렉토리의 크기를 표시합니다.
<code>-a</code>	각 파일 및 하위 디렉토리의 크기와 지정된 디렉토리에 포함된 총 블록 수를 표시합니다.
<code>-s</code>	지정된 디렉토리에 포함된 총 블록 수를 표시합니다.
<code>-h</code>	각 디렉토리의 크기를 1024바이트 블록 수로 표시합니다.
<code>-H</code>	각 디렉토리의 크기를 1000바이트 블록 수로 표시합니다.
<code>[directory...]</code>	확인할 하나 이상의 디렉토리를 식별합니다. 명령줄 구문의 여러 디렉토리를 공백으로 구분합니다.

예 6-9 디렉토리, 하위 디렉토리 및 파일의 크기 표시

다음 예에서는 두 개 디렉토리의 크기를 보여 줍니다.

```
$ du -s /var/adm /var/spool/lp
130    /var/adm
40     /var/spool/lp
```

다음 예에서는 두 개 디렉토리의 크기를 보여 주며 각 디렉토리에 포함된 모든 하위 디렉토리와 파일의 크기를 함께 나열합니다. 각 디렉토리에 포함된 총 블록 수도 표시됩니다.

```
$ du /var/adm /var/spool/lp
2      /var/adm/exacct
2      /var/adm/log
2      /var/adm/streams
2      /var/adm/acct/fiscal
2      /var/adm/acct/nite
2      /var/adm/acct/sum
8      /var/adm/acct
2      /var/adm/sa
2      /var/adm/sm.bin
258    /var/adm
4      /var/spool/lp/admins
2      /var/spool/lp/requests/printing...
4      /var/spool/lp/requests
4      /var/spool/lp/system
2      /var/spool/lp/fifos
24     /var/spool/lp
```

다음 예에서는 디렉토리 크기를 1024바이트 블록 수로 보여 줍니다.

```
$ du -h /usr/share/audio
796K   /usr/share/audio/samples/au
797K   /usr/share/audio/samples
798K   /usr/share/audio
```

▼ 로컬 UFS 파일 시스템의 사용자 소유권 표시 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 사용자, 디렉토리 또는 파일 시스템 및 사용된 1024바이트 블록 수를 표시합니다.

```
# quot [-a] [filesystem ...]
```

-a 마운트된 각 UFS 파일 시스템의 모든 사용자 및 사용된 1024바이트 블록 수를 나열합니다.

filesystem UFS 파일 시스템을 식별합니다. 해당 파일 시스템에 대한 사용자 및 사용된 블록 수가 표시됩니다.

주-quot 명령은 로컬 UFS 파일 시스템에서만 작동합니다.

예 6-10 로컬 UFS 파일 시스템의 사용자 소유권 표시

다음 예에서는 루트(/) 파일 시스템의 사용자가 표시됩니다. 다음 예에서는 마운트된 모든 UFS 파일 시스템의 사용자가 표시됩니다.

```
# quot /
/dev/rdisk/c0t0d0s0:
43340  root
3142   rimmer
47     uucp
35     lp
30     adm
4      bin
4      daemon

# quot -a
/dev/rdisk/c0t0d0s0 (/):
43340  root
3150   rimmer
47     uucp
35     lp
30     adm
4      bin
4      daemon
/dev/rdisk/c0t0d0s6 (/usr):
460651 root
206632 bin
791    uucp
46     lp
4      daemon
1      adm
/dev/rdisk/c0t0d0s7 (/export/home):
9      root
```

오래된 파일 또는 비활성 파일 찾기 및 제거

부하가 높은 파일 시스템 정리 작업 과정에서는 최근에 사용되지 않은 파일을 찾아 제거합니다. `ls` 또는 `find` 명령을 사용하여 사용되지 않은 파일을 찾을 수 있습니다. 자세한 내용은 `ls(1)` 및 `find(1)` 매뉴얼 페이지를 참조하십시오.

`/var/tmp` 또는 `/var/spool`에 있는 디렉토리 등 임시 디렉토리를 비우고 **코어** 및 충돌 덤프 파일을 삭제하는 것도 디스크 공간을 절약할 수 있는 방법입니다. 충돌 덤프 파일에 대한 자세한 내용은 17 장, “시스템 충돌 정보 관리(작업)”를 참조하십시오.

▼ 최신 파일 나열 방법

- `ls -t` 명령을 사용하여 가장 최근에 생성되었거나 변경된 파일이 먼저 표시되도록 파일을 나열합니다.

```
$ ls -t [directory]
-t          최신 시간 기록부터 파일을 정렬합니다.
directory  검색할 디렉토리를 식별합니다.
```

예 6-11 최신 파일 나열

다음 예에서는 `ls -tl` 명령을 사용하여 `/var/adm` 디렉토리에서 가장 최근에 생성되었거나 변경된 파일을 찾는 방법을 보여 줍니다. `suolog` 파일이 가장 최근에 생성되었거나 편집되었음을 알 수 있습니다.

```
$ ls -tl /var/adm
total 134
-rw----- 1 root   root      315 Sep 24 14:00 suolog
-r--r--r-- 1 root   other    350700 Sep 22 11:04 lastlog
-rw-r--r-- 1 root   bin      4464 Sep 22 11:04 utmpx
-rw-r--r-- 1 adm    adm      20088 Sep 22 11:04 wtmpx
-rw-r--r-- 1 root   other     0 Sep 19 03:10 messages
-rw-r--r-- 1 root   other     0 Sep 12 03:10 messages.0
-rw-r--r-- 1 root   root     11510 Sep 10 16:13 messages.1
-rw-r--r-- 1 root   root     0 Sep 10 16:12 vold.log
drwxr-xr-x 2 root   sys      512 Sep 10 15:33 sm.bin
drwxrwxr-x 5 adm    adm      512 Sep 10 15:19 acct
drwxrwxr-x 2 adm    sys      512 Sep 10 15:19 sa
-rw----- 1 uucp   bin       0 Sep 10 15:17 aculog
-rw-rw-rw- 1 root   bin       0 Sep 10 15:17 spellhist
drwxr-xr-x 2 adm    adm      512 Sep 10 15:17 log
drwxr-xr-x 2 adm    adm      512 Sep 10 15:17 passwd
```

▼ 오래된 파일 또는 비활성 파일 찾기 및 제거 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 지정된 일수 동안 액세스되지 않은 파일을 찾아 파일에 나열합니다.

```
# find directory -type f[-atime +nnn] [-mtime +nnn] -print > filename &
directory      검색할 디렉토리를 식별합니다. 이 디렉토리에 포함된 디렉토리도
                검색됩니다.
```

`-atime +nnn` 지정한 일 수(*nnn*) 동안 액세스되지 않은 파일을 찾습니다.
`-mtime +nnn` 지정한 일 수(*nnn*) 동안 수정되지 않은 파일을 찾습니다.
filename 비활성 파일 목록이 포함된 파일을 식별합니다.

3 이전 단계에서 나열된 찾은 비활성 파일을 제거합니다.

```
# rm 'cat filename'
```

여기서 *filename*은 이전 단계에서 생성된 파일을 식별합니다. 이 파일에는 비활성 파일 목록이 포함되어 있습니다.

예 6-12 오래된 파일 또는 비활성 파일 찾기 및 제거

다음 예에서는 `/var/adm` 디렉토리 및 하위 디렉토리에서 지난 60일 동안 액세스되지 않은 파일을 보여 줍니다. `/var/tmp/deadfiles` 파일에는 비활성 파일 목록이 포함되어 있습니다. `rm` 명령은 해당 비활성 파일을 제거합니다.

```
# find /var/adm -type f -atime +60 -print > /var/tmp/deadfiles &
# more /var/tmp/deadfiles
/var/adm/aculog
/var/adm/spellhist
/var/adm/wtmpx
/var/adm/sa/sa13
/var/adm/sa/sa27
/var/adm/sa/sa11
/var/adm/sa/sa23
/var/adm/sulog
/var/adm/vold.log
/var/adm/messages.1
/var/adm/messages.2
/var/adm/messages.3
# rm 'cat /var/tmp/deadfiles'
#
```

▼ 임시 디렉토리를 지우는 방법

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 지울 디렉토리로 변경합니다.

```
# cd directory
```



주의 -3단계를 완료하기 전에 올바른 디렉토리에 있는지 확인합니다. 3단계에서는 현재 디렉토리의 모든 파일을 삭제합니다.

- 3 현재 디렉토리의 파일 및 하위 디렉토리를 삭제합니다.

```
# rm -r *
```

- 4 불필요하거나 임시로 사용되거나 오래된 하위 디렉토리 및 파일을 포함하는 다른 디렉토리로 변경합니다.

- 5 3단계를 반복하여 해당 하위 디렉토리 및 파일을 삭제합니다.

예 6-13 임시 디렉토리 지우기

다음 예에서는 mywork 디렉토리를 지우는 방법과 모든 파일 및 하위 디렉토리가 제거되었는지 확인하는 방법을 보여 줍니다.

```
# cd mywork
# ls
filea.000
fileb.000
filec.001
# rm -r *
# ls
#
```

▼ 코어 파일 찾기 및 삭제 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 코어 파일을 검색할 디렉토리로 변경합니다.

- 3 이 디렉토리 및 해당 하위 디렉토리에서 코어 파일을 찾아 제거합니다.

```
# find . -name core -exec rm {} \;
```

예 6-14 코어 파일 찾기 및 삭제

다음 예에서는 find 명령을 사용하여 jones 사용자 계정에서 코어 파일을 찾아 제거하는 방법을 보여 줍니다.

```
# cd /home/jones
# find . -name core -exec rm {} \;
```

▼ 충돌 덤프 파일 삭제 방법

충돌 덤프 파일은 매우 큰 파일일 수 있습니다. 시스템에 해당 파일이 저장되도록 설정한 경우 해당 파일이 필요 이상 오랫동안 보관되지 않도록 하십시오.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 충돌 덤프 파일이 저장된 디렉토리로 변경합니다.

```
# cd /var/crash/system
```

여기서 *system*은 충돌 덤프 파일을 만든 시스템을 식별합니다.



주의 -3단계를 완료하기 전에 올바른 디렉토리에 있는지 확인합니다. 3단계에서는 현재 디렉토리의 모든 파일을 삭제합니다.

3 충돌 덤프 파일을 제거합니다.

```
# rm *
```

4 충돌 덤프 파일이 제거되었는지 확인합니다.

```
# ls
```

예 6-15 충돌 덤프 파일 삭제

다음 예에서는 *venus* 시스템에서 충돌 덤프 파일을 제거하는 방법과 충돌 덤프 파일이 제거되었는지 확인하는 방법을 보여 줍니다.

```
# cd /var/crash/venus
# rm *
# ls
```

UFS 쿼터 관리(작업)

이 장에서는 디스크 공간 및 inode에 대한 UFS 쿼터를 설정하고 관리하는 방법에 대해 설명합니다.

UFS 쿼터 관리와 관련된 자세한 내용은 다음을 참조하십시오.

- 92 페이지 “UFS 쿼터 설정(작업 맵)”
- 96 페이지 “UFS 쿼터 유지 관리(작업 맵)”

Oracle Solaris ZFS 쿼터 관리에 대한 자세한 내용은 **Oracle Solaris ZFS 관리 설명서**의 “ZFS 쿼터 및 예약 설정”를 참조하십시오.

UFS 쿼터란?

UFS 쿼터를 사용하여 시스템 관리자는 파일 시스템의 크기를 제어할 수 있습니다. 쿼터는 개별 사용자가 사용할 수 있는 파일 수와 거의 동일한 inode 수 및 디스크 공간을 제한합니다. 따라서 쿼터는 사용자 홈 디렉토리가 상주하는 파일 시스템에 특히 유용합니다. 일반적으로 `public` 및 `/tmp` 파일 시스템에서는 쿼터를 설정해도 큰 도움이 되지 않습니다.

UFS 쿼터 사용

쿼터를 설정한 후에는 사용자가 사용할 수 있는 inode 수 또는 디스크 공간에 맞게 쿼터를 변경할 수 있습니다. 시스템에 맞게 쿼터를 추가하거나 제거할 수도 있습니다. 쿼터 또는 쿼터가 초과될 수 있는 시간을 변경하는 방법, 개별 쿼터를 사용 안함으로 설정하는 방법 또는 파일 시스템에서 쿼터를 제거하는 방법은 **99 페이지 “UFS 쿼터 변경 및 제거”**를 참조하십시오.

또한 쿼터 상태를 모니터링할 수 있습니다. UFS 쿼터 명령을 사용하여 관리자는 파일 시스템에 대한 쿼터 정보를 표시하거나 쿼터를 초과한 사용자를 검색할 수 있습니다. 해당 명령 사용 방법을 설명하는 절차는 **97 페이지 “UFS 쿼터 확인”**을 참조하십시오.

UFS 쿼터에 대한 소프트웨어 제한 및 하드 제한 설정

소프트 제한과 하드 제한을 모두 설정할 수 있습니다. 시스템에서는 사용자가 하드 제한을 초과할 수 없도록 하지만, 시스템 관리자는 사용자가 일시적으로 초과할 수 있는 소프트웨어 제한을 설정할 수 있습니다. 소프트웨어 제한은 하드 제한보다 작아야 합니다.

사용자가 소프트웨어 제한을 초과하면 쿼터 타이머가 시작됩니다. 쿼터 타이머가 작동하는 동안 사용자는 소프트웨어 제한을 초과할 수 있지만 하드 제한은 초과할 수 없습니다. 사용량이 소프트웨어 제한 아래로 떨어지면 타이머가 재설정됩니다. 하지만 타이머가 만료될 때까지 사용자의 사용량이 소프트웨어 제한을 계속 초과할 경우 소프트웨어 제한이 하드 제한으로 적용됩니다. 기본적으로 소프트웨어 제한 타이머는 7일로 설정되어 있습니다.

repquota 및 quota 명령의 timeleft 필드는 타이머 값을 보여 줍니다.

예를 들어, 사용자의 블록 수 소프트웨어 제한이 10,000이고 블록 수 하드 제한이 12,000일 경우, 사용자의 블록 사용량이 10,000개를 초과하고 7일 타이머도 초과되면 사용자는 블록 사용량이 소프트웨어 제한 아래로 떨어질 때까지 해당 파일 시스템에서 디스크 블록을 더 이상 할당할 수 없습니다.

디스크 블록과 파일 제한의 차이

파일 시스템은 사용자에게 두 가지 리소스(데이터에 대한 블록과 파일에 대한 inode)를 제공합니다. 각 파일은 하나의 inode를 사용하며, 파일 데이터는 데이터 블록에 저장됩니다. 일반적으로 데이터 블록은 1KB 블록으로 구성됩니다.

디렉토리가 없을 경우 사용자는 블록을 사용하지 않고도 빈 파일을 만들어 inode 쿼터를 초과할 수 있습니다. 또한 하나의 inode를 사용할 수 있으며, 사용자 쿼터의 모든 데이터 블록을 사용할 만큼 큰 하나의 파일을 만들어 블록 쿼터를 초과할 수 있습니다.

UFS 쿼터 설정

쿼터 설정은 다음과 같은 일반적인 단계로 구성됩니다.

1. /etc/vfstab 파일 항목에 쿼터 옵션을 추가하여 시스템이 재부트될 때마다 쿼터가 적용되도록 합니다. 또한 파일 시스템의 최상위 디렉토리에 quotas 파일을 만듭니다.
2. 일회용 쿼터를 만든 후 쿼터를 프로토타입으로 복사하여 다른 사용자 쿼터를 설정합니다.
3. 쿼터를 켜기 전에 제안된 쿼터와 현재 디스크 사용량 간의 일관성을 확인하여 충돌이 없도록 합니다.
4. 하나 이상의 파일 시스템에 대해 쿼터를 켭니다.

해당 절차에 대한 구체적인 내용은 92 페이지 “UFS 쿼터 설정(작업 맵)”을 참조하십시오.

다음 표에서는 디스크 쿼터 설정에 사용할 명령에 대해 설명합니다.

표 7-1 UFS 쿼터 설정 명령

명령	작업	매뉴얼 페이지
edquota	사용자별로 inode 수와 디스크 공간에 대한 하드 제한 및 소프트 제한을 설정합니다.	edquota(1M)
quotacheck	마운트된 각 UFS 파일 시스템을 검사하여 파일 시스템의 현재 디스크 사용량과 파일 시스템의 디스크 쿼터 파일에 저장된 정보를 비교합니다. 그런 다음 불일치를 해결합니다.	quotacheck(1M)
quotaon	지정된 파일 시스템에 대한 쿼터를 활성화합니다.	quotaon(1M)
quota	마운트된 파일 시스템에 대한 사용자의 UFS 디스크 쿼터를 표시하여 쿼터가 제대로 설정되었는지 확인합니다.	quota(1M)

UFS 쿼터 설정 지침

UFS 쿼터를 설정하기 전에 각 사용자에게 할당할 디스크 공간과 inode 수를 확인해야 합니다. 전체 파일 시스템 공간이 초과되지 않도록 하려는 경우 전체 파일 시스템 크기를 사용자 수로 나누면 됩니다. 예를 들어, 세 명의 사용자가 100MB 슬라이스를 공유하며 동일한 디스크 공간을 필요로 하는 경우 각 사용자에게 33MB를 할당할 수 있습니다.

일부 사용자에게 제한을 적용하지 않으려는 환경에서 전체 파일 시스템 크기 이상으로 추가되도록 개별 쿼터를 설정할 수 있습니다. 예를 들어, 세 명의 사용자가 100MB 슬라이스를 공유하는 경우 각 사용자에게 40MB를 할당할 수 있습니다.

edquota 명령을 사용하여 한 사용자에게 대한 쿼터를 설정한 경우 이 쿼터를 프로토타입으로 사용하여 동일한 파일 시스템의 다른 사용자에게 대해 동일한 쿼터를 설정할 수 있습니다.

쿼터를 켜기 전에 다음 작업을 수행하십시오.

- 먼저 UFS 파일 시스템에서 쿼터를 구성합니다.
- 각 사용자에게 대한 쿼터를 설정하고 quotacheck 명령을 실행하여 현재 디스크 사용량과 쿼터 파일 간의 일관성을 확인합니다.
- 시스템이 자주 재부트되지 않는 경우 quotacheck 명령을 주기적으로 실행합니다.

edquota 명령을 사용하여 설정한 쿼터는 quotaon 명령을 사용하여 쿼터를 켜기 전까지 적용되지 않습니다. 쿼터 파일을 제대로 구성한 경우 시스템이 재부트되고 파일 시스템이 마운트될 때마다 쿼터가 자동으로 켜집니다.

UFS 쿼터 설정(작업 맵)

작업	설명	수행 방법
1. 파일 시스템에서 쿼터를 구성합니다.	파일 시스템이 마운트될 때마다 쿼터가 활성화되도록 /etc/vfstab 파일을 편집합니다. 또한 quotas 파일을 만듭니다.	92 페이지 “파일 시스템에서 UFS 쿼터 구성 방법”
2. 사용자에게 대한 UFS 쿼터를 설정합니다.	edquota 명령을 사용하여 단일 사용자 계정에 대한 디스크 쿼터 및 inode 쿼터를 만듭니다.	93 페이지 “사용자에게 대한 UFS 쿼터 설정 방법”
3. (선택 사항) 다중 사용자에게 대한 UFS 쿼터를 설정합니다.	edquota 명령을 사용하여 다른 사용자 계정에 프로토타입 쿼터를 적용합니다.	94 페이지 “다중 사용자에게 대한 UFS 쿼터 설정 방법”
4. 일관성을 확인합니다.	quotacheck 명령을 사용하여 하나 이상의 파일 시스템에서 쿼터와 현재 디스크 사용량의 일관성을 비교합니다.	94 페이지 “UFS 쿼터 일관성 확인 방법”
5. UFS 쿼터를 켭니다.	quotaon 명령을 사용하여 하나 이상의 파일 시스템에서 UFS 쿼터를 시작합니다.	95 페이지 “UFS 쿼터를 켜는 방법”

▼ 파일 시스템에서 UFS 쿼터 구성 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 /etc/vfstab 파일을 편집하고 쿼터가 설정될 각 UFS 파일 시스템에 대해 `rq`를 `mount options` 필드에 추가합니다.
- 3 쿼터가 설정될 파일 시스템의 루트로 디렉토리를 변경합니다.
- 4 `quotas`라는 파일을 만듭니다.
`# touch quotas`
- 5 권한을 슈퍼유저 액세스에 대해서만 읽기/쓰기로 변경합니다.
`# chmod 600 quotas`

예 7-1 파일 시스템에서 UFS 쿼터 구성

다음 `/etc/vfstab` 예에서는 `pluto` 시스템의 `/export/home` 디렉토리가 로컬 시스템의 NFS 파일 시스템으로 마운트되었음을 보여 줍니다. 쿼터가 `mount options` 열의 `rq` 항목에 따라 사용으로 설정되었음을 알 수 있습니다.

```
# device device mount FS fsck mount mount
# to mount to fsck point type pass at boot options
# pluto:/export/home - /export/home nfs - yes rq
```

`/etc/vfstab` 파일의 다음 라인 예에서는 `mount options` 열의 `rq` 항목에 따라 쿼터가 사용으로 설정된 상태로 로컬 `/work` 디렉토리가 마운트되었음을 알 수 있습니다.

```
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
#/dev/dsk/c0t4d0s0 /dev/rdisk/c0t4d0s0 /work ufs 3 yes rq
```

- 참조
- 93 페이지 “사용자에 대한 UFS 쿼터 설정 방법”
 - 94 페이지 “다중 사용자에게 대한 UFS 쿼터 설정 방법”
 - 94 페이지 “UFS 쿼터 일관성 확인 방법”
 - 95 페이지 “UFS 쿼터를 켜는 방법”

▼ 사용자에게 대한 UFS 쿼터 설정 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “[Configuring RBAC \(Task Map\)](#)”를 참조하십시오.

- 2 쿼터 편집기를 사용하여 파일 시스템의 루트 디렉토리에 `quotas` 파일이 있는 마운트된 각 UFS 파일 시스템에 대해 한 라인의 쿼터 정보를 포함하는 임시 파일을 만듭니다.

```
# edquota username
```

여기서 `username`은 쿼터를 설정할 사용자입니다.

- 3 1KB 디스크 블록 수(소프트 및 하드 모두)를 각 파일 시스템에 대해 지정한 쿼터로 변경합니다.

- 4 inode 수(소프트 및 하드 모두)를 기본값인 0에서 각 파일 시스템에 대해 지정한 쿼터로 변경합니다.

- 5 사용자의 UFS 쿼터를 확인합니다.

```
# quota -v username
```

- v 쿼터가 있는 마운트된 모든 파일 시스템에 대한 사용자의 쿼터 정보를 표시합니다.
- username* 쿼터 제한을 확인할 사용자 이름을 지정합니다.

예 7-2 사용자에 대한 UFS 쿼터 설정

다음 예에서는 루트 디렉토리에 `quotas` 파일을 포함하는 유일한 마운트된 파일 시스템이 `/files`인 시스템에서 `edquota`로 연 임시 파일의 콘텐츠를 보여 줍니다.

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

다음 예에서는 쿼터 설정 후 임시 파일의 동일한 라인을 보여 줍니다.

```
fs /files blocks (soft = 50, hard = 60) inodes (soft = 90, hard = 100)
```

▼ 다중 사용자에 대한 UFS 쿼터 설정 방법

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 쿼터 편집기를 사용하여 프로토타입 사용자에 대해 이미 설정한 쿼터를 지정한 추가 사용자에게 적용합니다.

```
# edquota -p prototype-user username ...
```

prototype-user 쿼터를 설정한 계정의 사용자 이름입니다.

username... 추가 계정에 대한 하나 이상의 사용자 이름을 지정합니다. 두 개 이상의 사용자 이름을 지정할 때는 각 사용자 이름을 공백으로 구분합니다.

예 7-3 다중 사용자에 대한 프로토타입 UFS 쿼터 설정

다음 예에서는 사용자 bob에 대해 설정된 쿼터를 사용자 mary 및 john에게 적용하는 방법을 보여 줍니다.

```
# edquota -p bob mary john
```

▼ UFS 쿼터 일관성 확인 방법

시스템이 재부트될 때 `quotacheck` 명령이 자동으로 실행됩니다. 일반적으로 쿼터가 설정된 빈 파일 시스템에서는 `quotacheck` 명령을 실행할 필요가 없습니다. 하지만 기존

파일이 있는 파일 시스템에서 쿼터를 설정하는 경우 `quotacheck` 명령을 실행하여 쿼터 데이터베이스와 파일 시스템의 기존 파일 또는 `inode`를 동기화해야 합니다.

큰 파일 시스템에서 `quotacheck` 명령을 실행할 때는 시간이 오래 걸릴 수도 있습니다.

주 - 정확한 디스크 데이터를 얻으려면 수동으로 `quotacheck` 명령을 실행할 때 확인하려는 파일 시스템이 정지 상태여야 합니다.

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 UFS 파일 시스템에서 일관성 검사를 실행합니다.

```
# quotacheck [-va] filesystem
```

`-v` (선택 사항) 특정 파일 시스템에서 각 사용자에게 대한 디스크 쿼터를 식별합니다.

`-a` `/etc/vfstab` 파일에 `rq` 항목을 포함하는 모든 파일 시스템을 확인합니다.

`filesystem` 확인할 파일 시스템을 지정합니다.

자세한 내용은 `quotacheck(1M)` 매뉴얼 페이지를 참조하십시오.

예 7-4 UFS 쿼터 일관성 확인

다음 예에서는 `/dev/rdisk/c0t0d0s7` 슬라이스에서 `/export/home` 파일 시스템에 대한 쿼터를 확인하는 방법을 보여 줍니다. `/export/home` 파일 시스템이 `/etc/vfstab` 파일에 `rq` 항목을 포함하는 유일한 파일 시스템임을 알 수 있습니다.

```
# quotacheck -va
*** Checking quotas for /dev/rdisk/c0t0d0s7 (/export/home)
```

▼ UFS 쿼터를 켜는 방법

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 파일 시스템 쿼터를 켭니다.

```
# quotaon [-v] -a filesystem ...
```

- v 쿼터를 켜 후 각 파일 시스템에 대한 메시지를 표시합니다.
- a /etc/vfstab 파일에 rq 항목을 포함하는 모든 파일 시스템에 대해 쿼터를 켭니다.
- filesystem... 지정한 하나 이상의 파일 시스템에 대해 쿼터를 켭니다. 두 개 이상의 파일 시스템을 지정할 때는 각 파일 시스템 이름을 공백으로 구분합니다.

예 7-5 UFS 쿼터 켜기

다음 예에서는 /dev/dsk/c0t4d0s7 및 /dev/dsk/c0t3d0s7 슬라이스에서 파일 시스템에 대한 쿼터를 켜는 방법을 보여 줍니다.

```
# quotaon -v /dev/dsk/c0t4d0s7 /dev/dsk/c0t3d0s7
/dev/dsk/c0t4d0s7: quotas turned on
/dev/dsk/c0t3d0s7: quotas turned on
```

UFS 쿼터 유지 관리(작업 맵)

작업	설명	수행 방법
UFS 쿼터 초과를 확인합니다.	quota 명령을 사용하여 UFS 쿼터가 활성화된 파일 시스템에 대한 사용자별 UFS 쿼터 및 디스크 사용량을 표시합니다.	97 페이지 “UFS 쿼터 초과 확인 방법”
파일 시스템에서 UFS 쿼터를 확인합니다.	repquota 명령을 사용하여 하나 이상의 파일 시스템에 대한 모든 사용자의 UFS 쿼터 및 디스크 사용량을 표시합니다.	98 페이지 “파일 시스템에 대한 UFS 쿼터 확인 방법”
소프트 제한 기본값을 변경합니다.	edquota 명령을 사용하여 사용자가 디스크 공간 쿼터 또는 inode 쿼터를 초과할 수 있는 기간을 변경합니다.	99 페이지 “소프트 제한 기본값 변경 방법”
사용자에 대한 UFS 쿼터를 변경합니다.	쿼터 편집기 edquota를 사용하여 개별 사용자에 대한 쿼터를 변경합니다.	100 페이지 “사용자에 대한 UFS 쿼터 변경 방법”
사용자에 대한 UFS 쿼터를 사용 안함으로 설정합니다.	쿼터 편집기 edquota를 사용하여 개별 사용자에 대한 쿼터를 사용 안함으로 설정합니다.	102 페이지 “사용자에 대한 UFS 쿼터를 사용 안함으로 설정하는 방법”
UFS 쿼터를 끕니다.	quotaoff 명령을 사용하여 UFS 쿼터를 끕니다.	103 페이지 “UFS 쿼터를 끄는 방법”

UFS 쿼터 확인

UFS 디스크 쿼터 및 inode 쿼터를 설정하고 컨 후에는 해당 쿼터를 초과한 사용자를 확인할 수 있습니다. 또한 전체 파일 시스템에 대한 쿼터 정보를 확인할 수 있습니다.

다음 표에서는 쿼터 확인에 사용할 명령에 대해 설명합니다.

표 7-2 UFS 쿼터 확인 명령

명령	작업
<code>quota(1M)</code>	사용자 쿼터, 현재 디스크 사용량 및 해당 쿼터를 초과하는 사용자에 대한 정보를 표시합니다.
<code>repquota(1M)</code>	쿼터, 파일 및 지정된 파일 시스템에 대해 소유된 공간을 표시합니다.

▼ UFS 쿼터 초과 확인 방법

`quota` 명령을 사용하여 쿼터가 활성화된 파일 시스템에 대한 사용자별 UFS 쿼터 및 디스크 사용량을 표시할 수 있습니다.

1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 쿼터가 사용으로 설정된 마운트된 파일 시스템에 대한 사용자 쿼터를 표시합니다.

```
# quota [-v] username
```

`-v` 쿼터가 설정된 마운트된 모든 파일 시스템에 대한 하나 이상의 사용자 쿼터를 표시합니다.

`username` 사용자 계정의 로그인 이름 또는 UID입니다.

예 7-6 UFS 쿼터 초과 확인

다음 예에서는 UID 301로 식별된 사용자 계정에 하나의 1KB 쿼터가 있지만 디스크 공간이 사용되지 않았음을 알 수 있습니다.

```
# quota -v 301
Disk quotas for bob (uid 301):
Filesystem usage quota limit timeleft files quota limit timeleft
/export/home 0 1 2 0 2 3
```

Filesystem 파일 시스템에 대한 마운트 지점입니다.

usage 현재 블록 사용량입니다.

quota	소프트 블록 제한입니다.
limit	하드 블록 제한입니다.
timeleft	쿼터 타이머에 남은 시간(일)입니다.
files	현재 inode 사용량입니다.
quota	소프트 inode 제한입니다.
limit	하드 inode 제한입니다.
timeleft	쿼터 타이머에 남은 시간(일)입니다.

▼ 파일 시스템에 대한 UFS 쿼터 확인 방법

repquota 명령을 사용하여 하나 이상의 파일 시스템에 대한 모든 사용자의 UFS 쿼터 및 디스크 사용량을 표시합니다.

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 사용량이 없는 경우에도 하나 이상의 파일 시스템에 대한 모든 UFS 쿼터를 표시합니다.

```
# repquota [-v] -a filesystem
```

-v 사용자가 리소스를 소비하지 않는 경우에도 모든 사용자에 대한 UFS 쿼터를 보고합니다.

-a 모든 파일 시스템을 보고합니다.

filesystem 지정된 파일 시스템을 보고합니다.

예 7-7 파일 시스템에 대한 UFS 쿼터 확인

다음 예에서는 유일한 파일 시스템(/export/home)에서 쿼터가 사용으로 설정된 시스템의 repquota 명령 출력을 보여 줍니다.

```
# repquota -va
/dev/dsk/c0t3d0s7 (/export/home):
      Block limits                File limits
User   used  soft  hard  timeleft  used  soft  hard  timeleft
#301  --    0    1    2.0 days    0    2    3
#341  --   57   60   7.0 days  2    90  100

Block limits    정의
used            현재 블록 사용량입니다.
```

soft	소프트 블록 제한입니다.
hard	하드 블록 제한입니다.
timeleft	쿼터 타이머에 남은 시간(일)입니다.
File limits	파일 제한 정의입니다.
used	현재 inode 사용량입니다.
soft	소프트 inode 제한입니다.
hard	하드 inode 제한입니다.
timeleft	쿼터 타이머에 남은 시간(일)입니다.

UFS 쿼터 변경 및 제거

사용자가 사용할 수 있는 디스크 공간 또는 inode 수에 맞게 쿼터를 변경할 수 있습니다. 필요에 따라 개별 사용자에게 쿼터 또는 전체 파일 시스템의 쿼터를 제거할 수도 있습니다.

다음 표에서는 쿼터 변경 또는 제거에 사용할 명령에 대해 설명합니다.

표 7-3 UFS 쿼터 변경 및 제거 명령

명령	매뉴얼 페이지	설명
edquota	edquota(1M)	사용자별 inode 수 또는 디스크 공간에 대한 하드 제한 및 소프트 제한을 변경합니다. 또한 쿼터가 설정된 각 파일 시스템에 대한 소프트 제한을 변경합니다.
quotaoff	quotaon(1M)	지정된 파일 시스템에 대한 쿼터를 끕니다.

▼ 소프트 제한 기본값 변경 방법

기본적으로 사용자는 1주일 동안 UFS 쿼터에 대한 소프트 시간 제한을 초과할 수 있습니다. 따라서 1주일 동안 디스크 공간 쿼터 또는 inode 쿼터의 소프트 시간 제한을 반복적으로 위반한 후에는 사용자가 더 이상 inode 또는 디스크 블록을 사용하지 못합니다.

edquota 명령을 사용하여 사용자가 디스크 공간 쿼터 또는 inode 쿼터를 초과할 수 있는 기간을 변경할 수 있습니다.

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 쿼터 편집기를 사용하여 소프트 시간 제한을 포함하는 임시 파일을 만듭니다.
`# edquota -t`
여기서 `-t` 옵션은 각 파일 시스템에 대한 소프트 시간 제한 편집을 지정합니다.
- 3 시간 제한을 기본값인 0에서 지정한 시간 제한으로 변경합니다. 숫자 및 `month, week, day, hour, min` 또는 `sec` 키워드를 사용합니다.

주 - 이 절차는 현재 쿼터 위반자에게 영향을 끼치지 않습니다.

예 7-8 소프트 제한 기본값 변경

다음 예에서는 쿼터가 설정된 유일한 마운트된 파일 시스템이 `/export/home`인 시스템에서 `edquota` 명령으로 연 임시 파일의 콘텐츠를 보여 줍니다. 기본값인 0을 적용하면 기본 시간 제한인 1주일이 사용됩니다.

```
fs /export/home blocks time limit = 0 (default), files time limit = 0 (default)
```

다음 예에서는 블록 쿼터 초과 시간 제한이 2주로 변경된 후의 동일한 임시 파일을 보여 줍니다. 파일 수 초과 시간 제한도 16일로 변경되었음을 알 수 있습니다.

```
fs /export/home blocks time limit = 2 weeks, files time limit = 16 days
```

▼ 사용자에 대한 UFS 쿼터 변경 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 쿼터 편집기를 사용하여 파일 시스템의 루트 디렉토리에 `quotas` 파일이 있는 마운트된 각 파일 시스템에 대해 한 라인을 포함하는 임시 파일을 엽니다.
`# edquota username`
여기서 `username`은 쿼터를 변경할 사용자 이름을 지정합니다.



주의 - `edquota` 명령에 대한 인수로 다중 사용자를 지정할 수 있습니다. 단, 이 정보에 해당하는 사용자는 표시되지 않습니다. 혼동되지 않도록 하려면 사용자 이름을 하나만 지정하십시오.

- 3 1KB 디스크 블록 수(소프트 및 하드 모두)와 inode 수(소프트 및 하드 모두)를 지정합니다.
- 4 사용자의 UFS 쿼터가 제대로 변경되었는지 확인합니다.

```
# quota -v username
```

```
-v          쿼터가 사용으로 설정된 마운트된 모든 파일 시스템에 대한 사용자 UFS
           쿼터 정보를 표시합니다.
```

```
username   쿼터를 확인할 사용자 이름을 지정합니다.
```

예 7-9 사용자에 대한 UFS 쿼터 변경

다음 예에서는 `edquota` 명령으로 연 임시 파일의 콘텐츠를 보여 줍니다. 이 임시 파일은 파일 시스템의 루트 디렉토리에 `quotas` 파일을 포함하는 유일한 마운트된 파일 시스템이 `/files`인 시스템에서 열린 것입니다.

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

다음 출력에서는 쿼터 변경 후의 동일한 임시 파일을 보여 줍니다.

```
fs /files blocks (soft = 0, hard = 500) inodes (soft = 0, hard = 100)
```

예 7-10 하드 UFS 쿼터가 변경되었는지 확인

다음 예에서는 사용자 `smith`에 대한 하드 쿼터가 500개의 1KB 블록 및 100개의 inode로 변경되었음을 알 수 있습니다.

```
# quota -v smith
Disk quotas for smith (uid 12):
Filesystem  usage  quota  limit  timeleft  files  quota  limit  timeleft
  /files    1      0     500           1      0     100
```

▼ 사용자에 대한 UFS 쿼터를 사용 안함으로 설정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 쿼터 편집기를 사용하여 최상위 디렉토리에 `quotas` 파일이 있는 마운트된 각 파일 시스템에 대해 한 라인을 포함하는 임시 파일을 만듭니다.

```
# edquota username
```

여기서 `username`은 쿼터를 사용 안함으로 설정할 사용자 이름을 지정합니다.



주의 - `edquota` 명령에 대한 인수로 다중 사용자를 지정할 수 있습니다. 단, 이 정보에 해당하는 사용자는 표시되지 않습니다. 혼동되지 않도록 하려면 사용자 이름을 하나만 지정하십시오.

- 3 1KB 디스크 블록 수(소프트 및 하드 모두)를 0으로 변경합니다.

- 4 inode 수(소프트 및 하드 모두)를 0으로 변경합니다.

주 - 값을 0으로 변경했는지 확인합니다. 텍스트 파일에서 해당 라인을 삭제하지 마십시오.

- 5 사용자의 UFS 쿼터를 사용 안함으로 설정했는지 확인합니다.

```
# quota -v username
```

`-v` 쿼터가 사용으로 설정된 마운트된 모든 파일 시스템에 대한 사용자 UFS 쿼터 정보를 표시합니다.

`username` UFS 쿼터를 확인할 사용자 이름(UID)을 지정합니다.

예 7-11 사용자에 대한 UFS 쿼터를 사용 안함으로 설정

다음 예에서는 파일 시스템의 루트 디렉토리에 쿼터 파일을 포함하는 유일한 마운트된 파일 시스템이 `/files`인 시스템에서 `edquota` 명령으로 연 임시 파일의 콘텐츠를 보여 줍니다.

```
fs /files blocks (soft = 50, hard = 60) inodes (soft = 90, hard = 100)
```

다음 예에서는 UFS 쿼터가 사용 안함으로 설정된 후의 동일한 임시 파일을 보여 줍니다.

```
fs /files blocks (soft = 0, hard = 0) inodes (soft = 0, hard = 0)
```

▼ UFS 쿼터를 끄는 방법

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 파일 시스템 쿼터를 끕니다.

```
# quotaoff [-v] -a filesystem ...
```

-v UFS 쿼터를 끌 때 나타나는 각 파일 시스템의 메시지를 표시합니다.

-a 모든 파일 시스템에 대한 UFS 쿼터를 끕니다.

filesystem 지정한 하나 이상의 파일 시스템에 대한 UFS 쿼터를 끕니다. 두 개 이상의 파일 시스템을 지정할 때는 각 파일 시스템 이름을 공백으로 구분합니다.

예 7-12 쿼터 끄기

다음 예에서는 /export/home 파일 시스템에 대한 쿼터를 끄는 방법을 보여 줍니다.

```
# quotaoff -v /export/home
/export/home: quotas turned off
```


시스템 작업 예약(작업)

이 장에서는 `crontab` 및 `at` 명령을 사용하여 루틴 또는 단일(일회성) 시스템 작업을 예약하는 방법을 설명합니다.

또한 다음 파일을 사용하여 이러한 명령에 대한 액세스를 제어하는 방법을 설명합니다.

- `cron.deny`
- `cron-allow`
- `at.deny`

시스템 작업 일정 잡기와 연관된 절차에 대한 내용은 다음을 참조하십시오.

- 105 페이지 “`crontab` 파일 만들기 및 편집(작업 맵)”
- 117 페이지 “`at` 명령 사용(작업 맵)”

crontab 파일 만들기 및 편집(작업 맵)

작업	설명	수행 방법
crontab 파일 만들기 또는 편집	<code>crontab -e</code> 명령을 사용하여 <code>crontab</code> 파일을 만들거나 편집합니다.	111 페이지 “ <code>crontab</code> 파일을 만들거나 편집하는 방법”
crontab 파일이 존재하는지 확인	<code>ls -l</code> 명령을 사용하여 <code>/var/spool/cron/crontabs</code> 파일의 내용을 확인합니다.	112 페이지 “ <code>crontab</code> 파일이 존재하는지 확인하는 방법”
crontab 파일 표시	<code>ls -l</code> 명령을 사용하여 <code>crontab</code> 파일을 표시합니다.	112 페이지 “ <code>crontab</code> 파일을 표시하는 방법”
crontab 파일 제거	<code>crontab</code> 파일은 제한적 권한으로 설정됩니다. <code>rm</code> 명령이 아닌 <code>crontab -r</code> 명령을 사용하여 <code>crontab</code> 파일을 제거합니다.	113 페이지 “ <code>crontab</code> 파일을 제거하는 방법”

작업	설명	수행 방법
crontab 액세스 거부	crontab 명령에 대한 사용자 액세스를 거부하려면 /etc/cron.d/cron.deny 파일을 편집하여 이 파일에 사용자 이름을 추가합니다.	115 페이지 “crontab 명령 액세스를 거부하는 방법”
crontab 액세스를 지정된 사용자로 제한	crontab 명령에 대한 사용자 액세스를 허용하려면 /etc/cron.d/cron.allow 파일에 사용자 이름을 추가합니다.	116 페이지 “crontab 명령 액세스를 지정된 사용자로 제한하는 방법”

시스템 작업을 자동으로 실행하는 방법

많은 시스템 작업을 자동으로 실행하도록 설정할 수 있습니다. 이러한 작업 중 일부는 정기적 간격으로 발생해야 합니다. 다른 작업은 한 번만, 보통 저녁 또는 주말에 근무의 시간 중 실행해야 합니다.

이 단원에서는 crontab 및 at의 두 명령에 대한 개요 정보를 다룹니다. 이러한 명령으로 루틴 작업이 자동으로 실행되도록 예약할 수 있습니다. crontab 명령은 반복적 명령을 예약합니다. at 명령은 한 번 실행하는 작업을 예약합니다.

다음 표는 crontab 및 at 명령과 이러한 명령에 대한 액세스를 제어하는 파일을 요약합니다.

표 8-1 명령 요약: 시스템 작업 예약

명령	예약대상	파일 위치	액세스 제어 파일
crontab	정기적 간격으로 다중 시스템 작업	/var/spool/cron/crontabs	/etc/cron.d/cron.allow 및 /etc/cron.d/cron.deny
at	단일 시스템 작업	/var/spool/cron/atjobs	/etc/cron.d/at.deny

또한 Solaris Management Console의 일정이 잡힌 작업 도구를 사용하여 루틴 작업 일정을 잡을 수 있습니다. Solaris Management Console 사용 및 시작에 대한 자세한 내용은 [시스템 관리 설명서: 기본 관리의 2 장, “Solaris Management Console 작업\(작업\)”](#)을 참조하십시오.

반복적 작업 예약: crontab

crontab 명령을 사용하여 루틴 시스템 관리 작업이 매일, 매주 또는 매월 실행되도록 예약할 수 있습니다.

일별 `crontab` 시스템 관리 작업은 다음을 포함할 수 있습니다.

- 임시 디렉토리에서 며칠 이상 지난 파일 제거
- 계산 요약 명령 실행
- `df` 및 `ps` 명령을 사용하여 시스템 스냅샷 생성
- 일별 보안 모니터링 수행
- 시스템 백업 실행

주별 `crontab` 시스템 관리 작업은 다음을 포함할 수 있습니다.

- `man -k` 명령에서 사용할 `catman` 데이터베이스 재구성
- `fsck -n` 명령을 실행하여 디스크 문제 나열

월별 `crontab` 시스템 관리 작업은 다음을 포함할 수 있습니다.

- 특정 월 동안 사용되지 않은 파일 나열
- 월별 계산 보고서 생성

추가로 사용자는 미리 알림 보내기 및 백업 파일 제거와 같은 다른 루틴 시스템 작업을 실행하도록 `crontab` 명령을 예약할 수 있습니다.

`crontab` 작업 예약에 대한 단계별 지침은 [111 페이지 “crontab 파일을 만들거나 편집하는 방법”](#)을 참조하십시오.

단일 작업 예약:at

`at` 명령으로 나중에 실행할 작업을 예약할 수 있습니다. 작업은 단일 명령 또는 스크립트로 구성할 수 있습니다.

`crontab`과 마찬가지로, `at` 명령으로 루틴 작업의 자동 실행을 예약할 수 있습니다. 그러나 `crontab` 파일과 달리, `at` 파일은 작업을 한 번 실행한 후에 디렉토리에서 제거됩니다. 따라서 `at` 명령은 나중에 조사를 위해 별도의 파일로 출력을 재지정하는 단순 명령 또는 스크립트를 실행하는 데 가장 유용합니다.

`at` 작업을 제출하려면 명령을 입력하고 `at` 명령 구문에 따라 작업을 실행할 시간을 예약하는 옵션을 지정하면 됩니다. `at` 작업 제출에 대한 자세한 내용은 [118 페이지 “at 명령의 설명”](#)을 참조하십시오.

`at` 명령은 실행된 명령 또는 스크립트를 현재 환경 변수의 복사본과 함께 `/var/spool/cron/atjobs` 디렉토리에 저장합니다. `at` 작업 파일 이름은 `at` 대기열의 위치를 지정하는 긴 숫자 뒤에 `.a` 확장자를 붙여서 만듭니다(예: `793962000.a`).

`cron` 데몬은 시작 시 `at` 작업을 검사하고 제출된 새 작업을 수신합니다. `cron` 데몬이 `at` 작업을 실행한 후에 `at` 작업의 파일이 `atjobs` 디렉토리에서 제거됩니다. 자세한 내용은 [at\(1\)](#) 매뉴얼 페이지를 참조하십시오.

`at` 작업 예약에 대한 단계별 지침은 [119 페이지 “at 작업을 만드는 방법”](#)을 참조하십시오.

반복적 시스템 작업 예약(cron)

다음 단원에서는 crontab 파일을 만들고, 편집, 표시, 제거하는 방법과 이들의 액세스를 제어하는 방법을 설명합니다.

crontab 파일 내부

cron 데몬은 각 crontab 파일에서 발견된 명령에 따라 시스템 작업을 예약합니다. crontab 파일은 정기적 간격으로 실행될 명령들(한 라인에 명령 하나씩)로 구성됩니다. 각 라인의 시작 부분은 명령을 실행할 때 cron 데몬을 알려주는 날짜 및 시간 정보를 포함합니다.

예를 들어, root라는 crontab 파일이 SunOS 소프트웨어 설치 중 제공됩니다. 파일 내용에 다음 명령줄이 포함됩니다.

```
10 3 * * * /usr/sbin/logadm      (1)
15 3 * * 0 /usr/lib/fs/nfs/nfsfind  (2)
1 2 * * * [ -x /usr/sbin/rtc ] && /usr/sbin/rtc -c > /dev/null 2>&1  (3)
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean  (4)
```

다음은 이러한 명령줄의 출력을 설명합니다.

- 첫번째 라인은 매일 오전 3:10에 logadm 명령을 실행합니다.
- 두번째 라인은 매주 일요일 오전 3:15에 nfsfind 스크립트를 실행합니다.
- 세번째 라인은 매일 오전 2:10에 일광 절약 시간을 검사하는(필요한 경우 수정하는) 스크립트를 실행합니다.

RTC 시간대나 /etc/rtc_config 파일이 없는 경우 이 항목은 아무것도 아닙니다.

x86 전용 - /usr/sbin/rtc 스크립트는 x86 기반 시스템에서만 실행할 수 있습니다.

- 네번째 라인은 매일 오전 3:30에 일반 보안 서비스 테이블 /etc/gss/gsscred_db에서 중복 항목을 검사하고 제거합니다.

crontab 파일 내의 라인 구문에 대한 자세한 내용은 [109 페이지 “crontab 파일 항목의 구문”](#)을 참조하십시오.

crontab 파일은 /var/spool/cron/crontabs 디렉토리에 저장됩니다. root 외에 여러 crontab 파일이 Oracle Solaris 소프트웨어 설치 중 제공됩니다. 다음 표를 참조하십시오.

표 8-2 기본 crontab 파일

crontab 파일	기능
adm	계산

표 8-2 기본 crontab 파일 (계속)

crontab 파일	기능
lp	인쇄
root	일반 시스템 함수 및 파일 시스템 정리
sys	성능 데이터 수집
uucp	일반 uucp 정리

기본 crontab 파일 외에, 사용자는 crontab 파일을 만들어 자신의 시스템 작업을 예약할 수 있습니다. 다른 crontab 파일은 bob, mary, smith, jones와 같은 생성된 사용자 계정에서 이름을 따옵니다.

root 또는 다른 사용자에게 속하는 crontab 파일에 액세스하려면 슈퍼유저 권한이 필요합니다.

crontab 파일을 만들고, 편집, 표시, 제거하는 방법을 설명하는 절차는 후속 단원에서 설명합니다.

cron 데몬이 예약을 처리하는 방법

cron 데몬은 crontab 명령의 자동 예약을 관리합니다. cron 데몬의 역할은 /var/spool/cron/crontab 디렉토리에 crontab 파일이 있는지 검사하는 것입니다.

cron 데몬은 시작 시 다음 작업을 수행합니다.

- 새 crontab 파일을 검사합니다.
- 파일 내에 나열된 실행 시간을 읽습니다.
- 적절한 시간에 실행할 명령을 제출합니다.
- 업데이트된 crontab 파일에 관해 crontab 명령에서 알림을 받습니다.

대부분 똑같은 방법으로 cron 데몬은 at 파일의 예약을 제어합니다. 이러한 파일은 /var/spool/cron/atjobs 디렉토리에 저장됩니다. 또한 cron 데몬은 제출된 at 작업에 관해 crontab 명령에서 알림을 받습니다.

crontab 파일 항목의 구분

crontab 파일은 각 명령줄의 처음 5개 필드로 지정된 시간에 자동으로 실행하는 명령들(한 라인에 명령 하나씩)로 구성됩니다. 다음 표에 설명된 이러한 5개 필드는 공백으로 구분됩니다.

표 8-3 crontab 시간 필드에 허용 가능한 값

시간 필드	값
분	0-59
시	0-23
월의 일	1-31
월	1-12
주의 일	0-6 (0 = 일요일)

crontab 시간 필드에서 특수 문자를 사용할 때 다음 지침을 따르십시오.

- 공백을 사용하여 각 필드를 구분합니다.
- 逗를 사용하여 여러 값을 구분합니다.
- 하이픈을 사용하여 값 범위를 지정합니다.
- 별표를 와일드카드로 사용하여 모든 가능한 값을 포함합니다.
- 라인 시작 부분에 주석 마크(#)를 사용하여 주석이나 빈 라인을 나타냅니다.

예를 들어, 다음 crontab 명령 입력은 매월 1일과 15일, 오후 4시에 사용자 콘솔 창에 미리 알림을 표시합니다.

```
0 16 1,15 * * echo Timesheets Due > /dev/console
```

crontab 파일의 각 명령은 너무 길더라도 한 라인씩 구성해야 합니다. crontab 파일은 여러분의 캐리지 리턴을 인식하지 않습니다. crontab 항목 및 명령 옵션에 대한 자세한 내용은 [crontab\(1\)](#) 매뉴얼 페이지를 참조하십시오.

crontab 파일 만들기 및 편집

crontab 파일을 만드는 가장 간단한 방법은 `crontab -e` 명령을 사용하는 것입니다. 이 명령은 시스템 환경에 설정된 텍스트 편집기를 호출합니다. 시스템 환경의 기본 편집기는 `EDITOR` 환경 변수에 정의됩니다. 이 변수가 설정되지 않은 경우 `crontab` 명령이 기본 편집기인 `ed`를 사용합니다. 고급적 잘 아는 편집기를 선택해야 합니다.

다음 예는 편집기가 정의되었는지 확인하는 방법과 `vi`를 기본값으로 설정하는 방법을 보여줍니다.

```
$ which $EDITOR
$
$ EDITOR=vi
$ export EDITOR
```

crontab 파일을 만들 때 `/var/spool/cron/crontabs` 디렉토리에 자동으로 놓이고 사용자 이름이 부여됩니다. 슈퍼유저 권한이 있는 경우 다른 사용자 또는 `root`에 대해 crontab 파일을 만들거나 편집할 수 있습니다.

▼ crontab 파일을 만들거나 편집하는 방법

시작하기 전에 root 또는 다른 사용자에게 속하는 crontab 파일을 만들거나 편집하는 경우 슈퍼유저 또는 동등한 역할의 사용자여야 합니다. 역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

자신의 crontab 파일을 편집하기 위해 슈퍼유저일 필요는 없습니다.

1 새 crontab 파일을 만들거나 기존 파일을 편집합니다.

```
$ crontab -e [username]
```

여기서 *username*은 crontab 파일을 만들거나 편집할 사용자 계정의 이름을 지정합니다. 슈퍼유저 권한 없이 자신의 crontab 파일을 만들 수 있지만, root 또는 다른 사용자의 crontab 파일을 만들거나 편집하려면 슈퍼유저 권한이 있어야 합니다.



주의 - 실수로 crontab 명령을 아무 옵션 없이 입력한 경우 편집기의 인터럽트 문자를 누르십시오. 이 문자는 변경 사항을 저장하지 않고 종료하도록 허용합니다. 대신, 변경 사항을 저장하고 파일을 종료한 경우 기존 crontab 파일이 빈 파일로 겹쳐 쓰입니다.

2 crontab 파일에 명령줄을 추가합니다.

109 페이지 “crontab 파일 항목의 구문”에 설명된 구문을 따릅니다. crontab 파일은 /var/spool/cron/crontabs 디렉토리에 놓입니다.

3 crontab 파일 변경 사항을 확인합니다.

```
# crontab -l [username]
```

예 8-1 crontab 파일 만들기

다음 예는 다른 사용자의 crontab 파일을 만드는 방법을 보여줍니다.

```
# crontab -e jones
```

새 crontab 파일에 자동으로 추가된 다음 명령 입력은 매주 일요일 아침, 오전 1:00에 사용자의 홈 디렉토리에서 로그 파일을 제거합니다. 명령 입력이 출력을 재지정하지 않기 때문에 *.log 뒤의 명령줄에 재지정 문자가 추가됩니다. 이렇게 하면 명령이 올바르게 실행됩니다.

```
# This command helps clean up user accounts.
1 0 * * 0 rm /home/jones/*.log > /dev/null 2>&1
```

▼ crontab 파일이 존재하는지 확인하는 방법

- 사용자에 대해 crontab 파일이 존재하는지 확인하려면 `/var/spool/cron/crontabs` 디렉토리에 `ls -l` 명령을 사용합니다. 예를 들어, 다음 출력은 사용자 `jones` 및 `smith`에 대해 crontab 파일이 존재함을 보여줍니다.

```
$ ls -l /var/spool/cron/crontabs
-rw-r--r-- 1 root sys 190 Feb 26 16:23 adm
-rw----- 1 root staff 225 Mar 1 9:19 jones
-rw-r--r-- 1 root root 1063 Feb 26 16:23 lp
-rw-r--r-- 1 root sys 441 Feb 26 16:25 root
-rw----- 1 root staff 60 Mar 1 9:15 smith
-rw-r--r-- 1 root sys 308 Feb 26 16:23 sys
```

112 페이지 “crontab 파일을 표시하는 방법”에 설명된 `crontab -l` 명령을 사용하여 사용자의 crontab 파일 내용을 확인합니다.

crontab 파일 표시

`crontab -l` 명령은 `cat` 명령이 다른 유형의 파일 내용을 표시하는 것과 똑같은 방법으로 crontab 파일의 내용을 표시합니다. 이 명령을 사용하기 위해 디렉토리를 `/var/spool/cron/crontabs`(여기에 crontab 파일이 위치함)로 변경할 필요는 없습니다.

기본적으로 `crontab -l` 명령은 자신의 crontab 파일을 표시합니다. 다른 사용자에 속하는 crontab 파일을 표시하려면 슈퍼유저여야 합니다.

▼ crontab 파일을 표시하는 방법

시작하기 전에 root 또는 다른 사용자에 속하는 crontab 파일을 표시하려면 슈퍼유저 또는 동등한 역할의 사용자여야 합니다.

자신의 crontab 파일을 표시하기 위해 슈퍼유저 또는 동등한 역할의 사용자일 필요는 없습니다.

- crontab 파일을 표시합니다.

```
$ crontab -l [username]
```

여기서 `username`은 crontab 파일을 표시할 사용자 계정의 이름을 지정합니다. 다른 사용자의 crontab 파일을 표시하려면 슈퍼유저 권한이 필요합니다.



주의 - 실수로 crontab 명령을 아무 옵션 없이 입력한 경우 편집기의 인터럽트 문자를 누르십시오. 이 문자는 변경 사항을 저장하지 않고 종료하도록 허용합니다. 대신, 변경 사항을 저장하고 파일을 종료한 경우 기존 crontab 파일이 빈 파일로 겹쳐 쓰입니다.

예 8-2 crontab 파일 표시

이 예는 `crontab -l` 명령을 사용하여 사용자의 기본 `crontab` 파일의 내용을 표시하는 방법을 보여줍니다.

```
$ crontab -l
13 13 * * * chmod g+w /home1/documents/*.book > /dev/null 2>&1
```

예 8-3 기본 root crontab 파일 표시

이 예는 기본 `root crontab` 파일을 표시하는 방법을 보여줍니다.

```
$ suPassword:
Oracle Corporation SunOS 5.10 Generic Patch January 2005
# crontab -l
#ident "@(#)root 1.19 98/07/06 SMI" /* SVR4.0 1.1.3.1 */
#
# The root crontab should be used to perform accounting data collection.
#
#
10 3 * * * /usr/sbin/logadm
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
#10 3 * * * /usr/lib/krb5/kprop_script ___slave_kdcs___
```

예 8-4 다른 사용자의 crontab 파일 표시

이 예는 다른 사용자에게 속하는 `crontab` 파일을 표시하는 방법을 보여줍니다.

```
$ su
Password:
Oracle Corporation SunOS 5.10 Generic Patch January 2005
# crontab -l jones
13 13 * * * cp /home/jones/work_files /usr/backup/. > /dev/null 2>&1
```

crontab 파일 제거

기본적으로 `crontab` 파일 보호가 설정되므로 `rm` 명령으로 `crontab` 파일을 부주의하게 삭제할 수 없습니다. 대신, `crontab -r` 명령을 사용하여 `crontab` 파일을 제거하십시오.

기본적으로 `crontab -r` 명령은 자신의 `crontab` 파일을 제거합니다.

이 명령을 사용하기 위해 디렉토리를 `/var/spool/cron/crontabs`(여기에 `crontab` 파일이 위치함)로 변경할 필요는 없습니다.

▼ crontab 파일을 제거하는 방법

시작하기 전에 `root` 또는 다른 사용자에게 속하는 `crontab` 파일을 제거하려면 슈퍼유저 또는 동등한 역할의 사용자여야 합니다. 역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다.

자신의 crontab 파일을 제거하기 위해 슈퍼유저 또는 동등한 역할의 사용자일 필요는 없습니다.

1 crontab 파일을 제거합니다.

```
$ crontab -r [username]
```

여기서 *username*은 crontab 파일을 제거할 사용자 계정의 이름을 지정합니다. 다른 사용자의 crontab 파일을 제거하려면 슈퍼유저 권한이 필요합니다.



주의 - 실수로 crontab 명령을 아무 옵션 없이 입력한 경우 편집기의 인터럽트 문자를 누르십시오. 이 문자는 변경 사항을 저장하지 않고 종료하도록 허용합니다. 대신, 변경 사항을 저장하고 파일을 종료한 경우 기존 crontab 파일이 빈 파일로 겹쳐 쓰입니다.

2 crontab 파일이 제거되었는지 확인합니다.

```
# ls /var/spool/cron/crontabs
```

예 8-5 crontab 파일 제거

다음 예는 사용자 smith가 crontab -r 명령을 사용하여 자신의 crontab 파일을 제거하는 방법을 보여줍니다.

```
$ ls /var/spool/cron/crontabs
adm   jones   lp      root    smith   sys     uucp
$ crontab -r
$ ls /var/spool/cron/crontabs
adm   jones   lp      root    sys     uucp
```

crontab 명령에 대한 액세스 제어

/etc/cron.d 디렉토리의 두 파일 cron.deny 및 cron.allow를 사용하여 crontab 명령에 대한 액세스를 제어할 수 있습니다. 이러한 파일은 지정된 사용자만 자신의 crontab 파일 만들기, 편집, 표시, 제거와 같은 crontab 명령 작업을 수행하도록 허용합니다.

cron.deny 및 cron.allow 파일은 사용자 이름 목록(한 라인에 사용자 이름 하나씩)으로 구성됩니다.

이러한 액세스 제어 파일은 다음과 같이 작동합니다.

- cron.allow가 있는 경우 이 파일에 나열된 사용자만 crontab 파일을 만들거나, 편집, 표시, 제거할 수 있습니다.
- cron.allow가 없는 경우 cron.deny에 나열된 사용자를 제외한 모든 사용자가 crontab 파일을 제출할 수 있습니다.
- cron.allow도 없고 cron.deny도 없는 경우 crontab 명령을 실행하려면 슈퍼유저 권한이 필요합니다.

cron.deny 및 cron.allow 파일을 편집하거나 만들려면 슈퍼유저 권한이 필요합니다.

SunOS 소프트웨어 설치 중 생성된 cron.deny 파일은 다음 사용자 이름을 포함합니다.

```
$ cat /etc/cron.d/cron.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
```

기본 cron.deny 파일의 사용자 이름 중 아무도 crontab 명령에 액세스할 수 없습니다. 이 파일을 편집하여 crontab 명령에 액세스가 거부될 다른 사용자 이름을 추가할 수 있습니다.

기본 cron.allow 파일은 제공되지 않습니다. 그래서 Oracle Solaris 소프트웨어 설치 후에 모든 사용자(기본 cron.deny 파일에 나열된 사용자 제외)가 crontab 명령에 액세스할 수 있습니다. cron.allow 파일을 만들면 이러한 사용자만 crontab 명령에 액세스할 수 있습니다.

▼ crontab 명령 액세스를 거부하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 /etc/cron.d/cron.deny 파일을 편집하고 사용자 이름을 한 라인에 하나씩 추가합니다. crontab 명령에 대한 액세스가 거부될 사용자를 넣습니다.

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
username1
username2
username3
.
.
.
```

- 3 /etc/cron.d/cron.deny 파일이 새 항목을 포함하는지 확인합니다.

```
# cat /etc/cron.d/cron.deny
daemon
bin
```

```
nuucp
listen
nobody
noaccess
```

▼ crontab 명령 액세스를 지정된 사용자로 제한하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 `/etc/cron.d/cron.allow` 파일을 만듭니다.
- 3 `root` 사용자 이름을 `cron.allow` 파일에 추가합니다.
`root`를 파일에 추가하지 않으면 `crontab` 명령에 대한 슈퍼유저 액세스가 거부됩니다.
- 4 사용자 이름을 한 라인에 하나씩 추가합니다.
`crontab` 명령을 사용하도록 허용될 사용자를 넣습니다.

```
root
username1
username2
username3
.
.
.
```

예 8-6 crontab 명령 액세스를 지정된 사용자로 제한

다음 예는 사용자 이름 `jones`, `temp`, `visitor`가 `crontab` 명령에 액세스하지 못하게 막는 `cron.deny` 파일을 보여줍니다.

```
$ cat /etc/cron.d/cron.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
jones
temp
visitor
```

다음 예는 `cron.allow` 파일을 보여줍니다. 사용자 `root`, `jones`, `lp`, `smith`는 `crontab` 명령에 액세스할 수 있는 유일한 사용자입니다.

```
$ cat /etc/cron.d/cron.allow
root
jones
lp
smith
```

제한된 crontab 명령 액세스를 확인하는 방법

특정 사용자가 crontab 명령에 액세스할 수 있는지 확인하려면 사용자 계정으로 로그인한 동안 crontab -l 명령을 사용합니다.

```
$ crontab -l
```

사용자가 crontab 명령에 액세스할 수 있고 이미 crontab 파일을 만든 경우 파일이 표시됩니다. 그렇지 않으면, 사용자가 crontab 명령에 액세스할 수 있지만 crontab 파일이 없어서 다음과 비슷한 메시지가 표시됩니다.

```
crontab: can't open your crontab file
```

이 사용자가 cron.allow 파일(존재하는 경우)에 나열되거나 사용자가 cron.deny 파일에 나열되지 않습니다.

사용자가 crontab 명령에 액세스할 수 없을 경우 다음 메시지가 이전 crontab 파일이 존재하는지 여부를 표시합니다.

```
crontab: you are not authorized to use cron. Sorry.
```

이 메시지는 사용자가 cron.allow 파일(있는 경우)에 나열되지 않거나 사용자가 cron.deny 파일에 나열되어 있음을 의미합니다.

at 명령 사용(작업 맵)

작업	설명	수행 방법
at 작업 만들기	<p>at 명령을 사용하여 다음을 수행합니다.</p> <ul style="list-style-type: none"> 명령줄에서 at 유틸리티를 시작합니다. 실행할 명령 또는 스크립트를 한 라인에 하나씩 입력합니다. at 유틸리티를 종료하고 작업을 저장합니다. 	119 페이지 “at 작업을 만드는 방법”
at 대기열 표시	atq 명령을 사용하여 at 대기열을 표시합니다.	120 페이지 “at 대기열을 표시하는 방법”

작업	설명	수행 방법
at 작업 확인	atq 명령을 사용하여 특정 사용자에게 속하는 at 작업이 대기열에 제출되었는지 확인합니다.	120 페이지 “at 작업을 확인하는 방법”
at 작업 표시	at -l [job-id] 명령을 사용하여 대기열에 제출된 at 작업을 표시합니다.	120 페이지 “at 작업을 표시하는 방법”
at 작업 제거	at -r [job-id] 명령을 사용하여 대기열에서 at 작업을 제거합니다.	121 페이지 “at 작업을 제거하는 방법”
at 명령에 대한 액세스 거부	at 명령에 대한 사용자 액세스를 거부하려면 /etc/cron.d/at.deny 파일을 편집합니다.	122 페이지 “at 명령에 대한 액세스를 거부하는 방법”

단일 시스템 작업 예약(at)

다음 단원에서는 at 명령을 사용하여 다음 작업을 수행하는 방법을 설명합니다.

- 나중에 실행할 작업(명령 및 스크립트) 예약
- 이러한 작업을 표시 및 제거하는 방법
- at 명령에 대한 액세스를 제어하는 방법

기본적으로 사용자는 자신의 at 작업 파일을 만들고, 표시 및 제거할 수 있습니다. root 또는 다른 사용자에게 속하는 at 파일에 액세스하려면 슈퍼유저 권한이 있어야 합니다.

at 작업을 제출할 때 작업 식별 번호가 .a 확장자와 함께 지정됩니다. 이 지정이 작업의 파일 이름과 대기열 번호가 됩니다.

at 명령의 설명

at 작업 파일 제출은 다음 단계를 따릅니다.

1. at 유틸리티를 호출하고 명령 실행 시간을 지정합니다.
2. 나중에 실행할 명령 또는 스크립트를 입력합니다.

주- 이 명령 또는 스크립트의 출력이 중요하면 나중에 조사할 수 있도록 출력을 파일로 재지정해야 합니다.

예를 들어, 다음 at 작업은 7월의 마지막 날 자정에 사용자 계정 smith에서 core 파일을 제거합니다.

```
$ at 11:45pm July 31
at> rm /home/smith/*core*
at> Press Control-d
commands will be executed using /bin/csh
job 933486300.a at Tue Jul 31 23:45:00 2004
```

at 명령에 대한 액세스 제어

지정된 사용자만 at 작업에 대한 대기열 정보를 만들거나, 제거, 표시하도록 허용하여 at 명령에 대한 액세스를 제어하도록 파일을 설정할 수 있습니다. at 명령에 대한 액세스를 제어하는 파일인 /etc/cron.d/at.deny는 사용자 이름 목록(한 라인에 사용자 이름 하나씩)으로 구성됩니다. 이 파일에 나열된 사용자는 at 명령에 액세스할 수 없습니다.

SunOS 소프트웨어 설치 중 생성된 at.deny 파일은 다음 사용자 이름을 포함합니다.

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
```

수퍼유저 권한으로 at.deny 파일을 편집하여 제한할 at 명령 액세스를 가진 다른 사용자 이름을 추가할 수 있습니다.

▼ at 작업을 만드는 방법

- 1 at 유틸리티를 시작하여 작업을 실행할 시간을 지정합니다.

```
$ at [-m] time [date]
```

-m 작업이 완료된 후에 전자 메일을 보냅니다.

time 작업을 예약할 시를 지정합니다. 24 시간제에 따라 시를 지정하지 않으려면 am 또는 pm을 추가합니다. 허용 가능한 키워드는 midnight, noon, now입니다. 분은 선택 사항입니다.

date 월의 처음 3자 이상, 주의 일 또는 키워드 today나 tomorrow를 지정합니다.

- 2 at 프롬프트에서 실행할 명령 또는 스크립트를 한 라인에 하나씩 입력합니다.

각 라인의 끝에 Return을 누르면 여러 개의 명령을 입력할 수 있습니다.

- 3 at 유틸리티를 종료하고 Ctrl-D를 눌러 at 작업을 저장합니다.

at 작업에 대기열 번호(작업의 파일 이름)가 지정됩니다. 이 번호는 at 유틸리티를 종료할 때 표시됩니다.

예 8-7 at 작업 만들기

다음 예는 사용자 jones가 오후 7:30에 백업 파일을 제거하기 위해 만든 at 작업을 보여줍니다. 사용자는 작업이 완료된 후에 전자 메일 메시지를 받도록 -m 옵션을 사용했습니다.

```
$ at -m 1930
at> rm /home/jones/*.backup
at> Press Control-D
job 897355800.a at Thu Jul 12 19:30:00 2004
```

at 작업 실행을 확인한 전자 메일 메시지를 받았습니다.

```
Your "at" job "rm /home/jones/*.backup"
completed.
```

다음 예는 토요일 아침, 오전 4:00에 jones가 대규모 at 작업을 예약하는 방법을 보여줍니다. 작업 출력이 big.file이라는 파일로 재지정되었습니다.

```
$ at 4 am Saturday
at> sort -r /usr/dict/words > /export/home/jones/big.file
```

▼ at 대기열을 표시하는 방법

- at 대기열에 대기 중인 작업을 확인하려면 atq 명령을 사용합니다.

```
$ atq
```

이 명령은 생성된 at 작업에 대한 상태 정보를 표시합니다.

▼ at 작업을 확인하는 방법

- at 작업을 만들었는지 확인하려면 atq 명령을 사용합니다. 다음 예에서 atq 명령은 jones에 속하는 at 작업이 대기열에 제출되었음을 확인합니다.

```
$ atq
Rank   Execution Date   Owner   Job           Queue   Job Name
1st    Jul 12, 2004 19:30 jones  897355800.a   a       stdin
2nd    Jul 14, 2004 23:45 jones  897543900.a   a       stdin
3rd    Jul 17, 2004 04:00 jones  897732000.a   a       stdin
```

▼ at 작업을 표시하는 방법

- at 작업의 실행 시간에 대한 정보를 표시하려면 at -l 명령을 사용합니다.

```
$ at -l [job-id]
```

여기서 `-l job-id` 옵션은 상태를 표시할 작업의 식별 번호를 나타냅니다.

예 8-8 at 작업 표시

다음 예는 사용자가 제출한 모든 작업의 상태 정보를 제공하는 `at -l` 명령의 출력을 보여줍니다.

```
$ at -l
897543900.a    Sat Jul 14 23:45:00 2004
897355800.a    Thu Jul 12 19:30:00 2004
897732000.a    Tue Jul 17 04:00:00 2004
```

다음 예는 `at -l` 명령으로 단일 작업을 지정할 때 표시되는 출력을 보여줍니다.

```
$ at -l 897732000.a
897732000.a    Tue Jul 17 04:00:00 2004
```

▼ at 작업을 제거하는 방법

시작하기 전에 root 또는 다른 사용자에게 속하는 at 작업을 제거하려면 슈퍼유저 또는 동등한 역할의 사용자여야 합니다. 역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다.

자신의 at 작업을 제거하기 위해 슈퍼유저 또는 동등한 역할의 사용자일 필요는 없습니다.

1 작업을 실행하기 전에 대기열에서 at 작업을 제거합니다.

```
$ at -r [job-id]
```

여기서 `-r job-id` 옵션은 제거할 작업의 식별 번호를 지정합니다.

2 at -l(또는 atq) 명령을 사용하여 at 작업이 제거되었는지 확인합니다.

`at -l` 명령은 at 대기열에 남은 작업을 표시합니다. 지정된 식별 번호를 가진 작업이 나타나지 않아야 합니다.

```
$ at -l [job-id]
```

예 8-9 at 작업 제거

다음 예에서 사용자는 7월 17일 오전 4시에 실행하도록 예약된 at 작업을 제거하려고 합니다. 먼저, 사용자가 작업 식별 번호를 찾기 위해 at 대기열을 표시합니다. 그 다음, 사용자가 at 대기열에서 이 작업을 제거합니다. 마지막으로, 사용자가 이 작업이 대기열에서 제거되었는지 확인합니다.

```
$ at -l
897543900.a    Sat Jul 14 23:45:00 2003
897355800.a    Thu Jul 12 19:30:00 2003
```

```
897732000.a Tue Jul 17 04:00:00 2003
$ at -r 897732000.a
$ at -l 897732000.a
at: 858142000.a: No such file or directory
```

▼ at 명령에 대한 액세스를 거부하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 `/etc/cron.d/at.deny` 파일을 편집하고 at 명령의 사용이 금지될 사용자 이름을 한 라인에 하나씩 추가합니다.

```
daemon
bin
smtp
nuucp
listen
nobody
noaccess
username1
username2
username3
.
.
.
```

예 8-10 at 액세스 거부

다음 예는 사용자 smith 및 jones가 at 명령에 액세스할 수 없도록 편집한 `at.deny` 파일을 보여줍니다.

```
$ cat at.deny
daemon
bin
smtp
nuucp
listen
nobody
noaccess
jones
smith
```

▼ at 명령 액세스가 거부되었는지 확인하는 방법

- 사용자 이름이 `/etc/cron.d/at.deny` 파일에 올바르게 추가되었는지 확인하려면 사용자로 로그인한 동안 `at -l` 명령을 사용합니다. 사용자 `smith`가 `at` 명령에 액세스할 수 없으면 다음 명령이 표시됩니다.

```
# su smith
Password:
$ at -l
at: you are not authorized to use at. Sorry.
```

마찬가지로, 사용자가 `at` 작업을 제출하려면 다음 메시지가 표시됩니다.

```
$ at 2:30pm
at: you are not authorized to use at. Sorry.
```

이 메시지는 사용자가 `at.deny` 파일에 나열되었음을 확인합니다.

`at` 명령 액세스가 허용된 경우 `at -l` 명령이 아무것도 반환하지 않습니다.

시스템 계산 관리(작업)

이 장에서는 시스템 계산을 설정 및 유지 관리하는 방법을 설명합니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 126 페이지 “시스템 계산이란?”
- 131 페이지 “시스템 계산 설정”

확장 계산 사용에 대한 내용은 **시스템 관리 설명서: Oracle Solaris Containers-리소스 관리 및 Oracle Solaris 영역**의 4 장, “확장 계정(개요)”.

시스템 계산과 연관된 단계별 절차에 대한 내용은 130 페이지 “시스템 계산(작업 맵)”을 참조하십시오.

다양한 시스템 계산 보고서에 대한 참조 정보는 10 장, “시스템 계산(참조)”을 참조하십시오.

시스템 계산의 새로운 기능

이 절에서는 Oracle Solaris 릴리스에서 시스템 계산을 위해 새로 추가되거나 변경된 기능에 대해 설명합니다. 새로운 기능의 전체 목록 및 Oracle Solaris 릴리스에 대한 설명은 **Oracle Solaris 10 8/11 새로운 기능**을 참조하십시오.

Oracle Solaris 프로세스 계산 및 통계 개선

Oracle Solaris 10: 로드 평균화 `cpu usr/sys/idle` 및 계산 기능의 내부가 변경되었습니다. 미시상태 계산이 이전의 계산 방식으로 대체되었으며 항상 기본적으로 사용으로 설정됩니다. 따라서 약간 다른 프로세스 사용도와 타이밍 통계를 볼 수 있습니다.

미시상태 계산으로 전환하면 사용자 프로세스와 다양한 상태에서 사용자가 사용한 시간에 대한 더욱 정확한 데이터가 제공됩니다. 또한 이 정보를 사용하여 /proc 파일 시스템에서 더욱 정확한 로드 평균과 통계를 생성할 수 있습니다. 자세한 내용은 [proc\(4\)](#) 매뉴얼 페이지를 참조하십시오.

시스템 계산이란?

Oracle Solaris OS의 시스템 계산 소프트웨어는 사용자 연결 시간, 프로세스에 부과된 CPU 시간 및 디스크 사용량에 대한 데이터를 수집하고 기록할 수 있는 일련의 프로그램입니다. 이 데이터를 수집한 후에는 보고서를 생성하고 시스템 사용량에 대한 요금을 부과할 수 있습니다.

매일 또는 매월 단위로 시스템 계산을 사용할 수 있습니다. 또는 사용자별 디스크 사용량을 추적할 수 있습니다.

계산 프로그램을 사용하여 다음 작업을 수행할 수 있습니다.

- 시스템 사용량 모니터
- 성능 문제를 찾아서 수정하기
- 시스템 보안 유지

시스템 계산 프로그램을 설정한 후에는 대부분 스스로 실행됩니다.

시스템 계산의 작동 원리

먼저 계산 시작 스크립트를 루트의 crontab 파일에 입력하면 자동 계산이 설정됩니다. 그러면 계산 시작 스크립트를 cron 명령에 의해 자동으로 시작할 수 있습니다.

다음 개요는 시스템 계산 프로세스를 설명합니다.

1. 시스템 시작과 종료 사이에, 시스템 사용에 대한 원시 데이터(예: 사용자 로그인, 실행 프로세스, 데이터 저장소)가 계산 파일에 수집됩니다.
2. 정기적으로(대개 하루에 한번), /usr/lib/acct/runacct 스크립트가 다양한 계산 파일을 처리하고 누적 요약 파일과 일별 계산 보고서를 생성합니다. 그 다음, /usr/lib/acct/prdaily 스크립트가 일별 보고서를 인쇄합니다.
runacct 스크립트에 대한 자세한 내용은 [139 페이지 “runacct 스크립트”](#)를 참조하십시오.
3. 매월, monacct 스크립트를 실행하여 누적된 runacct 요약 파일을 처리하고 인쇄할 수 있습니다. monacct 스크립트로 생성된 요약 보고서는 매월 단위로 또는 다른 회계 기간에 사용자 요금을 청구하는 효율적인 수단입니다.

시스템 계산 구성 요소

시스템 계산 소프트웨어는 데이터를 요약 파일과 보고서로 구성하는 C 언어 프로그램 및 셸 스크립트를 제공합니다. 이러한 프로그램은 `/usr/lib/acct` 디렉토리에 있습니다. 계산 보고서는 `/var/adm/acct` 디렉토리에 있습니다.

일별 계산을 통해 다음 4가지 유형의 감사를 수행할 수 있습니다.

- 연결 계산
- 프로세스 계산
- 디스크 계산
- 요금 계산

연결 계산

연결 계산을 통해 다음 정보를 확인할 수 있습니다.

- 사용자가 로그인한 시간 길이
- tty 회선 사용 현황
- 시스템의 재부트 횟수
- 계산 소프트웨어가 설정/해제된 횟수

이 연결 세션 정보를 제공하기 위해 시스템이 다음 데이터를 저장합니다.

- 시간 조정 레코드
- 부트 시간
- 계산 소프트웨어가 설정/해제된 횟수
- 실행 레벨의 변경 사항
- 사용자 프로세스 만들기(login 프로세스와 init 프로세스)
- 프로세스의 종료

이러한 레코드는 `date`, `init`, `login`, `ttymon`, `acctwtmp`와 같은 시스템 프로그램 출력에서 생성됩니다. 이들은 `/var/adm/wtmpx` 파일에 저장됩니다.

`wtmpx` 파일의 항목은 다음 정보를 포함할 수 있습니다.

- 로그인 이름
- 장치 이름
- 프로세스 ID
- 항목 유형
- 입력 시간을 나타내는 시간 기록

프로세스 계산

프로세스 계산을 통해 시스템에서 실행되는 각 프로세스에 대해 다음 데이터를 추적할 수 있습니다.

- 프로세스를 사용하는 사용자의 사용자 ID 및 그룹 ID
- 프로세스의 시작 시간 및 경과 시간
- 프로세스의 CPU 시간(사용자 시간과 시스템 시간)
- 프로세스에서 사용된 메모리 양
- 프로세스에서 실행된 명령
- 프로세스를 제어하는 tty

프로세스를 종료할 때마다 `exit` 프로그램이 이 정보를 수집하여 `/var/adm/pacct` 파일에 기록합니다.

디스크 계산

디스크 계산을 통해 각 사용자가 디스크에 보관한 파일에 대해 다음 데이터를 수집하고 형식을 지정할 수 있습니다.

- 사용자의 사용자 이름 및 사용자 ID
- 사용자 파일에서 사용된 블록 수

이 데이터는 `/var/spool/cron/crontabs/root` 파일에 추가한 항목으로 결정된 간격마다 `/usr/lib/acct/dodisk` 셸 스크립트에 의해 수집됩니다. 이번에는 `dodisk` 스크립트가 `acctdisk` 및 `acctdusg` 명령을 호출합니다. 이러한 명령은 로그인 이름별로 디스크 사용량을 수집합니다.



주의 - `dodisk` 스크립트를 실행하여 수집된 정보는 `/var/adm/acct/nite/diskacct` 파일에 저장됩니다. 다음에 `dodisk` 스크립트를 실행할 때 이 정보를 겹쳐 씁니다. 따라서 `dodisk` 스크립트를 같은 날 두 번 실행하지 않도록 주의하십시오.

`acctdusg` 명령이 무작위로 쓰여진 파일에 요금을 과다 청구하여 파일에 손해를 입힐 수 있습니다. 이 문제는 `acctdusg` 명령이 파일 크기를 결정할 때 파일의 간접 블록을 읽지 않기 때문에 발생합니다. 오히려 `acctdusg` 명령은 파일의 `inode`에서 현재 파일 크기 값을 검사하여 파일 크기를 결정합니다.

요금 계산

`chargefee` 유틸리티는 사용자에게 제공된 특수 서비스의 과금을 `/var/adm/fee` 파일에 저장합니다. 특수 서비스는 파일 복원 같은 것입니다. 파일의 각 항목은 사용자 로그인 이름, 사용자 ID, 요금으로 구성됩니다. 이 파일을 매일 `runacct` 스크립트로 검사하여 새 항목을 계산 레코드로 병합합니다. 사용자 요금 청구를 위한 `chargefee` 스크립트 실행 지침은 134 페이지 “사용자 요금을 청구하는 방법”을 참조하십시오.

일별 계산의 작동 원리

다음은 일별 계산의 작동 원리를 단계별로 요약한 것입니다.

1. 시스템을 다중 사용자 모드로 전환하면 `/usr/lib/acct/startup` 프로그램이 실행됩니다. `startup` 프로그램이 일별 계산을 호출하는 여러 다른 프로그램을 실행합니다.
2. `acctwtmp` 프로그램이 "boot" 레코드를 `/var/adm/wtmpx` 파일에 추가합니다. 이 레코드에서 시스템 이름이 `wtmpx` 레코드의 사용자 이름으로 표시됩니다. 다음 표는 원시 계산 데이터의 수집 방법과 저장 위치를 요약한 것입니다.

<code>/var/adm</code> 의 파일	저장된 정보	작성자	형식
<code>wtmpx</code>	연결 세션	<code>login, init</code>	이진
	변경 사항	<code>date</code>	이진
	재부트	<code>acctwtmp</code>	이진
	종료	<code>shutacct</code>	이진
<code>pacctn</code>	프로세스	커널(프로세스가 끝날 때)	이진
		<code>turnacct switch</code> (이전 파일이 500개 블록에 도달하면 새 파일 생성)	이진
<code>fee</code>	특수 과금	<code>chargefee</code>	ASCII
<code>acct/nite/diskacct</code>	사용된 디스크 공간	<code>dodisk</code>	이진

3. `-on` 옵션으로 호출된 `turnacct` 스크립트가 프로세스 계산을 시작합니다. 특별히 `turnacct` 스크립트가 `accton` 프로그램을 `/var/adm/pacct` 인수와 함께 실행합니다.
4. 제거 셸 스크립트가 `runacct` 스크립트에 의해 `sum` 디렉토리에 남은 저장된 `pacct` 및 `wtmpx` 파일을 "정리"합니다.
5. `login` 및 `init` 프로그램이 `/var/adm/wtmpx` 파일에 레코드를 작성하여 연결 세션을 기록합니다. 날짜 변경 사항(날짜를 인수와 함께 사용)도 `/var/adm/wtmpx` 파일에 기록됩니다. `acctwtmp` 명령을 사용한 재부트 및 종료도 `/var/adm/wtmpx` 파일에 기록됩니다.
6. 프로세스가 끝날 때 커널이 `/var/adm/pacct` 파일에 `acct.h` 형식을 사용하여 프로세스당 하나의 레코드를 작성합니다.

매 시간마다, `cron` 명령이 `ckpacct` 스크립트를 실행하여 `/var/adm/pacct` 파일의 크기를 검사합니다. 파일이 500개 블록(기본값)을 넘으면 `turnacct switch` 명령이 실행됩니다. (프로그램이 `pacct` 파일을 `pacctn` 파일로 이동하고 새 파일을 만듭니다.)

이러한 레코드를 처리할 때 실패가 발생한 경우 runacct 스크립트를 다시 시작하려는 순간 여러 개의 작은 pacct 파일 사용의 장점이 발휘됩니다.

7. runacct 스크립트가 매일 밤 cron 명령으로 실행됩니다. runacct 스크립트가 계산 파일을 처리하여 사용자 이름별 명령 요약과 사용량 요약을 생성합니다. 계산 파일 /var/adm/pacctn, /var/adm/wtmpx, /var/adm/fee, /var/adm/acct/nite/diskacct가 처리됩니다.
8. /usr/lib/acct/prdaily 스크립트가 runacct 스크립트에 의해 매일 단위로 실행되어 /var/adm/acct/sum/rprtMMDD 파일에 일별 계산 정보를 기록합니다.
9. monacct 스크립트를 매월 단위로(또는 모든 회계 기간의 끝과 같이 사용자가 지정한 간격마다) 실행해야 합니다. monacct 스크립트가 runacct 스크립트에 의해 매일 업데이트된 sum 디렉토리에 저장된 데이터를 기반으로 하는 보고서를 만듭니다. 보고서를 만든 후에 monacct 스크립트가 sum 디렉토리를 "정리"하여 새로운 runacct 데이터를 위해 디렉토리의 파일을 준비합니다.

시스템 종료 시 발생 상황

shutdown 명령을 사용하여 시스템을 종료하면 shutacct 스크립트가 자동으로 실행됩니다. shutacct 스크립트가 /var/adm/wtmpx 파일에 **이유 레코드**를 기록하고 프로세스 계산을 끝니다.

시스템 계산(작업 맵)

작업	설명	수행 방법
시스템 계산 설정	다음 작업을 수행하여 시스템 계산을 설정합니다. <ul style="list-style-type: none"> ■ /etc/rc0.d/K22acct 및 /etc/rc2.d/S22acct 파일을 만듭니다. ■ /var/spool/cron/crontabs/adm 및 /var/spool/cron/crontabs/root crontab 파일을 수정합니다. 	132 페이지 “시스템 계산을 설정하는 방법”
사용자 요금 청구	/usr/lib/acct/chargefee <i>username amount</i> 명령을 실행합니다.	134 페이지 “사용자 요금을 청구하는 방법”
훼손된 wtmpx 파일 수정	wtmpx 파일을 이진 형식에서 ASCII 형식으로 변환합니다.	135 페이지 “훼손된 wtmpx 파일을 수정하는 방법”
tacct 오류 수정	prtacct 스크립트를 실행하여 /var/adm/acct/sum/tacctprev 파일을 검사합니다. 그 다음, /var/adm/acct/sum/tacct MMDD 파일에 최신 패치를 적용합니다. /var/adm/acct/sum/tacct 파일을 다시 만들어야 합니다.	135 페이지 “tacct 오류를 수정하는 방법”

작업	설명	수행 방법
runacct 스크립트 다시 시작	lastdate 파일과 잠금 파일을 제거합니다. 그 다음, runacct 스크립트를 수동으로 다시 시작합니다.	136 페이지 “runacct 스크립트를 다시 시작하는 방법”
시스템 계산을 일시적으로 사용 안함	adm crontab 파일을 편집하여 ckpacct, runacct, monacct 프로그램의 실행을 중지합니다.	137 페이지 “시스템 계산을 일시적으로 중지하는 방법”
시스템 계산을 영구적으로 사용 안함	adm crontab 파일에서 ckpacct, runacct, monacct 프로그램의 항목을 삭제합니다.	138 페이지 “시스템 계산을 영구적으로 사용 안함으로 설정하는 방법”

시스템 계산 설정

시스템이 다중 사용자 모드에 있는 동안 실행되도록 시스템 계산을 설정할 수 있습니다(실행 레벨 2). 일반적으로 이 작업은 다음 단계를 따릅니다.

1. /etc/rc0.d/K22acct 및 /etc/rc2.d/S22acct 시작 스크립트를 만듭니다.
2. /var/spool/cron/crontabs/adm 및 /var/spool/cron/crontabs/root crontab 파일을 수정합니다.

다음 표는 기본 계산 스크립트를 설명합니다.

표 9-1 기본 계산 스크립트

용도	계산 스크립트	매뉴얼 페이지	실행 빈도
/usr/adm/pacct 로그 파일의 크기를 검사하고 너무 크지 않은지 확인합니다.	ckpacct	acctsh(1M)	정기적으로
연결, 디스크, 요금 계산 정보를 처리합니다. 이 스크립트에서 처리하지 않을 계산 기능의 명령을 제거할 수 있습니다.	runacct	runacct(1M)	매일
매월 단위로 계산 요약 보고서를 생성합니다. 이 스크립트의 실행 빈도를 결정할 수 있습니다. 이 스크립트에서 사용하지 않을 계산 기능의 명령을 제거할 수 있습니다.	monacct	acctsh(1M)	회계 기간 단위로

기본적으로 실행할 계산 스크립트를 선택할 수 있습니다. 이러한 항목을 crontab 파일에 추가한 후에 시스템 계산이 자동으로 실행되어야 합니다.

▼ 시스템 계산을 설정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 필요한 경우, pkgadd 명령을 사용하여 시스템에 SUNWaccr 및 SUNWaccu 패키지를 설치합니다.
- 3 실행 레벨 2의 시작 스크립트로 /etc/init.d/acct를 설치합니다.

```
# ln /etc/init.d/acct /etc/rc2.d/S22acct
```
- 4 실행 레벨 0의 중지 스크립트로 /etc/init.d/acct를 설치합니다.

```
# ln /etc/init.d/acct /etc/rc0.d/K22acct
```
- 5 다음 라인을 adm crontab 파일에 추가하여 ckpacct, runacct, monacct 스크립트를 자동으로 시작합니다.

```
# EDITOR=vi; export EDITOR
# crontab -e adm
0 * * * * /usr/lib/acct/ckpacct
30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
30 7 1 * * /usr/lib/acct/monacct
```
- 6 다음 라인을 root crontab 파일에 추가하여 dodisk 스크립트를 자동으로 시작합니다.

```
# crontab -e
30 22 * * 4 /usr/lib/acct/dodisk
```
- 7 국가 공휴일과 지역 공휴일을 포함하도록 /etc/acct/holidays를 편집합니다.
자세한 내용은 holidays(4) 매뉴얼 페이지 및 이어지는 예제를 참조하십시오.
- 8 시스템을 재부트하거나, 다음을 입력하여 수동으로 시스템 계산을 시작합니다.

```
# /etc/init.d/acct start
```

예 9-1 계산 설정(adm crontab)

이 수정된 adm crontab은 ckpacct, runacct, monacct 스크립트의 항목을 포함합니다.

```
#ident "@(#)adm 1.5 92/07/14 SMI" /* SVr4.0 1.2 */
#
# The adm crontab file should contain startup of performance
```

```
# collection if the profiling and performance feature has been
# installed.
0 * * * * /usr/lib/acct/ckpacct
30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
30 7 1 * * /usr/lib/acct/monacct
```

예 9-2 계산 설정(root crontab)

이 수정된 root crontab은 dodisk 프로그램의 항목을 포함합니다.

```
#ident "@(#)root 1.19 98/07/06 SMI" /* SVr4.0 1.1.3.1 */
#
# The root crontab should be used to perform accounting data collection.
#
#
10 3 * * * /usr/sbin/logadm
15 3 * * 0 /usr/lib/fs/nfs/nfsfind
30 3 * * * [ -x /usr/lib/gss/gsscred_clean ] && /usr/lib/gss/gsscred_clean
30 22 * * 4 /usr/lib/acct/dodisk
```

예 9-3 계산 설정(/etc/acct/holidays)

다음 예는 /etc/acct/holidays 파일의 샘플을 보여줍니다.

```
* @(#)holidays January 1, 2004
*
* Prime/Nonprime Table for UNIX Accounting System
*
* Curr Prime Non-Prime
* Year Start Start
*
2004 0800 1800
*
* only the first column (month/day) is significant.
*
* month/day Company
* Holiday
*
1/1 New Years Day
7/4 Indep. Day
12/25 Christmas
```

사용자 요금 청구

요청별 특수 사용자 서비스를 제공하는 경우 chargefee 유틸리티를 실행하여 사용자 요금을 청구하고 싶을 수 있습니다. 특수 서비스에는 파일 복원이나 원격 인쇄가 포함됩니다. chargefee 유틸리티는 /var/adm/fee 파일에 과금을 기록합니다. runacct 유틸리티를 실행할 때마다 새 항목이 총 계산 레코드로 병합됩니다.

자세한 내용은 [acctsh\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

▼ 사용자 요금을 청구하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 특수 서비스에 대해 사용자 요금을 부과합니다.

```
# /usr/lib/acct/chargefee username amount
```

username 요금을 청구할 사용자 계정을 지정합니다.

amount 사용자 요금을 청구할 단위 수를 지정합니다. 이 값은 파일 인쇄나 복원과 같은 어떤 작업에 따라 사용자에게 부과하기로 정한 모든 단위입니다.
chargefee 유틸리티를 호출하고 특정 작업에 대해 사용자에게 과금하는 스크립트를 작성해야 합니다.

예 9-4 사용자 요금 청구

다음 예에서 사용자 `print_customer`에 10 단위로 요금이 부과됩니다.

```
# /usr/lib/acct/chargefee print_customer 10
```

계산 정보 유지 관리

이 단원에서는 훼손된 시스템 계산 파일을 수정하는 방법과 `runacct` 스크립트를 다시 시작하는 방법을 설명합니다.

훼손된 파일 및 `wtmpx` 오류 수정

불행하게도, 시스템 계산은 그다지 간단하지 않습니다. 때때로 파일이 훼손되거나 손실됩니다. 어떤 파일은 간단히 무시하거나 백업에서 복원할 수 있습니다. 그러나 어떤 파일은 시스템 계산의 무결성을 유지하도록 수정해야 합니다.

`wtmpx` 파일이 일일 시스템 계산 운영에 대부분의 문제를 일으키는 것 같습니다. 날짜를 수동으로 바꾸고 시스템이 다중 사용자 모드에 있을 때 일련의 날짜 변경 레코드가 `/var/adm/wtmpx` 파일에 기록됩니다. `wtmpfix` 유틸리티는 날짜 변경이 발생할 때 `wtmp` 레코드의 시간 기록을 조정하도록 만들어졌습니다. 그러나 날짜 변경과 재부트가 결합하면 `wtmpfix` 유틸리티를 놓치고 `acctcon` 프로그램이 실패하게 됩니다.

▼ 훼손된 wtmpx 파일을 수정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 `/var/adm` 디렉토리로 변경합니다.
- 3 `wtmpx` 파일을 이진 형식에서 ASCII 형식으로 변환합니다.

```
# /usr/lib/acct/fwtmp < wtmpx > wtmpx.ascii
```
- 4 `wtmpx.ascii` 파일을 편집하여 훼손된 레코드를 삭제합니다.
- 5 `wtmpx.ascii` 파일을 다시 이진 파일로 변환합니다.

```
# /usr/lib/acct/fwtmp -ic < wtmpx.ascii > wtmpx
```


자세한 내용은 `fwtmp(1M)` 매뉴얼 페이지를 참조하십시오.

tacct 오류 수정

시스템 리소스에 대해 사용자에게 과금하는 경우 `/var/adm/acct/sum/tacct` 파일의 무결성이 매우 중요합니다. 때때로 특이한 `tacct` 레코드가 음수, 중복 사용자 ID 또는 사용자 ID 65535로 나타납니다. 먼저, 파일을 인쇄하는 `prtacct` 스크립트를 사용하여 `/var/adm/acct/sum/tacctprev` 파일을 검사합니다. 내용이 모두 정상이면 `/var/adm/acct/sum/tacct MMDD` 파일에 최신 패치를 적용합니다. 그 다음, `/var/adm/acct/sum/tacct` 파일을 다시 만듭니다. 다음 단계는 간단한 패치 절차를 설명합니다.

▼ tacct 오류를 수정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 `/var/adm/acct/sum` 디렉토리로 변경합니다.
- 3 `tacct MMDD` 파일을 이진 형식에서 ASCII 형식으로 변환합니다.

```
# /usr/lib/acct/acctmerg -v < tacctMMDD > xtacct
```

*MMDD*는 월, 일을 나타내는 2자리 숫자 쌍입니다.

- 4 **xtacct** 파일을 편집하여 훼손된 레코드를 제거하고 중복 레코드를 다른 파일에 씁니다.
- 5 **xtacct** 파일을 ASCII 형식에서 이진 형식으로 변환합니다.

```
# /usr/lib/acct/acctmerg -i < xtacct > tacctMMDD
```
- 6 **tacctprev** 및 **tacct.MMDD** 파일을 **tacct** 파일로 병합합니다.

```
# /usr/lib/acct/acctmerg < tacctprev tacctMMDD > tacct
```

runacct 스크립트 다시 시작

몇 가지 이유로 **runacct** 스크립트를 실패할 수 있습니다.

다음이 가장 흔한 이유입니다.

- 시스템 충돌
- /var 디렉토리의 공간 부족
- 훼손된 **wtmpx** 파일

active.MMDD 파일이 존재하면 먼저 오류 메시지가 있는지 확인합니다. **active** 및 **lock** 파일이 존재하면 **fd2log** 파일에 관련 메시지가 있는지 확인합니다.

인수 없이 실행된 **runacct** 스크립트에서 이 호출이 그날의 첫번째 호출이라고 가정합니다. **runacct** 스크립트가 다시 시작될 경우 인수 *MMDD*가 필요하고 **runacct** 스크립트가 계산을 재실행하는 월, 일을 지정합니다. 프로세싱의 시작점은 **statefile** 파일 내용에 기반합니다. **statefile** 파일을 대체하려면 명령줄에 원하는 상태를 입력합니다. 사용 가능한 상태에 대한 설명은 **runacct(1M)** 매뉴얼 페이지를 참조하십시오.



주의 - **runacct** 프로그램을 수동으로 실행할 때 사용자 **adm**으로 실행해야 합니다.

▼ runacct 스크립트를 다시 시작하는 방법

- 1 **/var/adm/acct/nite** 디렉토리로 변경합니다.

```
$ cd /var/adm/acct/nite
```
- 2 **lastdate** 파일과 **lock*** 파일(있는 경우)을 제거합니다.

```
$ rm lastdate lock*
```

lastdate 파일은 runacct 프로그램이 마지막 실행된 날짜를 포함합니다. 다음 단계에서 runacct 스크립트를 다시 시작하면 이 파일이 다시 생성됩니다.

3 runacct 스크립트 다시 시작

```
$ /usr/lib/acct/runacct MMDD [state] 2> /var/adm/acct/nite/fd2log &
MMDD 2자리 숫자로 지정된 월, 일입니다.
```

state runacct 스크립트 프로세싱을 시작할 상태나 시작점을 지정합니다.

시스템 계산 중지 및 사용 안함

시스템 계산을 일시적으로 중지하거나 영구적으로 사용 안함으로 설정할 수 있습니다.

▼ 시스템 계산을 일시적으로 중지하는 방법

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 adm crontab 파일을 편집하면서 적절한 라인을 주석 처리하여 ckpacct, runacct, monacct 프로그램의 실행을 중지합니다.

```
# EDITOR=vi; export EDITOR
# crontab -e adm
#0 * * * * /usr/lib/acct/ckpacct
#30 2 * * * /usr/lib/acct/runacct 2> /var/adm/acct/nite/fd2log
#30 7 1 * * /usr/lib/acct/monacct
```

3 root crontab 파일을 편집하면서 적절한 라인을 주석 처리하여 dodisk 프로그램의 실행을 중지합니다.

```
# crontab -e
#30 22 * * 4 /usr/lib/acct/dodisk
```

4 시스템 계산 프로그램을 중지합니다.

```
# /etc/init.d/acct stop
```

5 (선택 사항) crontab 파일에서 새로 추가된 주석 기호를 제거합니다.

6 시스템 계산 프로그램을 다시 시작하여 시스템 계산을 다시 사용으로 설정합니다.

```
# /etc/init.d/acct start
```

▼ 시스템 계산을 영구적으로 사용 안함으로 설정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 `adm crontab` 파일을 편집하고 `ckpacct`, `runacct`, `monacct` 프로그램의 항목을 삭제합니다.

```
# EDITOR=vi; export EDITOR
# crontab -e adm
```
- 3 `root crontab` 파일을 편집하고 `dodisk` 프로그램의 항목을 삭제합니다.

```
# crontab -e
```
- 4 실행 레벨 2의 시작 스크립트를 제거합니다.

```
# unlink /etc/rc2.d/S22acct
```
- 5 실행 레벨 0의 중지 스크립트를 제거합니다.

```
# unlink /etc/rc0.d/K22acct
```
- 6 시스템 계산 프로그램을 중지합니다.

```
# /etc/init.d/acct stop
```

시스템 계산(참조)

이 장은 시스템 계산에 대한 참조 정보를 제공합니다.

이 장에서 다루는 참조 정보는 다음과 같습니다.

- 139 페이지 “runacct 스크립트”
- 141 페이지 “일별 계산 보고서”
- 148 페이지 “시스템 계산 파일”

시스템 계산 작업에 대한 자세한 내용은 9 장, “시스템 계산 관리(작업)”를 참조하십시오.

runacct 스크립트

기본 일별 계산 스크립트인 runacct는 일반적으로 정상 업무 시간 외의 시간에 cron 명령으로 호출됩니다. runacct 스크립트는 연결, 요금, 디스크, 프로세스 계산 파일을 처리합니다. 또한 요금 청구 목적으로 prdaily 및 monacct 스크립트에서 사용할 일별/누적 요약 파일을 준비합니다.

runacct 스크립트는 오류가 발생할 경우 파일이 손상되지 않도록 보호합니다.

다음 작업을 수행하기 위해 일련의 보호 방식이 사용됩니다.

- 오류 인식
- 지능형 진단 제공
- 최소한의 개입으로 runacct 스크립트를 다시 시작할 수 있는 방법으로 처리 완료

이 스크립트는 active 파일에 설명적 메시지를 작성하여 진행률을 기록합니다. runacct 스크립트에서 사용된 파일은 특별한 언급이 없는 한 /var/adm/acct/nite 디렉토리에 있다고 가정합니다. runacct 스크립트 실행 동안 모든 진단 출력은 fd2log 파일에 기록됩니다.

runacct 스크립트를 호출할 때 lock 및 lock1 파일을 생성합니다. 이러한 파일은 runacct 스크립트의 동시 실행을 막는 데 사용됩니다. runacct 프로그램을 호출할 때 이러한

파일이 있으면 오류 메시지를 출력합니다. `lastdate` 파일에는 runacct 스크립트가 마지막 호출된 월, 일이 있어서 하루에 한 번 이상 실행되는 것을 막을 수 있습니다.

runacct 스크립트가 오류를 감지하면 다음이 발생합니다.

- 메시지를 콘솔에 작성
- root 및 adm에 전자 메일 보내기
- 잠금이 제거될 수 있음
- 진단 저장
- 실행 종료

runacct 스크립트 다시 시작 방법에 대한 지침은 136 페이지 “runacct 스크립트를 다시 시작하는 방법”을 참조하십시오.

runacct 스크립트가 다시 시작되도록 허용하려면 프로세싱을 별도의 재진입 상태로 구분합니다. `statefile` 파일은 마지막으로 완료된 상태를 추적하는 데 사용됩니다. 각 상태가 완료되면 `statefile` 파일이 다음 상태를 반영하도록 업데이트됩니다. 상태 처리가 완료되고 나면 `statefile` 파일을 읽어 다음 상태가 처리됩니다. runacct 스크립트가 CLEANUP 상태에 도달하면 잠금을 제거하고 종료합니다. 다음 표에 나타난 대로 상태가 실행됩니다.

표 10-1 runacct 스크립트의 상태

상태	설명
SETUP	turnacct switch 명령을 실행하여 새 pacct 파일을 만듭니다. <code>/var/adm/pacct n</code> 프로세스 계산 파일(단, pacct 파일 제외)이 <code>/var/adm/Spacct n.MMDD</code> 파일로 이동합니다. <code>/var/adm/wtmpx</code> 파일이 <code>/var/adm/acct/nite/wtmp.MMDD</code> 파일(현재 시간 레코드가 끝에 추가됨)로 이동하고 새 <code>/var/adm/wtmp</code> 파일이 생성됩니다. <code>closewtmp</code> 및 <code>utmp2wtmp</code> 프로그램이 <code>wtmp.MMDD</code> 파일과 새 <code>wtmpx</code> 파일에 레코드를 추가하여 현재 로그인한 사용자를 알아냅니다.
WTMPFIX	wtmpfix 프로그램은 정확성을 위해 nite 디렉토리의 wtmp.MMDD 파일을 검사합니다. 어떤 날짜를 변경하면 acctcon 프로그램이 실패하기 때문에 날짜 변경 레코드가 나타날 경우 wtmpfix 프로그램이 wtmpx 파일에서 시간 기록을 조정하려고 시도합니다. 또한 이 프로그램은 wtmpx 파일에서 훼손된 항목을 삭제합니다. 수정된 버전의 wtmp.MMDD 파일이 tmpwtmp 파일에 기록됩니다.
CONNECT	acctcon 프로그램은 ctacct.MMDD 파일에 연결 계산 레코드를 기록하는 데 사용됩니다. 이러한 레코드는 tacct.h 형식입니다. 또한 acctcon 프로그램은 lineuse 및 reboots 파일을 만듭니다. reboots 파일은 wtmpx 파일에서 발견된 모든 부트 레코드를 기록합니다.
PROCESS	acctprc 프로그램은 <code>/var/adm/Spacct n.MMDD</code> 프로세스 계산 파일을 <code>ptacctn.MMDD</code> 파일의 완전한 계산 레코드로 변환하는 데 사용됩니다. Spacct 및 ptacct 파일은 숫자로 상호 연관되므로 runacct 스크립트가 실패할 경우 Spacct 파일이 처리되지 않습니다.

표 10-1 runacct 스크립트의 상태 (계속)

상태	설명
MERGE	acctmerg 프로그램은 프로세스 계산 레코드를 연결 계산 레코드와 병합하여 daytacct 파일을 만듭니다.
FEES	acctmerg 프로그램은 ASCII tacct 레코드를 fee 파일에서 daytacct 파일로 병합합니다.
DISK	dodisk 스크립트는 diskacct 파일을 만듭니다. dodisk 스크립트가 실행된 경우 diskacct 파일을 만들면, DISK 프로그램이 이 파일을 daytacct 파일로 병합하고 diskacct 파일을 /tmp/diskacct.MMDD 파일로 이동합니다.
MERGETACCT	acctmerg 프로그램은 daytacct 파일을 누적 총 계산 파일인 sum/tacct 파일과 병합합니다. 매일 daytacct 파일이 sum/tacct.MMDD 파일에 저장되므로 sum/tacct 파일이 훼손되거나 손실된 경우 다시 만들 수 있습니다.
CMS	acctcms 프로그램이 여러 번 실행됩니다. 이 프로그램을 처음 실행하여 Spacct n 파일로 명령 요약 생성하고 sum/daycms 파일에 데이터를 기록합니다. 그 다음, acctcms 프로그램을 실행하여 sum/daycms 파일을 sum/cms 누적 명령 요약 파일과 병합합니다. 마지막으로, acctcms 프로그램을 실행하여 sum/daycms 및 sum/cms 파일에서 각각 nite/daycms 및 nite/cms라는 ASCII 명령 요약 파일을 생성합니다. lastlogin 프로그램은 /var/adm/acct/sum/loginlog 로그 파일을 만드는 데 사용됩니다. 이 파일은 각 사용자의 마지막 로그인 시간을 보고합니다. runacct 스크립트가 자정 이후 실행되면 사용자의 마지막 로그인 시간을 보여주는 날짜가 하루씩 틀립니다.
USEREXIT	이 시점에서 설치 종속적(로컬) 계산 프로그램을 실행할 수 있습니다. runacct 스크립트는 이 프로그램이 /usr/lib/acct/runacct.local 프로그램을 호출할 것으로 기대합니다.
CLEANUP	이 상태는 임시 파일을 정리하고, prdaily 스크립트를 실행하고 sum/rpt.MMDD 파일에 출력을 저장하며, 잠금을 제거한 후 종료합니다.



주의 - CLEANUP 상태의 runacct 스크립트를 다시 시작할 때 마지막 ptacct 파일이 완료되지 않으므로 이 파일을 제거하십시오.

일별 계산 보고서

runacct 셸 스크립트는 각 호출마다 5개의 기본 보고서를 생성합니다. 다음 표는 이러한 보고서를 설명합니다.

표 10-2 일별 계산 보고서

보고서 유형	설명
142 페이지 “일별 보고서”	tty 숫자로 단말기 회선 사용률을 표시합니다.
143 페이지 “일별 사용량 보고서”	사용자 ID 순서로 나열된, 사용자별 시스템 리소스의 사용량을 나타냅니다.
144 페이지 “일별 명령 요약”	메모리 사용 내림차순으로 나열된, 명령별 시스템 리소스의 사용량을 나타냅니다. 다시 말해서, 가장 많은 메모리를 사용한 명령이 맨 처음 나옵니다. 이와 동일한 정보가 월별 명령 요약에 월별로 보고됩니다.
145 페이지 “월별 명령 요약”	monacct 프로그램의 마지막 호출 이후에 누적된 데이터를 반영하는 누적 요약입니다.
146 페이지 “마지막 로그인 보고서”	연대순으로 나열된, 각 사용자가 로그인한 마지막 시간을 표시합니다.

일별 보고서

이 보고서는 각 사용된 단말기 회선에 대한 정보를 제공합니다. 일별 보고서의 샘플은 다음과 같습니다.

Jan 16 02:30 2004 DAILY REPORT FOR venus Page 1

```
from Mon Jan 15 02:30:02 2004
to Tue Jan 16 02:30:01 2004
1      runacct
1      acctcon
```

TOTAL DURATION IS 1440 MINUTES

LINE	MINUTES	PERCENT	# SESS	# ON	# OFF
console	868	60	1	1	2
TOTALS	868	--	1	1	2

from 및 to 라인은 보고서에 반영된 기간을 지정합니다. 이 기간은 마지막 일별 보고서가 생성된 시간부터 현재 일별 보고서가 생성된 시간까지입니다. 그 다음, acctwtmp 프로그램으로 /var/adm/wtmpx 파일에 기록된 시스템 재부트, 종료, 전원 장애 복구 및 기타 레코드가 나타납니다. 자세한 내용은 acct(1M) 매뉴얼 페이지를 참조하십시오.

보고서의 두번째 부분은 단말기 회선 이용률을 분석합니다. TOTAL DURATION은 (단말기 회선을 통해 액세스 가능한) 다중 사용자 모드로 시스템이 실행된 기간을 알려줍니다. 다음 목록은 일별 보고서로 제공된 데이터를 설명합니다.

LINE 단말기 회선 또는 액세스 포트입니다.

MINUTES 계산 기간 동안 회선이 사용된 시간(분)입니다.

PERCENT TOTAL DURATION을 MINUTES 수로 나눈 값입니다.

- # SESS 이 회선이나 포트가 로그인 세션 동안 액세스된 횟수입니다.
- # ON SESS와 동일합니다. (이 열은 더 이상 의미가 없습니다. 이전에, 이 열은 회선/포트가 사용자 로그인에 사용된 횟수를 나열했습니다.)
- # OFF 해당 회선에서 발생한 사용자 로그아웃 및 인터럽트 횟수입니다. 일반적으로, 인터럽트는 시스템을 다중 사용자 모드로 가져온 후에 ttymon을 처음 호출할 때 포트에 발생합니다. # OFF가 # SESS를 크게 초과할 경우 다중 송신기, 모뎀 또는 케이블 상태가 나빠집니다. 또는, 잘못된 연결이 어딘가 존재합니다. 가장 흔한 원인은 다중 송신기에서 연결이 끊긴 케이블 배선입니다.

실제 시간 동안 /var/adm/wtmpx 파일은 연결 계산이 파생되는 파일이므로 이를 모니터링해야 합니다. wtmpx 파일이 급속히 커지면 다음 명령을 실행하여 가장 노이즈가 심한 tty 회선을 확인하십시오.

```
# /usr/lib/acct/acctcon -l file < /var/adm/wtmpx
```

인터럽트가 자주 발생하면 일반 시스템 성능에 영향을 미칩니다. 추가적으로 wtmp 파일이 훼손될 수 있습니다. 이 문제를 해결하려면 135 페이지 “[훼손된 wtmpx 파일을 수정하는 방법](#)”을 참조하십시오.

일별 사용량 보고서

일별 사용량 보고서는 사용자별 시스템 리소스 사용률을 분석합니다. 이 보고서의 샘플은 다음과 같습니다.

Jan 16 02:30 2004 DAILY USAGE REPORT FOR skisun Page 1

UID	LOGIN NAME	CPU PRIME	(MINS) NPRIME	KCORE- PRIME	MINS NPRIME	CONNECT PRIME	(MINS) NPRIME	DISK BLOCKS	# OF PROCS	# OF SESS	# OF DISK SAMPLES	FEE
0	TOTAL	72	148	11006173	51168	26230634	57792	539	330	0	2150	1
0	root	32	76	11006164	33664	26230616	22784	0	0	0	127	0
4	adm	0	0	22	51	0	0	0	420	0	0	0
101	rimmer	39	72	894385	1766020	539	330	0	1603	1	0	0

다음 표는 일별 사용량 보고서로 제공된 데이터를 설명합니다.

표 10-3 일별 사용량 보고서 데이터

열	설명
UID	사용자 ID 번호입니다.
LOGIN NAME	사용자의 로그인(또는 사용자) 이름입니다. 다중 로그인 이름을 가진 사용자를 식별합니다.

표 10-3 일별 사용량 보고서 데이터 (계속)

일	설명
CPU (MINS)	사용자 프로세스가 중앙 처리 장치를 사용한 시간(분)입니다. PRIME 및 NPRIME(nonprime) 사용률로 나뉩니다. 계산 시스템의 이 데이터 버전은 /etc/acct/holidays 파일에 있습니다.
KCORE-MINS	실행 중 프로세스가 사용하는 메모리의 누적 측정량을 분당 킬로바이트 세그먼트 단위로 나타냅니다. PRIME 및 NPRIME 사용률로 나뉩니다.
CONNECT (MINS)	사용자가 시스템에 로그인한 시간(분) 또는 "실제 시간"입니다. PRIME 및 NPRIME 사용률로 나뉩니다. 이러한 숫자가 높고 # OF PROCS가 낮으면 사용자가 아침에 처음 로그인한 후 그날 내내 단말기에 거의 손대지 않았다고 추정할 수 있습니다.
DISK BLOCKS	acctdusg 프로그램의 출력으로, 디스크 계산 프로그램을 실행하고 계산 레코드(daytacct)를 병합합니다. 계산 목적을 위해 블록은 512바이트입니다.
# OF PROCS	사용자가 호출한 프로세스 수입니다. 많은 수가 나타나면 사용자가 셸 프시저를 통제하지 못했을 수 있습니다.
# OF SESS	사용자가 시스템에 로그인한 횟수입니다.
# DISK SAMPLES	DISK BLOCKS의 평균 수를 얻기 위해 디스크 계산이 실행된 횟수입니다.
FREE	chargefee 스크립트로 사용자에게 부과된 총 단위 누적을 나타내는 자주 사용되지 않는 필드입니다.

일별 명령 요약

일별 명령 요약 보고서는 명령별 시스템 리소스 사용량을 표시합니다. 이 보고서로 가장 많이 사용된 명령을 식별할 수 있습니다. 그 다음, 이러한 명령이 시스템 리소스를 사용하는 방법에 준하여 시스템을 가장 잘 조정하는 방법을 예측할 수 있습니다.

이러한 보고서는 TOTAL KCOREMIN 기준으로 정렬되며, 임시 측정이지만 종종 시스템의 드레인을 계산하는 데 유용합니다.

일별 명령 요약의 샘플은 다음과 같습니다.

COMMAND NAME	NUMBER CMDS	TOTAL COMMAND SUMMARY							
		TOTAL KCOREMIN	TOTAL CPU-MIN	TOTAL REAL-MIN	MEAN SIZE-K	MEAN CPU-MIN	HOG FACTOR	CHARS TRNSFD	BLOCKS READ
TOTALS	2150	1334999.75	219.59	724258.50	6079.48	0.10	0.00	397338982	419448
netscape	43	2456898.50	92.03	54503.12	26695.51	2.14	0.00	947774912	225568
adeptedi	7	88328.22	4.03	404.12	21914.95	0.58	0.01	93155160	8774
dtmail	1	54919.17	5.33	17716.57	10308.94	5.33	0.00	213843968	40192
acroread	8	31218.02	2.67	17744.57	11682.66	0.33	0.00	331454464	11260
dtwm	1	16252.93	2.53	17716.57	6416.05	2.53	0.00	158662656	12848
dtterm	5	4762.71	1.30	76300.29	3658.93	0.26	0.00	33828352	11604

dtaction	23	1389.72	0.33	0.60	4196.43	0.01	0.55	18653184	539
dtessio	1	1174.87	0.24	17716.57	4932.97	0.24	0.00	23535616	5421
dtcm	1	866.30	0.18	17716.57	4826.21	0.18	0.00	3012096	6490

다음 목록은 일별 명령 요약으로 제공된 데이터를 설명합니다.

COMMAND NAME	명령 이름입니다. 모든 셸 프로시저는 sh 이름 아래에 함께 모이는데, 프로세스 계산 시스템에서 객체 모듈만 보고하기 때문입니다. a.out 또는 core 또는 다른 이상한 이름의 프로그램이 발생하는 빈도를 모니터해야 합니다. acctcom 프로그램을 사용하여 누가 이상한 이름의 명령을 실행했는지, 슈퍼유저 권한이 사용되었는지 확인해야 합니다.
NUMBER CMDS	이 명령이 실행된 총 횟수입니다.
TOTAL KCOREMIN	런타임에 프로세스에서 사용된 메모리의 전체 누적 측정량을 분당 킬로바이트 세그먼트 단위로 나타냅니다.
TOTAL CPU-MIN	이 프로그램이 누적한 총 처리 시간입니다.
TOTAL REAL-MIN	이 프로그램이 누적한 총 실제 시간(벽시계) 분입니다.
MEAN SIZE-K	NUMBER CMDS 로 반영된 호출 수에 대한 TOTAL KCOREMIN의 평균입니다.
MEAN CPU-MIN	NUMBER CMDS 및 TOTAL CPU-MIN에서 파생된 평균입니다.
HOG FACTOR	총 CPU 시간을 경과 시간으로 나눈 값입니다. 시스템 가용성 대 시스템 사용률의 비율을 나타내며, 실행 중 프로세스에서 소비한 총 사용 가능한 CPU 시간의 상대적 측정값입니다.
CHARS TRNSFD	읽기 및 쓰기 시스템 호출로 전송된 총 문자 수입니다. 오버플로우 때문에 음수일 수 있습니다.
BLOCKS READ	프로세스가 수행한 물리적 블록 읽기 및 쓰기의 총 개수입니다.

월별 명령 요약

일별 명령 요약 및 월별 명령 요약 보고서의 형식은 실질적으로 똑같습니다. 그러나 일별 요약 보고서는 현재 계산 기간에만 해당하고, 월별 요약 보고서는 회계 기간 시작일에서 현재 날짜까지입니다. 즉, 월별 보고서는 monacct 프로그램의 마지막 호출 이후에 누적된 데이터를 반영하는 누적 요약입니다.

월별 명령 요약의 샘플은 다음과 같습니다.

Jan 16 02:30 2004 MONTHLY TOTAL COMMAND SUMMARY Page 1

TOTAL COMMAND SUMMARY									
COMMAND NAME	NUMBER CMDS	TOTAL KCOREMIN	TOTAL CPU-MIN	TOTAL REAL-MIN	MEAN SIZE-K	MEAN CPU-MIN	HOG FACTOR	CHARS TRNSFD	BLOCKS READ

TOTALS	42718	4398793.50	361.92	956039.00	12154.09	0.01	0.00	16100942848	825171
netscape	789	3110437.25	121.03	79101.12	25699.58	0.15	0.00	3930527232	302486
adeptedi	84	1214419.00	50.20	4174.65	24193.62	0.60	0.01	890216640	107237
acroread	145	165297.78	7.01	18180.74	23566.84	0.05	0.00	1900504064	26053
dtmail	2	64208.90	6.35	20557.14	10112.43	3.17	0.00	250445824	43280
dtaction	800	47602.28	11.26	15.37	4226.93	0.01	0.73	640057536	8095
soffice.	13	35506.79	0.97	9.23	36510.84	0.07	0.11	134754320	5712
dtwm	2	20350.98	3.17	20557.14	6419.87	1.59	0.00	190636032	14049

월별 명령 요약으로 제공된 데이터에 대한 설명은 [144 페이지](#) “일별 명령 요약”을 참조하십시오.

마지막 로그인 보고서

이 보고서는 특정 로그인이 마지막 사용된 날짜를 제공합니다. 이 정보를 사용하여 사용되지 않은 로그인 및 로그인 디렉토리를 찾아서 아카이브하거나 삭제할 수 있습니다. 마지막 로그인 보고서는 다음과 같습니다.

Jan 16 02:30 2004 LAST LOGIN Page 1

01-06-12	kryten	01-09-08	protoA	01-10-14	ripley
01-07-14	lister	01-09-08	protoB	01-10-15	scutter1
01-08-16	pmorph	01-10-12	rimmer	01-10-16	scutter2

acctcom으로 pacct 파일 검사

언제든지 acctcom 프로그램을 사용하여 /var/adm/pacct *n* 파일 또는 기타 acct.h 형식의 레코드를 가진 파일의 내용을 검사할 수 있습니다. 이 명령을 실행할 때 아무 파일도 지정하지 않고 어떤 표준 입력도 제공하지 않으면 acctcom 명령이 pacct 파일을 읽습니다. acctcom 명령으로 읽은 각 레코드는 종료된 프로세스에 대한 정보를 제공합니다. ps 명령을 실행하여 활성 프로세스를 검사할 수 있습니다.

acctcom 명령의 기본 출력은 다음 정보를 제공합니다.

```
# acctcom
COMMAND
NAME      USER      TTYNAME      START      END          REAL        CPU        MEAN
TIME      TIME      (SECS)      (SECS)      SIZE(K)
#accton   root      ?            02:30:01  02:30:01    0.03        0.01      304.00
turnacct  adm       ?            02:30:01  02:30:01    0.42        0.01      320.00
mv        adm       ?            02:30:01  02:30:01    0.07        0.01      504.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.03        0.01      712.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01      824.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01      912.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01      920.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01     1136.00
utmp_upd  adm       ?            02:30:01  02:30:01    0.01        0.01      576.00
```

```
closewtm  adm      ?           02:30:01 02:30:01    0.10    0.01  664.00
```

다음 목록은 각 필드를 설명합니다.

COMMAND NAME	명령 이름(수퍼유저 권한으로 명령이 실행된 경우 파운드(#) 기호가 붙음)
USER	사용자 이름
TTYNAME	tty 이름(알 수 없으면 ?로 나열됨)
START TIME	명령 실행 시작 시간
END TIME	명령 실행 종료 시간
REAL (SECS)	실제 시간(초)
CPU (SECS)	CPU 시간(초)
MEAN SIZE (K)	평균 크기(킬로바이트)

acctcom 명령 옵션을 사용하여 다음 정보를 얻을 수 있습니다.

- fork/exec 플래그의 상태(exec 없는 fork의 경우 1)
- 시스템 종료 상태
- 호그 팩터
- 총 kcore 분
- CPU 팩터
- 전송된 문자
- 읽은 블록

다음 목록은 acctcom 명령 옵션을 설명합니다.

- a 선택한 프로세스에 대한 평균 통계를 표시합니다. 출력을 기록한 후에 통계가 인쇄됩니다.
- b 가장 최근 명령을 맨 앞에 보여주면서 파일을 역으로 읽습니다. 이 옵션은 표준 입력을 읽을 경우 아무 영향이 없습니다.
- f fork/exec 플래그 및 시스템 종료 상태 열을 인쇄합니다. 출력은 8진수입니다.
- h 평균 메모리 크기 대신, 실행 중 프로세스에서 소비한 총 사용 가능한 CPU 시간의 비율을 나타내는 호그 팩터를 표시합니다. 호그 팩터 = *total-CPU-time/elapsed-time*.
- i 출력에 I/O 수를 포함하는 열을 인쇄합니다.
- k 메모리 크기 대신, 총 kcore minutes를 표시합니다.
- m 평균 코어 크기를 표시합니다. 이 크기가 기본값입니다.
- q 출력 레코드가 아닌 평균 통계를 인쇄합니다.

- r CPU 팩터: $user-time / (system-time + user-time)$ 을 표시합니다.
- t 별도의 시스템 및 사용자 CPU 시간을 표시합니다.
- v 출력에서 열 머리글을 제외합니다.
- C *sec* 총 CPU 시간(시스템+사용자)이 *sec* 초를 초과하는 프로세스만 표시합니다.
- e *time* *hr[:min[:sec]]* 형식으로 제공된 *time* 정시에, 또는 그 이전에 존재하는 프로세스를 표시합니다.
- E *time* *hr[:min[:sec]]* 형식으로 제공된 *time* 정시에, 또는 그 이전에 시작하는 프로세스를 표시합니다. -S와 -E를 동시에 사용하면 그 시간에 존재한 프로세스를 표시합니다.
- g *group* *group*에 속하는 프로세스만 표시합니다.
- H *factor* *factor*를 초과하는 프로세스만 표시합니다. 여기서 *factor*는 "호그 팩터"(-h 옵션 참조)입니다.
- I *chars* *chars*로 지정된 경계 숫자보다 많은 문자를 전송한 프로세스만 표시합니다.
- \ *line* 단말기 /dev/ *line*에 속하는 프로세스만 표시합니다.
- n *pattern* *pattern*("+"가 하나 이상의 발생 수를 의미한다는 점을 제외한 정규 표현식)과 일치하는 명령만 표시합니다.
- o *ofile* 레코드를 인쇄하는 대신, acct.h 형식의 레코드를 *ofile*에 복사합니다.
- O *sec* CPU 시스템 시간이 *sec* 초를 초과하는 프로세스만 표시합니다.
- s *time* *hr[:min[:sec]]* 형식으로 제공된 *time* 정시에, 또는 그 이후에 존재하는 프로세스를 표시합니다.
- S *time* *hr[:min[:sec]]* 형식으로 제공된 *time* 정시에, 또는 그 이전에 시작하는 프로세스를 표시합니다.
- u *user* *user*에 속하는 프로세스만 표시합니다.

시스템 계산 파일

/var/adm 디렉토리에는 활성 데이터 수집 파일이 있습니다. 다음 목록은 이 디렉토리의 계산 파일을 설명합니다.

- dtmp acctdusg 프로그램의 출력
- fee chargefee 프로그램의 출력. ASCII tacct 레코드입니다.
- pacct 활성 프로세스 계산 파일
- pacct n turnacct 스크립트를 실행하여 전환되는 프로세스 계산 파일

Spacctn.MMDD runacct 스크립트 실행 동안 MMDD의 프로세스 계산 파일

/var/adm/acct 디렉토리는 nite, sum, fiscal 디렉토리가 있습니다. 이러한 디렉토리는 실제 데이터 수집 파일이 들어 있습니다. 예를 들어, nite 디렉토리는 runacct 스크립트에 의해 매일 재사용되는 파일이 들어 있습니다. /var/adm/acct/nite 디렉토리의 파일을 간단히 요약하면 다음과 같습니다.

표 10-4 /var/adm/acct/nite 디렉토리의 파일

파일	설명
active	runacct 스크립트에서 진행률을 기록하고 경고 및 오류 메시지를 인쇄하는데 사용됩니다.
active.MMDD	runacct 스크립트가 오류를 감지한 후에 active 파일과 동일합니다.
cms	prdaily 스크립트에서 사용된 ASCII 총 명령 요약입니다.
ctacct.MMDD	tacct.h 형식의 연결 계산 레코드입니다.
ctmp	acctcon1 프로그램의 출력으로, ctmp.h 형식의 연결 세션 레코드로 구성됩니다. acctcon1 및 acctcon2는 호환성 목적으로 제공됩니다.
daycms	prdaily 스크립트에서 사용된 ASCII 일별 명령 요약입니다.
daytacct	tacct.h 형식의, 하루 동안의 총 계산 레코드입니다.
disktacct	dodisk 스크립트로 생성된 tacct.h 형식의 디스크 계산 레코드입니다.
fd2log	runacct 스크립트 실행 동안 진단 출력입니다.
lastdate	runacct 스크립트가 실행된 마지막 날입니다(date +%m%d 형식).
lineuse	prdaily 스크립트에서 사용된 tty 회선 사용량 보고서입니다.
lock	runacct 스크립트의 연속 사용을 제어하는데 사용됩니다.
log	acctcon 프로그램의 진단 출력입니다.
log.MMDD	runacct 스크립트가 오류를 감지한 후에 log 파일과 동일합니다.
owtmpx	전날의 wtmpx 파일입니다.
reboots	wtmpx 파일의 시작 및 종료 날짜와 재부트 목록입니다.
statefile	runacct 스크립트 실행 동안 현재 상태를 기록하는데 사용됩니다.
tmpwtmp	wtmpfix 프로그램에서 수정된 wtmpx 파일입니다.
wtmperror	wtmpfix 오류 메시지를 포함합니다.
wtmperror.MMDD	runacct 스크립트가 오류를 감지한 후에 wtmperror 파일과 동일합니다.
wtmp.MMDD	runacct 스크립트의 wtmpx 파일 복사본입니다.

sum 디렉토리에는 runacct 스크립트로 업데이트되고 monacct 스크립트에서 사용된 누적 요약 파일이 있습니다. 다음 표는 /var/adm/acct/sum 디렉토리의 파일을 요약합니다.

표 10-5 /var/adm/acct/sum 디렉토리의 파일

파일	설명
cms	이진 형식의, 현재 회계 기간 동안의 총 명령 요약입니다.
cmsprev	최신 업데이트 없이 명령 요약 파일입니다.
daycms	내부 요약 형식의, 그날의 사용량에 대한 명령 요약 파일입니다.
loginlog	각 사용자가 로그인한 마지막 날짜의 레코드로, lastlogin 스크립트로 생성되고 prdaily 스크립트에서 사용됩니다.
rprrt.MMDD	prdaily 스크립트의 저장된 출력입니다.
tacct	현재 회계 기간 동안의 누적 총 계산 파일입니다.
tacctprev	최신 업데이트 없이 tacct 파일과 동일합니다.
tacct.MMDD	MMDD에 대한 총 계산 파일입니다.

fiscal 디렉토리에는 monacct 스크립트로 생성된 정기 요약 파일이 있습니다. 다음 표는 /var/adm/acct/fiscal 디렉토리의 파일을 요약합니다.

표 10-6 /var/adm/acct/fiscal 디렉토리의 파일

파일	설명
cmsn	내부 요약 형식의, 회계 기간 n 동안의 총 명령 요약 파일입니다.
fiscrptn	회계 기간 n 동안의 rprtn과 비슷한 보고서입니다.
tacctn	회계 기간 n 동안의 총 계산 파일입니다.

runacct 스크립트로 생성된 파일

다음 표는 runacct 스크립트로 생성된 가장 유용한 파일을 요약합니다. 이러한 파일은 /var/adm/acct 디렉토리에 있습니다.

표 10-7 runacct 스크립트로 생성된 파일

파일	설명
nite/daytacct	tacct.h 형식의, 그날의 총 계산 파일입니다.

표 10-7 runacct 스크립트로 생성된 파일 (계속)

파일	설명
nite/lineuse	runacct 스크립트는 acctcon 프로그램을 호출하여 /var/adm/acct/nite/tmpwtmp 파일에서 단말기 회선 사용량에 대한 데이터를 수집하고 /var/adm/acct/nite/lineuse 파일에 데이터를 기록합니다. prdaily 스크립트는 이 데이터를 사용하여 회선 사용량을 보고합니다. 이 보고서는 특히 잘못된 회선 감지에 유용합니다. 로그아웃 대 로그인 수 비율이 3:1보다 크면 회선에 문제가 있을 가능성이 높습니다.
sum/cms	이 파일은 매일의 명령 요약을 누적한 것입니다. monacct 스크립트가 실행될 때 누적을 다시 시작합니다. ASCII 버전은 nite/cms 파일입니다.
sum/daycms	runacct 스크립트는 acctcms 프로그램을 호출하여 그날 동안 사용된 명령으로 일별 명령 요약 보고서를 만들고 /var/adm/acct/sum/daycms 파일에 데이터를 저장합니다. ASCII 버전은 /var/adm/acct/nite/daycms 파일입니다.
sum/loginlog	runacct 스크립트는 lastlogin 스크립트를 호출하여 /var/adm/acct/sum/loginlog 파일의 로그인에 대해 마지막 로그인 날짜를 업데이트합니다. 또한 lastlogin 명령은 이 파일에서 더 이상 유효하지 않은 로그인을 제거합니다.
sum/rprt.MMDD	runacct 스크립트를 실행할 때마다 prdaily 스크립트로 인쇄된 일별 보고서의 복사본을 저장합니다.
sum/tacct	매일의 nite/daytacct 데이터 누적을 포함하며 요금 청구 목적에 사용됩니다. monacct 스크립트는 각 월마다 또는 회계 기간에 이 데이터 누적을 다시 시작합니다.

시스템 성능 관리(개요)

컴퓨터 또는 네트워크로부터 뛰어난 성능을 얻는 것은 시스템 관리의 중요한 부분입니다. 이 장에서는 사용자가 관리하는 컴퓨터 시스템의 성능을 관리하는 데 기여하는 몇 가지 요소에 대한 개요를 제공합니다.

이 장에서 다루는 개요 정보는 다음과 같습니다.

- 153 페이지 “시스템 성능 관리의 새로운 기능”
- 154 페이지 “시스템 성능 작업의 위치”
- 155 페이지 “시스템 성능 및 시스템 리소스”
- 155 페이지 “프로세스 및 시스템 성능”
- 157 페이지 “시스템 성능 모니터링 정보”

시스템 성능 관리의 새로운 기능

이 절에서는 Oracle Solaris 릴리스에서 시스템 성능을 관리하기 위해 새로 추가되거나 변경된 기능에 대해 설명합니다. 새로운 기능의 전체 목록 및 Oracle Solaris 릴리스에 대한 설명은 [Oracle Solaris 10/8/11 새로운 기능](#)을 참조하십시오.

향상된 pfiles 도구

Oracle Solaris 10: /proc 파일 시스템이 /proc/pic/path 디렉토리에 파일 이름 정보를 포함하도록 향상되었습니다. 이 정보는 pfiles에서 프로세스에 있는 각 파일의 파일 이름을 표시하는 데 사용됩니다. 이 변경 사항은 프로세스 동작에 대한 새로운 시각을 제공합니다. 자세한 내용은 [165 페이지 “프로세스에 대한 정보를 표시하는 방법”](#) 및 [proc\(1\)](#) 매뉴얼 페이지를 참조하십시오.

CPU 성능 카운터

Oracle Solaris 10: CPU 성능 카운터(CPC) 시스템이 Oracle Solaris OS에서 실행되는 SPARC 및 x86 플랫폼에서 사용할 수 있는 성능 분석 기능에 보다 잘 액세스할 수 있도록 업데이트되었습니다.

CPC 명령 `cpustat` 및 `cpustrack`에는 CPU 정보를 지정하는 용도의 고급 명령줄 구문이 있습니다. 예를 들어, 이전 버전의 Oracle Solaris OS에서는 두 개의 카운터를 지정해야 했습니다. 다음 예에 표시된 것과 같이, 이제 두 명령의 구성을 사용하여 한 개의 카운터만 지정할 수 있습니다.

```
# cputrack -c pic0=Cycle_cnt ls -d .
time lwp      event      pic0      pic1
.
0.034  1      exit      841167
```

다음 예에 표시된 것과 같이, 간단한 측정을 위해 카운터 구성을 생략할 수도 있습니다.

```
# cputrack -c Cycle_cnt ls -d .
time lwp      event      pic0      pic1
.
0.016  1      exit      850736
```

`cpustat` 명령 사용에 대한 자세한 내용은 [cpustat\(1M\)](#) 매뉴얼 페이지를 참조하십시오.
`cpustrack` 명령 사용에 대한 자세한 내용은 [cpustrack\(1\)](#) 매뉴얼 페이지를 참조하십시오.

시스템 성능 작업의 위치

시스템 성능 작업	자세한 정보
프로세스 관리	12 장, “시스템 프로세스 관리(작업)”
시스템 성능 모니터링	13 장, “시스템 성능 모니터링(작업)”
조정 가능 매개변수 변경	Oracle Solaris Tunable Parameters Reference Manual
시스템 성능 작업 관리	시스템 관리 설명서: Oracle Solaris Containers-리소스 관리 및 Oracle Solaris 영역의 2 장 , “프로젝트 및 작업(개요)”
FX 및 FS 스케줄러를 사용하여 프로세스 관리	시스템 관리 설명서: Oracle Solaris Containers-리소스 관리 및 Oracle Solaris 영역의 8 장 , “FSS(Fair Share Scheduler)(개요)”

시스템 성능 및 시스템 리소스

컴퓨터 시스템의 성능은 시스템에서 리소스를 사용하고 할당하는 방식에 따라 달라집니다. 정상적인 조건에서 시스템이 어떻게 작동하는지 알 수 있도록 시스템의 성능을 정기적으로 모니터링하십시오. 문제를 예상하고 문제 발생 시 문제를 인식할 수 있어야 합니다.

성능에 영향을 미치는 시스템 리소스는 다음 표에 설명되어 있습니다.

시스템 리소스	설명
중앙 처리 장치(CPU)	CPU는 메모리에서 명령어를 불러온 다음 실행하여 명령어를 처리합니다.
입/출력(I/O) 장치	I/O 장치는 컴퓨터에서 정보를 전송하거나 컴퓨터로 정보를 전송합니다. 터미널 및 키보드, 디스크 드라이브 또는 프린터가 이러한 장치일 수 있습니다.
메모리	물리적(또는 주) 메모리는 시스템의 RAM(Random Access Memory) 양입니다.

13 장, “시스템 성능 모니터링(작업)”에는 시스템의 작업 및 성능에 대한 통계를 표시하는 도구가 설명되어 있습니다.

프로세스 및 시스템 성능

다음 표는 프로세스와 관련된 용어에 대해 설명합니다.

표 11-1 프로세스 용어

용어	설명
프로세스	임의의 시스템 작업입니다. 사용자가 시스템을 부트하거나, 명령을 실행하거나, 응용 프로그램을 시작할 때마다 시스템에서 하나 이상의 프로세스가 활성화됩니다.
LWP(Lightweight Process)	가상 CPU 또는 실행 리소스입니다. LWP는 예약 클래스 및 우선 순위별 기준으로 사용 가능한 CPU 리소스를 사용하도록 커널에 의해 예약됩니다. LWP에는 커널 스레드와 LWP가 포함되어 있습니다. 커널 스레드에는 항상 메모리에 있어야 하는 정보가 포함되어 있습니다. LWP에는 스왑 가능한 정보가 포함되어 있습니다.
응용 프로그램 스레드	사용자의 주소 공간에서 개별적으로 실행할 수 있는 별도의 스택을 포함하는 일련의 명령어입니다. 응용 프로그램 스레드는 LWP를 기반으로 멀티플렉싱될 수 있습니다.

프로세스는 여러 개의 LWP와 여러 개의 응용 프로그램 스레드로 구성될 수 있습니다. 커널은 SunOS 환경의 예약 엔티티인 커널 스레드 구조를 예약합니다. 다양한 프로세스 구조는 다음 표에 설명되어 있습니다.

표 11-2 프로세스 구조

구조	설명
proc	전체 프로세스에 속하는 정보가 포함되어 있으며 항상 주 메모리에 있어야 합니다.
kthread	하나의 LWP에 속하는 정보가 포함되어 있으며 항상 주 메모리에 있어야 합니다.
user	스왑 가능한 "프로세스별" 정보가 포함되어 있습니다.
klwp	스왑 가능한 "LWP 프로세스별" 정보가 포함되어 있습니다.

다음 그림은 이러한 프로세스 구조 간의 관계를 보여줍니다.

그림 11-1 프로세스 구조 간의 관계



대부분의 프로세스 리소스는 프로세스의 모든 스레드에서 액세스할 수 있습니다. 거의 모든 프로세스 가상 메모리는 공유 메모리입니다. 한 스레드에 의해 공유 데이터가 변경되면 프로세스의 다른 스레드에서 이 변경 사항을 사용할 수 있습니다.

시스템 성능 모니터링 정보

컴퓨터가 실행 중인 동안 운영 체제의 카운터는 다양한 시스템 작업을 추적하기 위해 증분됩니다.

추적되는 시스템 작업은 다음과 같습니다.

- 중앙 처리 장치(CPU) 사용량
- 버퍼 사용량
- 디스크 및 테이프 입/출력(I/O) 작업
- 터미널 장치 작업
- 시스템 호출 작업
- 컨텍스트 전환
- 파일 액세스
- 대기열 작업
- 커널 테이블
- 프로세스 간 통신
- 페이징
- 사용 가능한 메모리 및 스왑 공간
- 커널 메모리 할당(KMA)

모니터링 도구

Oracle Solaris 소프트웨어는 시스템이 작업을 수행하는 방식을 추적하는 데 유용한 여러 가지 도구를 제공합니다. 다음 표는 이러한 도구를 보여줍니다.

표 11-3 성능 모니터링 도구

명령	설명	자세한 정보
cpustat 및 cputrack 명령	CPU 성능 카운터를 사용하여 시스템 또는 프로세스의 성능을 모니터링합니다.	cpustat(1M) 및 cputrack(1)
netstat 및 nfsstat 명령	네트워크 성능에 대한 정보를 표시합니다.	netstat(1M) 및 nfsstat(1M)
ps 및 prstat 명령	활성 프로세스에 대한 정보를 표시합니다.	12 장, “시스템 프로세스 관리(작업)”
sar 및 sadc 명령	시스템 작업 데이터를 수집하고 보고합니다.	13 장, “시스템 성능 모니터링(작업)”
Sun Enterprise SyMON	Oracle의 엔터프라이즈 레벨 시스템에 대한 시스템 작업 데이터를 수집합니다.	Sun Enterprise SyMON 2.0.1 Software User's Guide

표 11-3 성능 모니터링 도구 (계속)

명령	설명	자세한 정보
swap 명령	시스템의 사용 가능한 스왑 공간에 대한 정보를 표시합니다.	System Administration Guide: Devices and File Systems 의 19 장, “Configuring Additional Swap Space (Tasks)”
vmstat 및 iostat 명령	가상 메모리 통계, 디스크 사용량 및 CPU 작업 등의 시스템 작업 데이터를 요약합니다.	13 장, “시스템 성능 모니터링(작업)”
cputrack 및 cpustat 명령	마이크로프로세서에서 제공하는 하드웨어 성능 카운터 기능 액세스를 지원합니다.	cputrack(1) 및 cpustat(1M) 매뉴얼 페이지
kstat 및 mpstat 명령	시스템에서 사용 가능한 커널 통계 또는 kstats 를 확인하고, 명령줄에 지정된 기준과 일치하는 통계를 보고합니다. mpstat 명령은 프로세서 통계를 표 형식으로 보고합니다.	kstat(1M) 및 mpstat(1M) 매뉴얼 페이지

시스템 프로세스 관리(작업)

이 장에서는 시스템 프로세스 관리 절차를 설명합니다.

시스템 프로세스 관리와 관련된 절차에 대한 자세한 내용은 다음을 참조하십시오.

- 159 페이지 “시스템 프로세스 관리의 새로운 기능”
- 160 페이지 “시스템 프로세스 관리(작업 맵)”
- 170 페이지 “프로세스 클래스 정보 관리(작업 맵)”

시스템 프로세스 관리에 대한 개요 정보는 다음을 참조하십시오.

- 161 페이지 “시스템 프로세스 관리 명령”
- 171 페이지 “프로세스 클래스 정보 관리”

시스템 프로세스 관리의 새로운 기능

의사 커널 프로세스

Oracle Solaris에는 시스템에서 실행 중인 여러 프로세스가 포함되어 있습니다. 이러한 프로세스는 특정 작업을 수행하지만 관리할 필요가 없습니다. 이러한 프로세스에 대해 설명하는 참조 설명서 문서(매뉴얼 페이지)도 없습니다.

다음 표에서는 이러한 프로세스 각각에 대해 설명합니다.

프로세스	설명
fsflush	페이지를 디스크에 비우는 시스템 데몬입니다.
init	다른 프로세스 및 SMF 구성 요소를 시작 및 재시작하는 초기 시스템 프로세스입니다.

프로세스	설명
intrd	인터럽트로 인한 시스템 로드를 모니터하고 균형을 조정하는 시스템 프로세스입니다.
kmem_task	메모리 캐시 크기를 모니터하는 시스템 프로세스입니다.
pageout	디스크에 대한 메모리 페이징을 제어하는 시스템 프로세스입니다.
sched	OS 예약 및 프로세스 교체를 수행하는 시스템 프로세스입니다.
vm_tasks	성능 향상을 위해 CPU 사이에 가상 메모리 관련 작업 로드의 균형을 조정하고 분산시키는 프로세서당 하나의 스레드를 가지는 시스템 프로세스입니다
zpool- <i>pool-name</i>	연관된 풀에 대한 I/O 작업 스레드를 포함하는 각 ZFS 저장소 풀에 대한 시스템 프로세스입니다.

시스템 프로세스 관리(작업 맵)

작업	설명	수행 방법
프로세스를 나열합니다.	ps 명령을 사용하여 시스템의 모든 프로세스를 나열합니다.	164 페이지 “프로세스를 나열하는 방법”
프로세스에 대한 정보를 표시합니다.	pgrep 명령을 사용하여 추가 정보를 표시하고자 하는 프로세스에 대한 프로세스 ID를 가져옵니다.	165 페이지 “프로세스에 대한 정보를 표시하는 방법”
프로세스를 제어합니다.	pgrep 명령을 사용하여 프로세스를 찾습니다. 그런 다음 적당한 pcommand(/proc)를 사용하여 프로세스를 제어합니다. (/proc) 명령에 대한 설명은 표 12-3을 참조하십시오.	166 페이지 “프로세스를 제어하는 방법”
프로세스를 종료합니다.	프로세스 이름이나 프로세스 ID로 프로세스를 찾습니다. pkill 또는 kill 명령을 사용하여 프로세스를 종료할 수 있습니다.	167 페이지 “프로세스를 종료하는 방법(pkil)” 168 페이지 “프로세스를 종료하는 방법(kill)”

시스템 프로세스 관리 명령

다음 표는 시스템 프로세스 관리 명령을 설명합니다.

표 12-1 프로세스 관리 명령

명령	설명	매뉴얼 페이지
ps, pgrep, prstat, pkill	시스템에서 활성 프로세스의 상태를 확인하고, 프로세스에 대한 자세한 정보를 표시합니다.	ps(1), pgrep(1) 및 prstat(1M)
pkill	pgrep와 동일한 기능을 수행하지만, 이름이나 기타 속성으로 프로세스를 찾거나 신호하고 프로세스를 종료합니다. 일치하는 각 프로세스는 해당 프로세스 ID가 출력되는 대신 kill 명령인 것처럼 신호됩니다.	pgrep(1) 및 pkill(1) kill(1)
pargs, preap	프로세스 디버깅을 돕습니다.	pargs(1) 및 preap(1)
dispadmin	기본 프로세스 예약 정책을 나열합니다.	dispadmin(1M)
priocntl	우선 순위 클래스에 프로세스를 지정하고 프로세스 우선 순위를 관리합니다.	priocntl(1)
nice	시간 공유 프로세스의 우선 순위를 변경합니다.	nice(1)
psrset	특정 프로세스 그룹을 단일 프로세서가 아닌 프로세서 그룹에 바인드합니다.	psrset(1M)

Solaris Management Console의 프로세스 툴을 사용하여 사용자 친숙한 인터페이스로 프로세스를 관리할 수 있습니다. Solaris Management Console 사용 및 시작에 대한 자세한 내용은 [시스템 관리 설명서: 기본 관리의 2 장](#), “Solaris Management Console 작업(작업)”을 참조하십시오.

ps 명령 사용

ps 명령을 사용하여 시스템에서 활성 프로세스의 상태를 확인하고, 프로세스에 대한 기술적 정보를 표시할 수 있습니다. 이 데이터는 프로세스 우선 순위 설정 방법 결과와 같은 관리 작업에 유용합니다.

사용하는 옵션에 따라 ps 명령은 다음 정보를 보고합니다.

- 프로세스의 현재 상태
- 프로세스 ID
- 상위 프로세스 ID
- 사용자 ID
- 예약 클래스
- 우선 순위
- 프로세스의 주소
- 사용된 메모리
- 사용된 CPU 시간

다음 표는 ps 명령으로 보고되는 몇 가지 필드를 설명합니다. 표시되는 옵션은 선택하는 옵션에 따라 달라집니다. 사용 가능한 모든 옵션에 대한 설명은 [ps\(1\)](#) 매뉴얼 페이지를 참조하십시오.

표 12-2 ps 보고서의 필드 요약

필드	설명
UID	프로세스 소유자의 유효 사용자 ID입니다.
PID	프로세스 ID입니다.
PPID	상위 프로세스 ID입니다.
C	예약을 위한 프로세서 사용률입니다. 이 필드는 -c 옵션이 사용되면 표시되지 않습니다.
CLS	실시간, 시스템 또는 시간 공유와 같이 프로세스가 속하는 예약 클래스입니다. 이 필드는 -c 옵션을 사용할 때만 포함됩니다.
PRI	커널 스레드의 예약 우선 순위입니다. 숫자가 높을수록 높은 우선 순위를 나타냅니다.
NI	예약 우선 순위에 영향을 주는 프로세스의 nice 번호입니다. 프로세스가 더 "nice"해지면 우선 순위가 낮음을 의미합니다.
ADDR	proc 구조의 속성입니다.
SZ	프로세스의 가상 주소 크기입니다.
WCHAN	프로세스가 일시 정지 상태인 이벤트 또는 잠금의 주소입니다.
STIME	프로세스의 시작 시간(시, 분, 초)입니다.
TTY	프로세스 또는 상위 프로세스가 시작된 터미널입니다. 물음표는 제어 터미널이 없음을 나타냅니다.
TIME	시작된 이후 프로세스에서 사용한 총 CPU 시간입니다.
CMD	프로세스를 생성한 명령입니다.

/proc 파일 시스템 및 명령 사용

프로세스 명령을 사용하여 /proc 디렉토리에 나열된 프로세스에 대한 자세한 정보를 표시할 수 있습니다. 다음 표는 /proc 프로세스 명령을 나열합니다. 또한 /proc 디렉토리는 프로세스 파일 시스템(PROCFS)이라고 합니다. 활성 프로세스의 이미지는 여기에 해당 프로세스 ID 번호로 저장됩니다.

표 12-3 프로세스 명령(/proc)

프로세스 명령	설명
pcred	프로세스 자격 증명 정보를 표시합니다.
pfiles	프로세스에서 열린 파일에 대한 fstat 및 fcntl 정보를 보고합니다.
pflags	/proc 추적 플래그, 보류 중 신호/보류 신호 및 기타 상태 정보를 출력합니다.
pldd	프로세스에 연결된 동적 라이브러리를 나열합니다.
pmap	각 프로세스의 주소 공간 맵을 출력합니다.
psig	각 프로세스의 신호 작업 및 처리기를 나열합니다.
prun	각 프로세스를 시작합니다.
pstack	각 프로세스의 각 LWP에 대한 16진수+심볼릭 스택 추적을 출력합니다.
pstop	각 프로세스를 중지합니다.
ptime	미시상태 계산을 사용하여 프로세스 시간을 지정합니다.
ptree	프로세스를 포함하는 프로세스 트리를 표시합니다.
pwait	프로세스가 종료된 후 상태 정보를 표시합니다.
pwdx	프로세스에 대한 현재 작업 디렉토리를 표시합니다.

자세한 내용은 [proc\(1\)](#)을 참조하십시오.

프로세스 툴은 ps 명령의 일부 옵션과 유사하지만, 이러한 명령으로 제공되는 출력이 좀더 자세합니다.

일반적으로 프로세스 명령은 다음을 수행합니다.

- fstat 및 fcntl과 같은 프로세스, 작업 디렉토리 및 상위/하위 프로세스의 트리에 대한 추가 정보를 표시합니다.
- 사용자가 프로세스를 중지하거나 재개할 수 있도록 하여 프로세스에 대한 제어를 제공합니다.

프로세스 명령을 사용하여 프로세스 관리(/proc)

몇 가지 프로세스 명령을 사용하여 프로세스에 대한 자세한 기술적 정보를 표시하거나 활성 프로세스를 제어할 수 있습니다. 표 12-3은 일부 /proc 명령을 나열합니다.

프로세스가 무한 루프에 갇히게 되거나 프로세스를 실행하는 데 너무 오래 걸리는 경우 프로세스를 중지(종료)할 수 있습니다. kill 또는 pkill 명령을 사용하여 프로세스를 중지하는 방법에 대한 자세한 내용은 12 장, “시스템 프로세스 관리(작업)”를 참조하십시오.

/proc 파일 시스템은 상태 정보 및 제어 기능에 대한 추가 하위 디렉토리를 포함하는 디렉토리 계층입니다.

또한 /proc 파일 시스템은 프로세스 주소 공간의 개별 페이지에 대한 읽기 및 쓰기 권한을 재매핑하는 데 사용되는 xwatchpoint 기능도 제공합니다. 이 기능에는 제한 사항이 없으며 MT 안전입니다.

디버깅 툴은 /proc의 xwatchpoint 기능을 사용하도록 수정되어 전체 xwatchpoint 프로세스가 더 빨라졌습니다.

dbx 디버깅 툴을 사용하여 xwatchpoint를 설정할 때 다음 제한 사항이 제거되었습니다.

- SPARC 기반 시스템 등록 창으로 인한 스택의 로컬 변수에 대한 xwatchpoint 설정
- 다중 스레드 프로세스에 대한 xwatchpoint 설정

자세한 내용은 proc(4) 및 mdb(1) 매뉴얼 페이지를 참조하십시오.

▼ 프로세스를 나열하는 방법

- ps 명령을 사용하여 시스템의 모든 프로세스를 나열합니다.

```
$ ps [-efc]
```

ps 사용자의 로그인 세션과 연관된 프로세스만 표시합니다.

-ef 시스템에서 실행되는 모든 프로세스에 대한 전체 정보를 표시합니다.

-c 프로세스 스케줄 정보를 표시합니다.

예 12-1 프로세스 나열

다음 예는 옵션이 사용되지 않은 ps 명령의 출력을 보여줍니다.

```
$ ps
  PID TTY          TIME CMD
 1664 pts/4        0:06 csh
 2081 pts/4        0:00 ps
```

다음 예는 `ps -ef` 명령의 출력을 보여줍니다. 이 출력은 시스템이 부트될 때 실행된 첫번째 프로세스가 `sched`(스와퍼), `init` 프로세스, `pageout` 등으로 이어지고 있음을 나타냅니다.

```
$ ps -ef
  UID  PID  PPID  C   STIME TTY          TIME CMD
  root    0    0  0   Dec 20 ?           0:17 sched
  root    1    0  0   Dec 20 ?           0:00 /etc/init -
  root    2    0  0   Dec 20 ?           0:00 pageout
  root    3    0  0   Dec 20 ?           4:20 fsflush
  root   374   367  0   Dec 20 ?           0:00 /usr/lib/saf/ttymon
  root   367    1  0   Dec 20 ?           0:00 /usr/lib/saf/sac -t 300
  root   126    1  0   Dec 20 ?           0:00 /usr/sbin/rpcbind
  root    54    1  0   Dec 20 ?           0:00 /usr/lib/sysevent/syseventd
  root    59    1  0   Dec 20 ?           0:00 /usr/lib/picl/picld
  root   178    1  0   Dec 20 ?           0:03 /usr/lib/autofs/automountd
  root   129    1  0   Dec 20 ?           0:00 /usr/sbin/keyserv
  root   213    1  0   Dec 20 ?           0:00 /usr/lib/lpsched
  root   154    1  0   Dec 20 ?           0:00 /usr/sbin/inetd -s
  root   139    1  0   Dec 20 ?           0:00 /usr/lib/netsvc/yp/ypbind ...
  root   191    1  0   Dec 20 ?           0:00 /usr/sbin/syslogd
  root   208    1  0   Dec 20 ?           0:02 /usr/sbin/nscd
  root   193    1  0   Dec 20 ?           0:00 /usr/sbin/cron
  root   174    1  0   Dec 20 ?           0:00 /usr/lib/nfs/lockd
  daemon 175    1  0   Dec 20 ?           0:00 /usr/lib/nfs/statd
  root   376    1  0   Dec 20 ?           0:00 /usr/lib/ssh/sshd
  root   226    1  0   Dec 20 ?           0:00 /usr/lib/power/powerd
  root   315    1  0   Dec 20 ?           0:00 /usr/lib/nfs/mountd
  root   237    1  0   Dec 20 ?           0:00 /usr/lib/utmpd
  .
  .
  .
```

▼ 프로세스에 대한 정보를 표시하는 방법

- 1 추가 정보를 표시하고자 하는 프로세스의 프로세스 ID를 가져옵니다.

```
# pgrep process
```

여기서 `process`는 추가 정보를 표시하고자 하는 프로세스의 이름입니다.

프로세스 ID는 출력의 첫번째 열에 표시됩니다.

- 2 필요한 프로세스 정보를 표시합니다.

```
# /usr/bin/pcommand pid
```

`pcommand` 실행하고자 하는 (/proc) 명령입니다. 표 12-3은 이러한 명령을 나열하고 설명합니다.

`pid` 프로세스 ID를 식별합니다.

예 12-2 프로세스에 대한 정보 표시

다음 예는 프로세스 명령을 사용하여 cron 프로세스에 대한 추가 정보를 표시하는 방법을 보여줍니다.

```
# pgrep cron      1
4780
# pwdx 4780      2
4780: /var/spool/cron/atjobs
# ptree 4780     3
4780 /usr/sbin/cron
# pfiles 4780    4
4780: /usr/sbin/cron
Current rlimit: 256 file descriptors
0: S_IFCHR mode:0666 dev:290,0 ino:6815752 uid:0 gid:3 rdev:13,2
  O_RDONLY|O_LARGEFILE
  /devices/pseudo/mm@0:null
1: S_IFREG mode:0600 dev:32,128 ino:42054 uid:0 gid:0 size:9771
  O_WRONLY|O_APPEND|O_CREAT|O_LARGEFILE
  /var/cron/log
2: S_IFREG mode:0600 dev:32,128 ino:42054 uid:0 gid:0 size:9771
  O_WRONLY|O_APPEND|O_CREAT|O_LARGEFILE
  /var/cron/log
3: S_IFIFO mode:0600 dev:32,128 ino:42049 uid:0 gid:0 size:0
  O_RDWR|O_LARGEFILE
  /etc/cron.d/FIFO
4: S_IFIFO mode:0000 dev:293,0 ino:4630 uid:0 gid:0 size:0
  O_RDWR|O_NONBLOCK
5: S_IFIFO mode:0000 dev:293,0 ino:4630 uid:0 gid:0 size:0
  O_RDWR
```

1. cron 프로세스에 대한 프로세스 ID를 가져옵니다.
2. cron 프로세스에 대한 현재 작업 디렉토리를 표시합니다.
3. cron 프로세스를 포함하는 프로세스 트리를 표시합니다.
4. fstat 및 fcntl 정보를 표시합니다.

▼ 프로세스를 제어하는 방법

- 1 제어하고자 하는 프로세스의 프로세스 ID를 가져옵니다.

```
# pgrep process
```

여기서 *process*는 제어하고자 하는 프로세스의 이름입니다.

프로세스 ID는 출력의 첫번째 열에 표시됩니다.

2. 적당한 프로세스 명령을 사용하여 프로세스를 제어합니다.

```
# /usr/bin/pcommand pid
```

pcommand 실행하고자 하는 프로세스(/proc) 명령입니다. 표 12-3은 이러한 명령을 나열하고 설명합니다.

pid 프로세스 ID를 식별합니다.

3. 프로세스 상태를 확인합니다.

```
# ps -ef | grep pid
```

예 12-3 프로세스 제어

다음 예는 프로세스 명령을 사용하여 dtpad 프로세스를 중지하고 다시 시작하는 방법을 보여줍니다.

```
# pgrep dtpad    1
2921
# pstop 2921    2
# prun 2921    3
```

1. dtpad 프로세스에 대한 프로세스 ID를 가져옵니다.
2. dtpad 프로세스를 중지합니다.
3. dtpad 프로세스를 다시 시작합니다.

프로세스 종료(pkill, kill)

경우에 따라 프로세스를 중지(종료)해야 할 수 있습니다. 프로세스가 무한 루프에 있을 수 있습니다. 또는 완료되기 전에 중지하려고 하는 큰 작업을 시작했을 수 있습니다. 소유하고 있는 모든 프로세스를 종료할 수 있습니다. 슈퍼유저는 프로세스 ID가 0, 1, 2, 3 및 4인 프로세스를 제외하고 시스템의 모든 프로세스를 종료할 수 있습니다. 이러한 프로세스를 종료하면 시스템에 문제가 발생할 가능성이 큼니다.

자세한 내용은 [pgrep\(1\)](#) 및 [pkill\(1\)](#) 및 [kill\(1\)](#) 매뉴얼 페이지를 참조하십시오.

▼ 프로세스를 종료하는 방법(pkill)

1. (옵션) 다른 사용자의 프로세스를 종료하려면 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
2. 종료하고자 하는 프로세스의 프로세스 ID를 가져옵니다.

```
$ pgrep process
```

여기서 *process*는 종료하고자 하는 프로세스의 이름입니다.

예를 들어, 다음과 같습니다.

```
$ pgrep netscape
587
566
```

프로세스 ID는 출력에 표시됩니다.

주 - Sun Ray에 대한 프로세스 정보를 가져오려면 다음 명령을 사용합니다.

```
# ps -fu user
```

이 명령은 모든 사용자 프로세스를 나열합니다.

```
# ps -fu user | grep process
```

이 명령은 사용자에 대한 특정 프로세스를 찾습니다.

3 프로세스를 종료합니다.

```
$ pkill [signal] process
```

signal pkill 명령줄 구문에 신호가 포함되지 않은 경우 사용되는 기본 신호는 -15(SIGKILL)입니다. pkill 명령과 함께 -9 신호(SIGTERM)를 사용하면 프로세스가 즉시 종료됩니다. 하지만 -9 신호는 데이터베이스 프로세스나 LDAP 서버 프로세스와 같은 특정 프로세스를 종료하는 데 사용하면 안 됩니다. 이 경우 데이터 손실이 발생할 수 있습니다.

process 중지할 프로세스의 이름입니다.

참고 - pkill 명령을 사용하여 프로세스를 종료할 때 처음에는 신호 옵션을 포함시키지 말고 자체 명령만 사용하여 시도해 봅니다. pkill 명령을 -9 신호와 함께 사용하기 전에 몇 분 정도 기다린 후 프로세스가 종료되었는지 확인합니다.

4 프로세스가 종료되었는지 확인합니다.

```
$ pgrep process
```

종료한 프로세스는 pgrep 명령 출력에 더 이상 나열되지 않아야 합니다.

▼ 프로세스를 종료하는 방법(kill)

- 1 (옵션) 다른 사용자의 프로세스를 종료하려면 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

2 종료하고자 하는 프로세스의 프로세스ID를 가져옵니다.

```
$ ps -fu user
```

여기서 *user*는 프로세스를 표시하고자 하는 사용자입니다.

프로세스 ID는 출력의 첫번째 열에 표시됩니다.

3 프로세스를 종료합니다.

```
$ kill [signal-number] pid
```

signal kill 명령줄 구문에 신호가 포함되지 않은 경우 사용되는 기본 신호는 -15(SIGKILL)입니다. kill 명령과 함께 -9 신호(SIGTERM)를 사용하면 프로세스가 즉시 종료됩니다. 하지만 -9 신호는 데이터베이스 프로세스나 LDAP 서버 프로세스와 같은 특정 프로세스를 종료하는 데 사용하면 안 됩니다. 이 경우 데이터 손실이 발생할 수 있습니다.

pid 종료하고자 하는 프로세스의 프로세스 ID입니다.

참고 - kill 명령을 사용하여 프로세스를 중지할 때 처음에는 신호 옵션을 포함시키지 말고 자체 명령만 사용하여 시도해 봅니다. kill 명령을 -9 신호와 함께 사용하기 전에 몇 분 정도 기다린 후 프로세스가 종료되었는지 확인합니다.

4 프로세스가 종료되었는지 확인합니다.

```
$ pgrep pid
```

종료한 프로세스는 pgrep 명령 출력에 더 이상 나열되지 않아야 합니다.

프로세스 디버깅(pargs, preap)

pargs 명령 및 preap 명령은 프로세스 디버깅을 개선합니다. pargs 명령은 활성 프로세스 또는 코어 파일과 연관된 인수 및 환경 변수를 출력합니다. preap 명령은 defunct(좀비) 프로세스를 제거합니다. 좀비 프로세스는 상위 프로세스에서 요청한 종료 상태를 가지지 않습니다. 이러한 프로세스는 일반적으로 해를 끼치지 않지만 수가 많아지면 시스템 리소스를 소모할 수 있습니다. pargs 및 preap 명령을 사용하여 검사할 권한이 있는 프로세스를 검사할 수 있습니다. 슈퍼유저는 모든 프로세스를 검사할 수 있습니다.

preap 명령 사용에 대한 자세한 내용은 [preap\(1\)](#) 매뉴얼 페이지를 참조하십시오. pargs 명령 사용에 대한 자세한 내용은 [pargs\(1\)](#) 매뉴얼 페이지를 참조하십시오. 또한 [proc\(1\)](#) 매뉴얼 페이지를 참조하십시오.

예 12-4 프로세스 디버깅(pargs)

pargs 명령은 ps 명령으로 프로세스에 전달된 모든 인수를 표시할 수 없는 오래 지속되는 문제를 해결합니다. 다음 예는 pgrep 명령과 함께 pargs 명령을 사용하여 프로세스에 전달된 인수를 표시하는 방법을 보여줍니다.

예 12-4 프로세스 디버깅(pargs) (계속)

```
# pargs 'pgrep ttymon'
579: /usr/lib/saf/ttymon -g -h -p system-name console login:
-T sun -d /dev/console -l
argv[0]: /usr/lib/saf/ttymon
argv[1]: -g
argv[2]: -h
argv[3]: -p
argv[4]: system-name console login:
argv[5]: -T
argv[6]: sun
argv[7]: -d
argv[8]: /dev/console
argv[9]: -l
argv[10]: console
argv[11]: -m
argv[12]: ldterm,ttcompat
548: /usr/lib/saf/ttymon
argv[0]: /usr/lib/saf/ttymon
```

다음 예는 pargs -e 명령을 사용하여 프로세스와 연관된 환경 변수를 표시하는 방법을 보여줍니다.

```
$ pargs -e 6763
6763: tcsh
envp[0]: DISPLAY=:0.0
```

프로세스 클래스 정보 관리(작업 맵)

작업	설명	수행 방법
프로세스 클래스에 대한 기본 정보를 표시합니다.	priocntl -l 명령을 사용합니다. 프로세스 예약 클래스 및 우선 순위 범위를 표시합니다.	172 페이지 “프로세스 클래스에 대한 기본 정보를 표시하는 방법(priocntl)”
프로세스의 전역 우선 순위를 표시합니다.	ps -ecl 명령을 사용하여 프로세스의 전역 우선 순위를 표시합니다.	172 페이지 “프로세스의 전역 우선 순위를 표시하는 방법”
프로세스 우선 순위를 지정합니다.	priocntl -e -c 명령을 사용하여 지정된 우선 순위의 프로세스를 시작합니다.	173 페이지 “프로세스 우선 순위를 지정하는 방법(priocntl)”
시간 공유 프로세스의 예약 매개변수를 변경합니다.	priocntl -s -m 명령을 사용하여 시간 공유 프로세스의 예약 매개변수를 변경합니다.	173 페이지 “시간 공유 프로세스의 예약 매개변수를 변경하는 방법(priocntl)”

작업	설명	수행 방법
프로세스의 클래스를 변경합니다.	<code>priocntl -s -c</code> 명령을 사용하여 프로세스의 클래스를 변경합니다.	174 페이지 “프로세스의 클래스를 변경하는 방법(priocntl)”
프로세스의 우선 순위를 변경합니다.	<code>/usr/bin/nice</code> 명령을 적당한 옵션과 함께 사용하여 프로세스의 우선 순위를 낮추거나 높입니다.	176 페이지 “프로세스의 우선 순위를 변경하는 방법(nice)”

프로세스 클래스 정보 관리

다음 목록은 시스템에서 구성할 수 있는 프로세스 예약 클래스를 나타냅니다. 또한 시간 공유 클래스에 대한 사용자가 우선 순위 범위도 포함되어 있습니다.

가능한 프로세스 예약 클래스는 다음과 같습니다.

- 공유(FSS)
- 고정(FX)
- 시스템(SYS)
- 대화식(IA)
- 실시간(RT)
- 시간 공유(TS)
 - 사용자 지정 우선 순위 범위는 -60부터 +60까지입니다.
 - 프로세스의 우선 순위는 상위 프로세스에서 상속됩니다. 이 우선 순위를 **사용자 모드 우선 순위**라고 합니다.
 - 시스템은 시간 공유 전달 매개변수 테이블에서 사용자 모드 우선 순위를 조회합니다. 그런 다음 시스템은 nice 또는 priocntl(사용자 지정) 우선 순위를 추가하고 0-59 범위가 **전역 우선 순위**를 생성하도록 합니다.

프로세스의 예약 우선 순위 변경(priocntl)

프로세스의 예약 우선 순위는 예약 정책에 따라 프로세스 스케줄러에서 지정한 우선 순위입니다. `dispadm` 명령은 기본 예약 정책을 나열합니다. 자세한 내용은 [dispadm\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

`priocntl` 명령을 사용하여 프로세스를 우선 순위 클래스에 지정하고 프로세스 우선 순위를 관리할 수 있습니다. `priocntl` 명령을 사용하여 프로세스를 관리하는 방법에 대한 자세한 내용은 173 페이지 “프로세스 우선 순위를 지정하는 방법(priocntl)”을 참조하십시오.

▼ 프로세스 클래스에 대한 기본 정보를 표시하는 방법(priocntl)

- `priocntl -l` 명령을 사용하여 프로세스 예약 클래스 및 우선 순위 범위를 표시합니다.
\$ `priocntl -l`

예 12-5 프로세스 클래스에 대한 기본 정보 표시(priocntl)

다음 예는 `priocntl -l` 명령의 출력을 보여줍니다.

```
# priocntl -l
CONFIGURED CLASSES
=====

SYS (System Class)

TS (Time Sharing)
    Configured TS User Priority Range: -60 through 60

FX (Fixed priority)
    Configured FX User Priority Range: 0 through 60

IA (Interactive)
    Configured IA User Priority Range: -60 through 60
```

▼ 프로세스의 전역 우선 순위를 표시하는 방법

- `ps` 명령을 사용하여 프로세스의 전역 우선 순위를 표시합니다.
\$ `ps -ecl`
전역 우선 순위는 PRI 열에 나열됩니다.

예 12-6 프로세스의 전역 우선 순위 표시

다음 예는 `ps -ecl` 명령 출력을 보여줍니다. PRI 열의 값은 `pageout` 프로세스의 우선 순위가 가장 높고, `sh` 프로세스의 우선 순위가 가장 낮음을 나타냅니다.

```
$ ps -ecl
 F S  UID  PID  PPID  CLS  PRI  ADDR      SZ  WCHAN    TTY    TIME  CMD
19 T  0    0    0    SYS  96   f00d05a8  0   ?        ?    0:03  sched
 8 S  0    1    0    TS   50   ff0f4678 185  ff0f4848 ?    36:51  init
19 S  0    2    0    SYS  98   ff0f4018  0   f00c645c ?    0:01  pageout
19 S  0    3    0    SYS  60   ff0f5998  0   f00d0c68 ?    241:01 fsflush
 8 S  0   269    1    TS   58   ff0f5338 303  ff49837e ?    0:07   sac
 8 S  0   204    1    TS   43   ff2f6008  50   ff2f606e console 0:02   sh
```

▼ 프로세스 우선 순위를 지정하는 방법(priocntl)

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 지정된 우선 순위의 프로세스를 시작합니다.

```
# priocntl -e -c class -m user-limit -p pri command-name
```

-e 명령을 실행합니다.

-c class 프로세스를 실행할 클래스를 지정합니다. 유효한 클래스는 TS(시간 공유), RT(실시간), IA(대화식), FSS(공유) 및 FX(고정 우선 순위)입니다.

-m user-limit -p 옵션을 사용할 때 우선 순위를 높이거나 낮출 수 있는 최대 양을 지정합니다,

-p pri command-name 실시간 스프레드에 대해 RT 클래스의 상대 우선 순위를 지정할 수 있습니다. 시간 공유 프로세스의 경우 -p 옵션을 사용하여 -60부터 +60까지 사용자 지정 우선 순위를 지정할 수 있습니다.

- 3 프로세스 상태를 확인합니다.

```
# ps -ecl | grep command-name
```

예 12-7 프로세스 우선 순위 지정(priocntl)

다음 예는 가능한 가장 높은 사용자 지정 우선 순위로 find 명령을 시작하는 방법을 보여줍니다.

```
# priocntl -e -c TS -m 60 -p 60 find . -name core -print
# ps -ecl | grep find
```

▼ 시간 공유 프로세스의 예약 매개변수를 변경하는 방법(priocntl)

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 실행 중인 시간 공유 프로세스의 예약 매개변수를 변경합니다.

```
# priocntl -s -m user-limit [-p user-priority] -i idtype idlist
```

- s 사용자 우선 순위 범위에 대한 상한을 설정하고 현재 우선 순위를 변경할 수 있습니다.
- m *user-limit* -p 옵션을 사용할 때 우선 순위를 높이거나 낮출 수 있는 최대 양을 지정합니다.
- p *user-priority* 우선 순위를 지정할 수 있습니다.
- i *xidtype xidlist* *xidtype* 및 *xidlist*의 조합을 사용하여 프로세스를 식별합니다. *xidtype*은 프로세스 ID 또는 사용자 ID와 같은 ID의 유형을 지정합니다. *xidlist*를 사용하여 프로세스 ID 또는 사용자 ID의 목록을 식별합니다.

3 프로세스 상태를 확인합니다.

```
# ps -ecl | grep idlist
```

예 12-8 시간 공유 프로세스의 예약 매개변수 변경(priocntl)

다음 예는 500밀리초 간격, RT 클래스의 우선 순위 20 및 전역 우선 순위 120으로 명령을 실행하는 방법을 보여줍니다.

```
# priocntl -e -c RT -m 500 -p 20 myprog
# ps -ecl | grep myprog
```

▼ 프로세스의 클래스를 변경하는 방법(priocntl)

1 (옵션) 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

2 프로세스의 클래스를 변경합니다.

```
# priocntl -s -c class -i idtype idlist
```

- s 사용자 우선 순위 범위에 대한 상한을 설정하고 현재 우선 순위를 변경할 수 있습니다.
- c *class* 프로세스를 변경하는 클래스(시간 공유의 경우 TS 또는 실시간의 경우 RT)를 지정합니다.
- i *idtype idlist* *xidtype* 및 *xidlist*의 조합을 사용하여 프로세스를 식별합니다. *xidtype*은 프로세스 ID 또는 사용자 ID와 같은 ID의 유형을 지정합니다. *xidlist*를 사용하여 프로세스 ID 또는 사용자 ID의 목록을 식별합니다.

주 - 실시간 프로세스에서 또는 실시간 프로세스로 프로세스를 변경하려면 슈퍼유저이거나 실시간 셸에서 작업 중이어야 합니다. 슈퍼유저가 사용자 프로세스를 실시간 클래스로 변경할 경우 이후 사용자는 `priocntl -s` 명령을 사용하여 실시간 예약 매개변수를 변경할 수 없습니다.

3 프로세스 상태를 확인합니다.

```
# ps -ecl | grep idlist
```

예 12-9 프로세스의 클래스 변경(priocntl)

다음 예는 사용자 15249에 속한 모든 프로세스를 실시간 프로세스로 변경하는 방법을 보여줍니다.

```
# priocntl -s -c RT -i uid 15249
# ps -ecl | grep 15249
```

시간 공유 프로세스의 우선 순위 변경(nice)

`nice` 명령은 이전 Solaris 릴리스에 대한 역호환성을 위해서만 지원됩니다. `priocntl` 명령이 프로세스 관리에서 더욱 많은 유연성을 제공합니다.

프로세스의 우선 순위는 예약 클래스의 정책 및 해당 `nice` 번호로 결정됩니다. 각 시간 공유 프로세스에는 전역 우선 순위가 있습니다. 전역 우선 순위는 `nice` 또는 `priocntl` 명령의 영향을 받을 수 있는 사용자 지정 우선 순위와 시스템 계산 우선 순위를 더하여 계산됩니다.

프로세스의 실행 우선 순위 번호는 운영 체제에서 지정합니다. 우선 순위 번호는 프로세스의 예약 클래스, 사용한 CPU 시간 및 시간 공유 프로세스의 경우 해당 `nice` 번호를 포함한 여러 요소로 결정됩니다.

각 시간 공유 프로세스는 상위 프로세스에서 상속된 기본 `nice` 번호로 시작됩니다. `nice` 번호는 `ps` 보고서의 `NI` 열에 표시됩니다.

사용자는 사용자 지정 우선 순위를 높여 프로세스의 우선 순위를 낮출 수 있습니다. 하지만 슈퍼유저만 `nice` 번호를 낮추어 프로세스의 우선 순위를 높일 수 있습니다. 이 제한 사항은 사용자가 자신의 프로세스 우선 순위를 높여 CPU를 독점하지 못하도록 합니다.

`nice` 번호의 범위는 0부터 +39까지이며, 0이 가장 높은 우선 순위를 나타냅니다. 각 시간 공유 프로세스에 대한 `nice` 기본값은 20입니다. 표준 버전인 `/usr/bin/nice`와 C 셸 내장 명령의 두 가지 버전의 명령을 사용할 수 있습니다.

▼ 프로세스의 우선 순위를 변경하는 방법(nice)

이 절차에 따라 사용자가 프로세스의 우선 순위를 낮출 수 있습니다. 하지만 슈퍼유저는 프로세스의 우선 순위를 높이거나 낮출 수 있습니다.

주 - 이 섹션에서는 C 셸 내장 nice 명령이 아닌 /usr/bin/nice 명령의 구문을 설명합니다. C 셸 nice 명령에 대한 자세한 내용은 [csh\(1\)](#) 매뉴얼 페이지를 참조하십시오.

- 1 프로세스의 우선 순위를 사용자 또는 슈퍼유저로 변경할지 결정합니다. 그런 다음, 다음 중 하나를 선택합니다.

- 사용자는 단계 2의 예를 따라 명령의 우선 순위를 낮춥니다.
- 슈퍼유저는 단계 3의 예를 따라 명령의 우선 순위를 높이거나 낮춥니다.

- 2 사용자는 nice 번호를 높여 명령의 우선 순위를 낮춥니다.

다음 nice 명령은 nice 번호를 5단위씩 높여 낮은 우선 순위의 *command-name* 을 실행합니다.

```
$ /usr/bin/nice -5 command-name
```

위의 명령에서 마이너스 기호는 뒤따르는 옵션을 지정합니다. 이 명령은 다음과 같이 지정할 수도 있습니다.

```
% /usr/bin/nice -n 5 command-name
```

다음 nice 명령은 nice 번호를 기본 증분값인 10단위씩 높이지만 최대값인 39를 초과하지 않도록 하여 *command-name*의 우선 순위를 낮춥니다.

```
% /usr/bin/nice command-name
```

- 3 슈퍼유저 또는 동등 역할의 사용자는 nice 번호를 변경하여 명령의 우선 순위를 높이거나 낮출 수 있습니다.

다음 nice 명령은 nice 번호를 10단위씩 낮추지만 최소값인 0 미만이 되지 않도록 하여 *command-name*의 우선 순위를 높입니다.

```
# /usr/bin/nice --10 command-name
```

위의 명령에서 첫번째 마이너스 기호는 뒤따르는 옵션을 지정합니다. 두번째 마이너스 기호는 음수를 나타냅니다.

다음 nice 명령은 nice 번호를 5단위씩 높이지만 최대값인 39를 초과하지 않도록 하여 *command-name*의 우선 순위를 낮춥니다.

```
# /usr/bin/nice -5 command-name
```

참조 자세한 내용은 [nice\(1\)](#) 매뉴얼 페이지를 참조하십시오.

시스템 프로세스의 문제 해결

다음은 일반적으로 발생할 수 있는 문제에 대한 몇 가지 유용한 정보입니다.

- 같은 사용자가 소유하고 있는 여러 동일 작업을 찾습니다. 이 문제는 작업이 완료될 때까지 기다리지 않고 많은 백그라운드 작업을 시작하는 실행 스크립트로 인해 발생할 수 있습니다.
- 많은 양의 CPU 시간이 누적된 프로세스를 찾습니다. `ps` 출력에서 `TIME` 필드를 확인하여 이 문제를 파악할 수 있습니다. 프로세스가 무한 루프에 있을 수 있습니다.
- 너무 높은 우선 순위로 실행되는 프로세스를 찾습니다. `ps -c` 명령을 사용하여 각 프로세스의 예약 클래스를 표시하는 `CLS` 필드를 확인합니다. 실시간(RT) 프로세스로 실행되는 프로세스는 CPU를 독점할 수 있습니다. 또는 높은 `nice` 번호의 시간 공유(TS) 프로세스를 찾습니다. 슈퍼유저 권한을 가진 사용자가 프로세스의 우선 순위를 높였을 수 있습니다. 시스템 관리자는 `nice` 명령을 사용하여 우선 순위를 낮출 수 있습니다.
- runaway 프로세스를 찾습니다. runaway 프로세스는 갈수록 더 많은 CPU 시간을 사용합니다. 프로세스가 시작된 시간(`STIME`)을 확인하고 잠시 동안 CPU 시간 누적(`TIME`)을 살펴 이 문제를 파악할 수 있습니다.

시스템 성능 모니터링(작업)

이 장에서는 `vmstat`, `iostat`, `df` 및 `sar` 명령을 사용하여 시스템 성능을 모니터링하기 위한 절차를 설명합니다.

시스템 성능 모니터링과 관련된 절차에 대한 자세한 내용은 다음을 참조하십시오.

- 179 페이지 “시스템 성능 정보 표시(작업 맵)”
- 187 페이지 “시스템 작업 모니터링(작업 맵)”

시스템 성능 정보 표시(작업 맵)

작업	설명	수행 방법
가상 메모리 통계를 표시합니다.	<code>vmstat</code> 명령을 사용하여 가상 메모리 통계를 수집합니다.	181 페이지 “가상 메모리 통계를 표시하는 방법(<code>vmstat</code>)”
시스템 이벤트 정보를 표시합니다.	<code>vmstat</code> 명령을 <code>-s</code> 옵션과 함께 사용하여 시스템 이벤트 정보를 표시합니다.	182 페이지 “시스템 이벤트 정보를 표시하는 방법(<code>vmstat -s</code>)”
스왑핑 통계를 표시합니다.	<code>vmstat</code> 명령을 <code>-S</code> 옵션과 함께 사용하여 스왑핑 통계를 표시합니다.	182 페이지 “스왑핑 통계를 표시하는 방법(<code>vmstat -S</code>)”
장치당 인터럽트를 표시합니다.	<code>vmstat</code> 명령을 <code>-i</code> 옵션과 함께 사용하여 장치당 인터럽트의 수를 표시합니다.	183 페이지 “장치당 인터럽트를 표시하는 방법(<code>vmstat -i</code>)”
디스크 사용률을 표시합니다.	<code>iostat</code> 명령을 사용하여 디스크 입출력 통계를 보고합니다.	183 페이지 “디스크 사용률 정보를 표시하는 방법(<code>iostat</code>)”
확장 디스크 통계를 표시합니다.	<code>iostat</code> 명령을 <code>-xtc</code> 옵션과 함께 사용하여 확장 디스크 통계를 표시합니다.	185 페이지 “확장 디스크 통계를 표시하는 방법(<code>iostat -xtc</code>)”

작업	설명	수행 방법
디스크 공간 정보를 표시합니다.	df -k 명령은 디스크 공간 정보를 킬로바이트 단위로 표시합니다.	186 페이지 “디스크 공간 정보를 표시하는 방법(df -k)”

가상 메모리 통계 표시(vmstat)

vmstat 명령을 사용하여 CPU 로드, 페이징, 컨텍스트 전환 수, 장치 인터럽트 및 시스템 호출과 같은 시스템 이벤트에 대한 가상 메모리 통계 및 정보를 보고할 수 있습니다. vmstat 명령은 스와핑, 캐시 비우기 및 인터럽트에 대한 통계도 표시할 수 있습니다.

다음 표는 vmstat 명령 출력의 필드를 설명합니다.

표 13-1 vmstat 명령의 출력

범주	필드 이름	설명
procs		다음에 대해 보고합니다.
	r	전달 대기열의 커널 스레드 수
	b	리소스를 대기 중인 차단된 커널 스레드 수
	w	처리 중인 리소스 완료를 대기 중인 스왑 아웃된 LWP 수
memory		실제 메모리 및 가상 메모리의 사용에 대해 보고합니다.
	swap	사용 가능한 스왑 공간
	free	해제 목록의 크기
page		페이지 폴트 및 페이지 작업을 초 단위로 보고합니다.
	re	재확보된 페이지
	mf	작은 결함 및 큰 결함
	pi	페이징인(킬로바이트)
	po	페이징아웃(킬로바이트)
	fr	해제됨(킬로바이트)
	de	최근 스왑 인된 프로세스에서 필요한 예상 메모리
	sr	현재 사용되고 있지 않은 page 데몬으로 스캔된 페이지. sr이 0이 아니면 page 데몬이 실행된 것입니다.
disk	최대 4개 디스크의 데이터에 대한 초당 디스크 작업 수를 보고합니다.	
faults	초당 트랩/인터럽트 비율을 보고합니다.	

표 13-1 vmstat 명령의 출력 (계속)

범주	필드 이름	설명
	in	초당 인터럽트
	sy	초당 시스템 호출
	cs	CPU 컨텍스트 전환 비율
cpu		CPU 시간 사용을 보고합니다.
	us	사용자 시간
	sy	시스템 시간
	id	유휴 시간

이 명령에 대한 자세한 내용은 `vmstat(1M)` 매뉴얼 페이지를 참조하십시오.

▼ 가상 메모리 통계를 표시하는 방법(vmstat)

- `vmstat` 명령을 시간 간격(초)과 함께 사용하여 가상 메모리 통계를 수집합니다.

```
$ vmstat n
```

여기서 *n*은 보고 사이의 간격(초)입니다.

예 13-1 가상 메모리 통계 표시

다음 예는 5초 간격으로 수집된 통계의 `vmstat` 표시를 보여줍니다.

```
$ vmstat 5
kthr      memory          page        disk        faults       cpu
r  b  w   swap  free re  mf  pi  po  fr  de  sr  dd  f0  s1  --   in  sy   cs  us  sy  id
0  0  0  863160 365680 0   3  1  0  0  0  0  0  0  0  0  406 378 209  1  0 99
0  0  0  765640 208568 0  36  0  0  0  0  0  0  0  0  0  479 4445 1378  3  3 94
0  0  0  765640 208568 0   0  0  0  0  0  0  0  0  0  0  423  214  235  0  0 100
0  0  0  765712 208640 0   0  0  0  0  0  0  0  3  0  0  0  412  158  181  0  0 100
0  0  0  765832 208760 0   0  0  0  0  0  0  0  0  0  0  402  157  179  0  0 100
0  0  0  765832 208760 0   0  0  0  0  0  0  0  0  0  0  403  153  182  0  0 100
0  0  0  765832 208760 0   0  0  0  0  0  0  0  0  0  0  402  168  177  0  0 100
0  0  0  765832 208760 0   0  0  0  0  0  0  0  0  0  0  402  153  178  0  0 100
0  0  0  765832 208760 0  18  0  0  0  0  0  0  0  0  0  407  165  186  0  0 100
```


주 - vmstat 명령은 si 및 so 필드의 출력을 자릅니다. 스왑 통계의 좀더 정확한 계산을 표시하려면 sar 명령을 사용합니다.

▼ 장치당 인터럽트를 표시하는 방법(vmstat -i)

- vmstat -i 명령을 실행하여 장치당 인터럽트 수를 표시합니다.

예 13-2 장치당 인터럽트 표시

다음 예는 vmstat -i 명령의 출력을 보여줍니다.

```
$ vmstat -i
interrupt          total          rate
-----
clock              52163269      100
esp0               2600077       4
zsc0               25341         0
zsc1               48917         0
cgsixc0            459           0
lec0               400882        0
fdc0               14            0
bppc0              0             0
audiocs0           0             0
-----
Total              55238959      105
```

디스크 사용률 정보 표시(iostat)

iostat 명령을 사용하여 디스크 입출력에 대한 통계를 보고하고 처리량, 사용률, 대기열 길이, 트랜잭션 비율 및 서비스 시간에 대한 측정 결과를 표시합니다. 이 명령에 대한 자세한 내용은 [iostat\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

▼ 디스크 사용률 정보를 표시하는 방법(iostat)

- iostat 명령을 시간 간격(초)과 함께 사용하여 디스크 사용률 정보를 표시할 수 있습니다.

```
$ iostat 5
      tty          fd0          sd3          nfs1          nfs31          cpu
tin tout kps tps serv  kps tps serv  kps tps serv  kps tps serv  us sy wt id
  0   1   0   0 410    3   0  29    0   0   9    3   0  47   4  2  0 94
```

출력의 첫 라인은 시스템이 마지막으로 부트된 이후의 통계를 보여줍니다. 이후 각 라인은 간격 통계를 보여줍니다. 기본값은 터미널(tty), 디스크(fd 및 sd) 및 CPU(cpu)에 대한 통계를 보여주는 것입니다.

예 13-3 디스크 사용률 정보 표시

다음 예는 5초마다 수집된 디스크 통계를 보여줍니다.

```
$ iostat 5
tty          sd0             sd6             nfs1            nfs49           cpu
tin tout    kps tps  serv    kps tps  serv    kps tps  serv    kps tps  serv    us sy wt id
0    0    1  0  49    0  0  0    0  0  0    0  0  15    0  0  0 100
0    47  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  44  6  132  0  0  0    0  0  0    0  0  0    0  0  1  99
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  3  1  23  0  0  0    0  0  0    0  0  0    0  0  1  99
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
0    16  0  0  0    0  0  0    0  0  0    0  0  0    0  0  0 100
```

다음 표는 iostat n 명령 출력의 필드를 설명합니다.

장치 유형	필드 이름	설명
터미널		
	tin	터미널 입력 대기열의 문자 수
	tout	터미널 출력 대기열의 문자 수
디스크		
	bps	초당 블록
	tps	초당 트랜잭션
	serv	평균 서비스 시간(밀리초)
CPU		
	us	사용자 모드
	sy	시스템 모드
	wt	I/O 대기 중
	id	유휴

▼ 확장 디스크 통계를 표시하는 방법(iostat -xtc)

- `iostat -xtc` 명령을 실행하여 확장 디스크 통계를 표시합니다.

```
$ iostat -xtc
                extended device statistics
device         r/s      w/s    kr/s    kw/s wait actv  svc_t  %w  %b   tty          cpu
               tin tout   us sy wt id
fd0            0.0     0.0    0.0    0.0  0.0  0.0    0.0  0  0    0    0    0  0 100
sd0            0.0     0.0    0.4    0.4  0.0  0.0   49.5  0  0
sd6            0.0     0.0    0.0    0.0  0.0  0.0    0.0  0  0
nfs1           0.0     0.0    0.0    0.0  0.0  0.0    0.0  0  0
nfs49          0.0     0.0    0.0    0.0  0.0  0.0   15.1  0  0
nfs53          0.0     0.0    0.4    0.0  0.0  0.0   24.5  0  0
nfs54          0.0     0.0    0.0    0.0  0.0  0.0    6.3  0  0
nfs55          0.0     0.0    0.0    0.0  0.0  0.0    4.9  0  0
```

`iostat -xtc` 명령은 각 디스크에 대해 한 라인에 출력을 표시합니다. 출력 필드는 다음 목록에 설명되어 있습니다.

r/s	초당 읽기
w/s	초당 쓰기
kr/s	초당 읽은 킬로바이트
kw/s	초당 쓴 킬로바이트
wait	서비스를 대기 중인 평균 트랜잭션 수(대기열 길이)
actv	서비스 중인 평균 활성 트랜잭션 수
svc_t	평균 서비스 시간(밀리초)
%w	대기열이 비어 있지 않은 시간(백분율)
%b	디스크를 사용 중인 시간(백분율)

디스크 공간 통계 표시(df)

`df` 명령을 사용하여 각 마운트된 디스크에서 사용 가능한 디스크 공간을 표시합니다. 보고 통계에서는 총 사용 가능 공간보다 10%를 허용하므로 `df`로 보고되는 **사용 가능** 디스크 공간에는 전체 용량의 90%만 반영됩니다. 이 **헤드 공간**은 일반적으로 보다 좋은 성능을 위해 비워 둡니다.

`df` 명령으로 실제로 보고되는 디스크 공간의 비율은 사용 가능 공간으로 나눈 사용된 공간입니다.

파일 시스템이 90% 용량을 초과할 경우 `cp` 명령을 사용하여 가득 차지 않은 디스크로 파일을 전송할 수 있습니다. `tar` 또는 `cpio` 명령을 사용하여 테이프로 파일을 전송할 수도 있습니다. 또는 파일을 제거할 수 있습니다.

이 명령에 대한 자세한 내용은 [df\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

▼ 디스크 공간 정보를 표시하는 방법(df -k)

- **df -k** 명령을 사용하여 디스크 공간 정보를 킬로바이트 단위로 표시합니다.

```
$ df -k
Filesystem      kbytes  used  avail capacity  Mounted on
/dev/dsk/c0t3d0s0 192807 40231 133296    24%      /
```

예 13-4 파일 시스템 정보 표시

다음 예는 df -k 명령의 출력을 보여줍니다.

```
$ df -k
Filesystem      kbytes  used  avail capacity  Mounted on
/dev/dsk/c0t0d0s0 254966 204319 25151    90%      /
/devices          0         0         0         0%      /devices
ctfs              0         0         0         0%      /system/contract
proc             0         0         0         0%      /proc
mnttab           0         0         0         0%      /etc/mnttab
swap            496808     376 496432     1%      /etc/svc/volatile
objfs            0         0         0         0%      /system/object
/dev/dsk/c0t0d0s6 3325302 3073415 218634    94%      /usr
fd               0         0         0         0%      /dev/fd
swap            496472     40 496432     1%      /var/run
swap            496472     40 496432     1%      /tmp
/dev/dsk/c0t0d0s5 13702    1745 10587    15%      /opt
/dev/dsk/c0t0d0s7 9450     1045 7460     13%      /export/home
```

다음 표는 df -k 명령의 출력을 설명합니다.

필드 이름	설명
kbytes	파일 시스템에서 총 사용 가능한 공간 크기
used	사용된 공간의 양
avail	사용 가능한 공간의 양
capacity	사용된 공간의 양(총 용량의 백분율)
mounted on	마운트 지점

시스템 작업 모니터링(작업 맵)

작업	설명	수행 방법
파일 액세스를 확인합니다.	sar 명령을 -a 옵션과 함께 사용하여 파일 액세스 작업 상태를 표시합니다.	188 페이지 “파일 액세스를 확인하는 방법(sar -a)”
버퍼 작업을 확인합니다.	sar 명령을 -b 옵션과 함께 사용하여 버퍼 작업 통계를 표시합니다.	189 페이지 “버퍼 작업을 확인하는 방법(sar -b)”
시스템 호출 통계를 확인합니다.	sar 명령을 -c 옵션과 함께 사용하여 시스템 호출 통계를 표시합니다.	191 페이지 “시스템 호출 통계를 확인하는 방법(sar -c)”
디스크 작업을 확인합니다.	sar 명령을 -d 옵션과 함께 사용하여 디스크 작업을 확인합니다.	192 페이지 “디스크 작업을 확인하는 방법(sar -d)”
페이지 아웃 및 메모리를 확인합니다.	sar 명령을 -g 옵션과 함께 사용하여 페이지 아웃 메모리 해제 작업을 표시합니다.	193 페이지 “페이지 아웃 및 메모리를 확인하는 방법(sar -g)”
커널 메모리 할당을 확인합니다.	커널 메모리 할당(KMA)을 통해 커널 부속 시스템에서 필요에 따라 메모리를 할당하고 해제할 수 있습니다. sar 명령을 -k 옵션과 함께 사용하여 KMA를 확인합니다.	195 페이지 “커널 메모리 할당을 확인하는 방법(sar -k)”
프로세스간 통신을 확인합니다.	sar 명령을 -m 옵션과 함께 사용하여 프로세스간 통신 작업을 보고합니다.	196 페이지 “프로세스간 통신을 확인하는 방법(sar -m)”
페이지 인 작업을 확인합니다.	sar 명령을 -p 옵션과 함께 사용하여 페이지 인 작업을 보고합니다.	197 페이지 “페이지 인 작업을 확인하는 방법(sar -p)”
대기열 작업을 확인합니다.	sar 명령을 -q 옵션과 함께 사용하여 다음을 확인합니다. <ul style="list-style-type: none"> ■ 대기열이 채워진 동안의 평균 대기열 길이 ■ 대기열이 채워진 시간(백분율) 	198 페이지 “대기열 작업을 확인하는 방법(sar -q)”
사용되지 않은 메모리를 확인합니다.	sar 명령을 -r 옵션과 함께 사용하여 현재 사용되고 있는 메모리 페이지 및 스왑 파일 디스크 블록 수를 보고합니다.	199 페이지 “사용되지 않은 메모리를 확인하는 방법(sar -r)”
CPU 사용률을 확인합니다.	sar 명령을 -u 옵션과 함께 사용하여 CPU 사용률 통계를 표시합니다.	200 페이지 “CPU 사용률을 확인하는 방법(sar -u)”

작업	설명	수행 방법
시스템 테이블 상태를 확인합니다.	<p>sar 명령을 -v 옵션과 함께 사용하여 다음 시스템 테이블에 대한 상태를 보고합니다.</p> <ul style="list-style-type: none"> ■ 프로세스 ■ Inode ■ 파일 ■ 공유 메모리 레코드 	201 페이지 “시스템 테이블 상태를 확인하는 방법(sar -v)”
스와핑 작업을 확인합니다.	sar 명령을 -w 옵션과 함께 사용하여 스와핑 작업을 확인합니다.	202 페이지 “스와핑 작업을 확인하는 방법(sar -w)”
터미널 작업을 확인합니다.	sar 명령을 -y 옵션과 함께 사용하여 터미널 장치 작업을 모니터링합니다.	203 페이지 “터미널 작업을 확인하는 방법(sar -y)”
전체 시스템 성능을 확인합니다.	sar -A 명령은 모든 옵션의 통계를 표시하여 전체 시스템 성능 정보를 제공합니다.	204 페이지 “전체 시스템 성능을 확인하는 방법(sar -A)”
자동 데이터 수집을 설정합니다.	<p>시스템에서 자동으로 데이터를 수집하도록 설정하고 sar 명령을 실행하려면 다음을 수행합니다.</p> <ul style="list-style-type: none"> ■ svcadm enable system/sar:default 명령을 실행합니다. ■ /var/spool/cron/crontabs/sys 파일을 편집합니다. 	207 페이지 “자동 데이터 수집을 설정하는 방법”

시스템 작업 모니터링(sar)

sar 명령을 사용하여 다음 작업을 수행합니다.

- 시스템 작업에 대한 데이터를 구성하고 확인합니다.
- 특별 요청에 따라 시스템 작업 데이터에 액세스합니다.
- 자동 보고서를 생성하여 시스템 성능을 측정 및 모니터링하고, 특별 요청 보고서를 생성하여 특정 성능 문제를 파악합니다. sar 명령이 시스템에서 실행되도록 설정하는 방법 및 이러한 툴에 대한 자세한 내용은 204 페이지 “자동으로 시스템 작업 데이터 수집(sar)”을 참조하십시오.

이 명령에 대한 자세한 내용은 sar(1) 매뉴얼 페이지를 참조하십시오.

▼ 파일 액세스를 확인하는 방법(sar -a)

- sar -a 명령을 사용하여 파일 액세스 작업 통계를 표시합니다.

```
$ sar -a
```

```
SunOS balmyday 5.10 s10_51 sun4u 03/18/2004
```

```
00:00:00  iget/s namei/s dirbk/s
01:00:00      0      3      0
02:00:00      0      3      0
03:00:00      0      3      0
04:00:00      0      3      0
05:00:00      0      3      0
06:00:00      0      3      0
07:00:00      0      3      0
08:00:00      0      3      0
08:20:01      0      3      0
08:40:00      0      3      0
09:00:00      0      3      0
09:20:01      0     10      0
09:40:01      0      1      0
10:00:02      0      5      0

Average      0      4      0
```

다음 목록은 sar -a 명령으로 보고되는 필드 이름 및 운영 체제 루틴을 설명합니다.

- iget/s 디렉토리 이름 조회 캐시(DNLC)에 없는 inode에 대한 요청 수입입니다.
- namei/s 초당 파일 시스템 경로 검색 수입입니다. namei가 DNLC에서 디렉토리 이름을 찾지 못할 경우 iget을 호출하여 파일이나 디렉토리에 대한 inode를 가져옵니다. 따라서 대부분의 igets는 DNLC 실패의 결과입니다.
- dirbk/s 초당 수행된 디렉토리 블록 읽기 수입입니다.

이러한 운영 체제 루틴에 대해 보고되는 값이 클수록 커널이 사용자 파일에 액세스하는데 소비하는 시간이 많아집니다. 시간의 양에는 얼마나 많은 프로그램 및 응용 프로그램이 파일 시스템을 사용하고 있는지 반영됩니다. -a 옵션은 응용 프로그램이 얼마나 디스크 종속적인지 보는 데 유용합니다.

▼ 버퍼 작업을 확인하는 방법(sar -b)

- sar -b 명령을 사용하여 버퍼 작업 통계를 표시합니다.

버퍼는 메타 데이터를 캐시에 저장하는 데 사용됩니다. 메타 데이터에는 inode, 실린더 그룹 블록 및 간접 블록이 포함됩니다.

```
$ sar -b
00:00:00  bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s pwrit/s
01:00:00      0      0     100      0      0     55      0      0
```

예 13-5 버퍼 작업 확인(sar -b)

sar -b 명령 출력의 다음 예는 %rcache 및 %wcache 버퍼가 성능 저하를 유발하고 있지 않음을 보여줍니다. 모든 데이터가 허용 가능한 한도 내에 있습니다.

```
$ sar -b
```

```

SunOS balmyday 5.10 s10_51 sun4u    03/18/2004

00:00:04 bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s pwrit/s
01:00:00      0      0    100      0      0     94      0      0
02:00:01      0      0    100      0      0     94      0      0
03:00:00      0      0    100      0      0     92      0      0
04:00:00      0      1    100      0      1     94      0      0
05:00:00      0      0    100      0      0     93      0      0
06:00:00      0      0    100      0      0     93      0      0
07:00:00      0      0    100      0      0     93      0      0
08:00:00      0      0    100      0      0     93      0      0
08:20:00      0      1    100      0      1     94      0      0
08:40:01      0      1    100      0      1     93      0      0
09:00:00      0      1    100      0      1     93      0      0
09:20:00      0      1    100      0      1     93      0      0
09:40:00      0      2    100      0      1     89      0      0
10:00:00      0      9    100      0      5     92      0      0
10:20:00      0      0    100      0      0     68      0      0
10:40:00      0      1     98      0      1     70      0      0
11:00:00      0      1    100      0      1     75      0      0

Average      0      1    100      0      1     91      0      0
    
```

다음 표는 -b 옵션으로 표시되는 버퍼 작업을 표시합니다.

필드 이름	설명
bread/s	디스크에서 버퍼 캐시로 제출된 초당 평균 읽기 수
lread/s	버퍼 캐시에서 초당 평균 논리적 읽기 수
%rcache	버퍼 캐시에서 발견된 논리적 읽기의 비율(100% - bread/s/lread/s 비율)
bwrit/s	버퍼 캐시에서 디스크에 쓰여진 초당 평균 물리적 블록(512 블록) 수
lwrit/s	버퍼 캐시로 초당 평균 논리적 쓰기 수
%wcache	버퍼 캐시에서 발견된 논리적 쓰기의 비율(100% - bwrit/s/lwrit/s의 비율)
pread/s	문자 장치 인터페이스를 사용하는 초당 평균 물리적 읽기 수
pwrit/s	문자 장치 인터페이스를 사용하는 초당 평균 물리적 쓰기 요청 수

가장 중요한 항목은 캐시 적중률 %rcache 및 %wcache입니다. 이러한 항목은 시스템 버퍼링의 유효성을 측정합니다. %rcache가 90% 아래로 떨어지거나 %wcache가 65% 아래로 떨어질 경우 버퍼 공간을 늘려 성능을 향상시킬 수 있습니다.

▼ 시스템 호출 통계를 확인하는 방법(sar -c)

- sar -c 명령을 사용하여 시스템 호출 통계를 표시합니다.

```
$ sar -c
00:00:00 scall/s sread/s swrit/s fork/s exec/s rchar/s wchar/s
01:00:00      38      2      2  0.00  0.00   149   120
```

예 13-6 시스템 호출 통계 확인(sar -c)

다음 예는 sar -c 명령의 출력을 보여줍니다.

```
$ sar -c
SunOS balmyday 5.10 s10_51 sun4u    03/18/2004

00:00:04 scall/s sread/s swrit/s fork/s exec/s rchar/s wchar/s
01:00:00      89      14      9  0.01  0.00   2906   2394
02:00:01      89      14      9  0.01  0.00   2905   2393
03:00:00      89      14      9  0.01  0.00   2908   2393
04:00:00      90      14      9  0.01  0.00   2912   2393
05:00:00      89      14      9  0.01  0.00   2905   2393
06:00:00      89      14      9  0.01  0.00   2905   2393
07:00:00      89      14      9  0.01  0.00   2905   2393
08:00:00      89      14      9  0.01  0.00   2906   2393
08:20:00      90      14      9  0.01  0.01   2914   2395
08:40:01      90      14      9  0.01  0.00   2914   2396
09:00:00      90      14      9  0.01  0.01   2915   2396
09:20:00      90      14      9  0.01  0.01   2915   2396
09:40:00     880     207     156  0.08  0.08  26671   9290
10:00:00    2020     530     322  0.14  0.13  57675  36393
10:20:00     853     129      75  0.02  0.01  10500   8594
10:40:00    2061     524     450  0.08  0.08  579217  567072
11:00:00    1658     404     350  0.07  0.06 1152916 1144203

Average      302      66      49  0.02  0.01  57842  55544
```

다음 표는 -c 옵션으로 보고되는 시스템 호출 범주를 설명합니다. 일반적으로 읽기 및 쓰기가 전체 시스템 호출의 약 절반을 차지합니다. 하지만 비율은 시스템에서 수행되는 작업에 따라 크게 달라집니다.

필드 이름	설명
scall/s	초당 모든 유형의 시스템 호출 수입입니다. 일반적으로 4-6명의 시스템에서 초당 약 30입니다.
sread/s	초당 read 시스템 호출 수입입니다.
swrit/s	초당 write 시스템 호출 수입입니다.
fork/s	초당 fork 시스템 호출 수입입니다. 4-6명의 시스템에서 초당 약 0.5입니다. 셸 스크립트가 실행되는 경우 이 수는 늘어납니다.

필드 이름	설명
exec/s	초당 exec 시스템 호출 수입니다. fork/s로 나눈 exec/s가 3보다 클 경우 비효율적인 PATH 변수를 찾으십시오.
rchar/s	초당 read 시스템 호출로 전송된 문자 수(바이트)입니다.
wchar/s	초당 write 시스템 호출로 전송된 문자 수(바이트)입니다.

▼ 디스크 작업을 확인하는 방법(sar -d)

- sar -d 명령을 사용하여 디스크 작업 통계를 표시합니다.

```
$ sar -d
00:00:00 device          %busy  avque  r+w/s  blks/s  await  avserv
```

예 13-7 디스크 작업 확인

이 약식 예는 sar -d 명령의 출력을 보여줍니다.

```
$ sar -d
SunOS balmyday 5.10 s10_51 sun4u    03/18/2004
12:36:32 device          %busy  avque  r+w/s  blks/s  await  avserv
12:40:01 dad1             15     0.7    26     399     18.1   10.0
        dad1,a        15     0.7    26     398     18.1   10.0
        dad1,b         0     0.0    0       1       1.0    3.0
        dad1,c         0     0.0    0       0       0.0    0.0
        dad1,h         0     0.0    0       0       0.0    6.0
        fd0            0     0.0    0       0       0.0    0.0
        nfs1           0     0.0    0       0       0.0    0.0
        nfs2           1     0.0    1      12      0.0   13.2
        nfs3           0     0.0    0       2       0.0    1.9
        nfs4           0     0.0    0       0       0.0    7.0
        nfs5           0     0.0    0       0       0.0   57.1
        nfs6           1     0.0    6     125     4.3    3.2
        nfs7           0     0.0    0       0       0.0    6.0
        sd1            0     0.0    0       0       0.0    5.4
        ohci0,bu       0     0.0    0       0       0.0    0.0
        ohci0,ct       0     0.0    0       0       0.0    0.0
        ohci0,in       0     0.0    7       0       0.0    0.0
        ohci0,is       0     0.0    0       0       0.0    0.0
        ohci0,to       0     0.0    7       0       0.0    0.0
```

다음 표는 -d 옵션으로 보고되는 디스크 장치 작업을 설명합니다.

필드 이름	설명
device	모니터링되는 디스크 장치의 이름입니다.

필드 이름	설명
%busy	장치가 전송 요청을 서비스하는 시간입니다.
avque	장치가 전송 요청을 서비스하는 시간 동안의 평균 요청 수입입니다.
r+w/s	장치로 초당 읽기 및 쓰기 전송 수입입니다.
blks/s	장치로 전송된 초당 512바이트 블록 수입입니다.
await	전송 요청이 대기열에서 유휴 상태로 대기하는 평균 시간(밀리초)입니다. 이 시간은 대기열이 채워진 경우에만 측정됩니다.
avserv	장치에서 전송 요청을 완료하는 평균 시간(밀리초)입니다. 디스크의 경우 이 값에는 탐색 시간, 회전 지연 시간 및 데이터 전송 시간이 포함됩니다.

대기열 길이 및 대기 시간은 대기열이 채워진 경우에만 측정됩니다. %busy가 작은 경우 큰 대기열 및 서비스 시간은 시스템에서 변경된 블록이 디스크에 즉시 쓰여지도록 주기적인 작업을 수행하고 있음을 나타냅니다.

▼ 페이지 아웃 및 메모리를 확인하는 방법(sar -g)

- sar -g 명령을 사용하여 평균 페이지 아웃 및 메모리 해제 작업을 표시합니다.

```
$ sar -g
00:00:00 pgout/s ppgout/s pgfree/s pgscan/s %ufs_ipf
01:00:00 0.00 0.00 0.00 0.00 0.00
```

sar -g 명령으로 표시되는 출력은 더 많은 메모리가 필요한지 여부를 알 수 있는 좋은 지표입니다. ps -elf 명령을 사용하여 page 데몬에서 사용되는 주기 수를 표시합니다. pgfree/s 및 pgscan/s 필드에 대한 높은 값과 함께 높은 주기 수는 메모리 부족을 나타냅니다.

또한 sar -g 명령은 inode가 너무 빨리 재사용되어 재사용 가능한 페이지 유실을 일으키는 지 여부를 보여줍니다.

예 13-8 페이지 아웃 및 메모리 확인(sar -g)

다음 예는 sar -g 명령의 출력을 보여줍니다.

```
$ sar -g
SunOS balmyday 5.10 s10_51 sun4u 03/18/2004

00:00:00 pgout/s ppgout/s pgfree/s pgscan/s %ufs_ipf
01:00:00 0.00 0.00 0.00 0.00 0.00
02:00:00 0.01 0.01 0.01 0.00 0.00
```

03:00:00	0.00	0.00	0.00	0.00	0.00
04:00:00	0.00	0.00	0.00	0.00	0.00
05:00:00	0.00	0.00	0.00	0.00	0.00
06:00:00	0.00	0.00	0.00	0.00	0.00
07:00:00	0.00	0.00	0.00	0.00	0.00
08:00:00	0.00	0.00	0.00	0.00	0.00
08:20:01	0.00	0.00	0.00	0.00	0.00
08:40:00	0.00	0.00	0.00	0.00	0.00
09:00:00	0.00	0.00	0.00	0.00	0.00
09:20:01	0.05	0.52	1.62	10.16	0.00
09:40:01	0.03	0.44	1.47	4.77	0.00
10:00:02	0.13	2.00	4.38	12.28	0.00
10:20:03	0.37	4.68	12.26	33.80	0.00
Average	0.02	0.25	0.64	1.97	0.00

다음 표는 -g 옵션의 출력을 설명합니다.

필드 이름	설명
pgout/s	초당 페이지 아웃 요청 수입니다.
ppgout/s	초당 페이지 아웃된 실제 페이지 수입니다. 단일 페이지 아웃 요청에는 여러 페이지의 페이지 아웃이 포함될 수 있습니다.
pgfree/s	여유 목록에 있는 초당 페이지 수입니다.
pgscan/s	page 데몬으로 스캔된 초당 페이지 수입니다. 이 값이 높을 경우 page 데몬이 여유 메모리를 확인하는 데 많은 시간을 소비하고 있는 것입니다. 이 상황은 더 많은 메모리가 필요함을 나타냅니다.
%ufs_ipf	연관된 재사용 가능한 페이지가 있는 여유 목록에서 iget으로 가져간 ufs inode의 백분율입니다. 이러한 페이지는 비워지며 프로세스에서 재확보할 수 없습니다. 따라서 이 필드는 페이지 비우기에서 igets의 백분율을 나타냅니다. 높은 값은 inode의 여유 목록이 페이지 제한적이고, ufs inode의 수를 늘려야 함을 나타냅니다.

커널 메모리 할당 확인

KMA를 통해 커널 부속 시스템에서 필요에 따라 메모리를 할당하고 해제할 수 있습니다.

부하가 가장 많은 시간대에 필요할 것으로 예상되는 최대 메모리 양을 정적으로 할당하는 대신 KMA는 메모리에 대한 요청을 세 범주로 나눕니다.

- 작음(256바이트 미만)
- 큼(512바이트-4킬로바이트)
- 매우 큼(4킬로바이트 이상)

KMA는 작은 요청과 큰 요청을 충족하기 위해 두 메모리 풀을 유지합니다. 매우 큰 요청은 시스템 페이지 할당기에서 메모리를 할당하여 충족됩니다.

KMA 리소스를 사용하는 드라이버 또는 STREAMS를 쓰는 데 사용되는 시스템을 확인하는 경우 sar -k 명령이 유용할 수 있습니다. 그렇지 않은 경우 제공되는 정보가 필요하지 않을 수 있습니다. KMA 리소스를 사용하지만 종료하기 전에 명시적으로 리소스를 반환하지 않는 드라이버나 모듈은 메모리 누수를 유발할 수 있습니다. 메모리 누수로 인해 시간에 따라 KMA로 할당되는 메모리의 양이 늘어나게 됩니다. 따라서 sar -k 명령의 alloc 필드가 시간에 따라 계속 늘어나면 메모리 누수가 있을 수 있습니다. 또 하나의 메모리 누수 지표는 요청 실패입니다. 이 문제가 발생할 경우 메모리 누수로 인해 KMA에서 메모리를 확보하고 할당할 수 없게 됩니다.

메모리 누수가 발생할 것으로 보이면 KMA에서 메모리를 요청하고 반환하지 않는 드라이버나 STREAMS를 확인해야 합니다.

▼ 커널 메모리 할당을 확인하는 방법(sar -k)

- sar -k 명령을 사용하여 커널 메모리 할당기(KMA)의 다음 작업을 보고합니다.

```
$ sar -k
00:00:00 sml_mem alloc fail lg_mem alloc fail ovsz alloc fail
01:00:00 2523136 1866512 0 18939904 14762364 0 360448 0
02:00:02 2523136 1861724 0 18939904 14778748 0 360448 0
```

예 13-9 커널 메모리 할당 확인(sar -k)

다음은 sar -k 출력의 약식 예입니다.

```
$ sar -k
SunOS balmyday 5.10 s10_51 sun4u 03/18/2004
00:00:04 sml_mem alloc fail lg_mem alloc fail ovsz_alloc fail
01:00:00 6119744 4852865 0 60243968 54334808 156 9666560 0
02:00:01 6119744 4853057 0 60243968 54336088 156 9666560 0
03:00:00 6119744 4853297 0 60243968 54335760 156 9666560 0
04:00:00 6119744 4857673 0 60252160 54375280 156 9666560 0
05:00:00 6119744 4858097 0 60252160 54376240 156 9666560 0
06:00:00 6119744 4858289 0 60252160 54375608 156 9666560 0
07:00:00 6119744 4858793 0 60252160 54442424 156 9666560 0
08:00:00 6119744 4858985 0 60252160 54474552 156 9666560 0
08:20:00 6119744 4858169 0 60252160 54377400 156 9666560 0
08:40:01 6119744 4857345 0 60252160 54376880 156 9666560 0
09:00:00 6119744 4859433 0 60252160 54539752 156 9666560 0
09:20:00 6119744 4858633 0 60252160 54410920 156 9666560 0
09:40:00 6127936 5262064 0 60530688 55619816 156 9666560 0
10:00:00 6545728 5823137 0 62996480 58391136 156 9666560 0
10:20:00 6545728 5758997 0 62996480 57907400 156 9666560 0
10:40:00 6734144 6035759 0 64389120 59743064 156 10493952 0
11:00:00 6996288 6394872 0 65437696 60935936 156 10493952 0
```

Average 6258044 5150556 0 61138340 55609004 156 9763900 0

다음 표는 -k 옵션의 출력을 설명합니다.

필드 이름	설명
sml_mem	KMA가 작은 메모리 요청 풀에서 사용할 수 있는 메모리의 양(바이트)입니다. 이 풀에서 작은 요청은 256바이트 미만입니다.
alloc	KMA가 작은 메모리 요청 풀에서 작은 메모리 요청에 할당된 메모리의 양(바이트)입니다.
fail	작은 메모리 양에 대해 실패한 요청 수입니다.
lg_mem	KMA가 큰 메모리 요청 풀에서 사용할 수 있는 메모리의 양(바이트)입니다. 이 풀에서 큰 요청은 512바이트-4킬로바이트입니다.
alloc	KMA가 큰 메모리 요청 풀에서 큰 메모리 요청에 할당된 메모리의 양(바이트)입니다.
fail	큰 메모리 양에 대해 실패한 요청 수입니다.
ovsz_alloc	4킬로바이트보다 큰 요청인 매우 큰 요청에 대해 할당된 메모리의 양입니다. 이러한 요청은 페이지 할당기로 충족됩니다. 따라서 풀이 없습니다.
fail	매우 큰 메모리 양에 대해 실패한 요청 수입니다.

▼ 프로세스간 통신을 확인하는 방법(sar -m)

- sar -m 명령을 사용하여 프로세스간 통신 작업을 보고합니다.

```
$ sar -m
00:00:00 msg/s sema/s
01:00:00 0.00 0.00
```

메시지나 세마포어를 사용하는 응용 프로그램을 실행하지 않는다면 이러한 수치는 대개 0.00입니다.

다음 목록은 -m 옵션의 출력을 설명합니다.

```
msg/s    초당 메시지 작업(송수신) 수
sema/s   초당 세마포어 작업 수
```

예 13-10 프로세스간 통신 확인(sar -m)

다음 약식 예는 sar -m 명령의 출력을 보여줍니다.

```

$ sar -m

SunOS balmyday 5.10 s10_51 sun4u    03/18/2004

00:00:00  msg/s  sema/s
01:00:00  0.00   0.00
02:00:02  0.00   0.00
03:00:00  0.00   0.00
04:00:00  0.00   0.00
05:00:01  0.00   0.00
06:00:00  0.00   0.00

Average   0.00   0.00

```

▼ 페이지 인 작업을 확인하는 방법(sar -p)

- sar -p 명령을 사용하여 보호 및 변환 폴트를 포함하는 페이지 인 작업을 보고합니다.

```

$ sar -p
00:00:00  atch/s  pgin/s  ppgin/s  pflt/s  vflt/s  slock/s
01:00:00  0.07    0.00    0.00    0.21    0.39    0.00

```

예 13-11 페이지 인 작업 확인(sar -p)

다음 예는 sar -p 명령의 출력을 보여줍니다.

```

$ sar -p

SunOS balmyday 5.10 s10_51 sun4u    03/18/2004

00:00:04  atch/s  pgin/s  ppgin/s  pflt/s  vflt/s  slock/s
01:00:00  0.09    0.00    0.00    0.78    2.02    0.00
02:00:01  0.08    0.00    0.00    0.78    2.02    0.00
03:00:00  0.09    0.00    0.00    0.81    2.07    0.00
04:00:00  0.11    0.01    0.01    0.86    2.18    0.00
05:00:00  0.08    0.00    0.00    0.78    2.02    0.00
06:00:00  0.09    0.00    0.00    0.78    2.02    0.00
07:00:00  0.08    0.00    0.00    0.78    2.02    0.00
08:00:00  0.09    0.00    0.00    0.78    2.02    0.00
08:20:00  0.11    0.00    0.00    0.87    2.24    0.00
08:40:01  0.13    0.00    0.00    0.90    2.29    0.00
09:00:00  0.11    0.00    0.00    0.88    2.24    0.00
09:20:00  0.10    0.00    0.00    0.88    2.24    0.00
09:40:00  2.91    1.80    2.38    4.61    17.62    0.00
10:00:00  2.74    2.03    3.08    8.17    21.76    0.00
10:20:00  0.16    0.04    0.04    1.92    2.96    0.00
10:40:00  2.10    2.50    3.42    6.62    16.51    0.00
11:00:00  3.36    0.87    1.35    3.92    15.12    0.00

Average   0.42    0.22    0.31    1.45    4.00    0.00

```

다음 표는 -p 옵션으로 보고된 통계를 설명합니다.

필드 이름	설명
atcb/s	현재 메모리에 있는 페이지를 확보하여 충족된 초당 페이지 폴트 수입니다(초당 첨부). 인스턴스에는 여유 목록에서 잘못된 페이지 확보 및 현재 다른 프로세스에서 사용되고 있는 텍스트 페이지 공유가 포함됩니다. 동일한 프로그램 텍스트에 액세스하고 있는 둘 이상의 프로세스를 예로 들 수 있습니다.
pgin/s	파일 시스템이 페이지 인 요청을 수신하는 초당 횟수입니다.
ppgin/s	초당 페이지 인된 페이지 수입니다. 소프트 잠금 요청(slock/s 참조) 또는 큰 블록 크기와 같은 단일 페이지 인 요청에는 여러 페이지의 페이지 인이 포함될 수 있습니다.
pflt/s	보호 오류의 페이지 폴트 수입니다. 보호 결함 인스턴스는 페이지 및 “쓰기 시 복사”에 대한 잘못된 액세스를 나타냅니다. 일반적으로 이 수는 주로 “쓰기 시 복사”로 구성됩니다.
vflt/s	초당 주소 변환 페이지 폴트 수입니다. 이러한 결함을 유효성 결함이라고 합니다. 유효성 결함은 유효한 프로세스 테이블 항목이 해당 가상 주소에 대해 존재하지 않을 때 발생합니다.
slock/s	물리적 I/O가 필요한 소프트웨어 잠금 요청으로 인해 발생하는 초당 결함 수입니다. 예를 들어, 소프트웨어 잠금 요청은 디스크에서 메모리로 데이터를 전송할 때 발생할 수 있습니다. 시스템이 데이터를 수신해야 하는 페이지를 잠그므로 다른 페이지에서 페이지를 확보하고 사용할 수 없습니다.

▼ 대기열 작업을 확인하는 방법(sar -q)

- **sar -q 명령을 사용하여 다음 정보를 보고합니다.**

- 대기열이 채워진 동안의 평균 대기열 길이
- 대기열이 채워진 시간(백분율)

```
$ sar -q
00:00:00 runq-sz %runocc swpq-sz %swpocc
```

다음 목록은 -q 옵션의 출력을 설명합니다.

- runq-sz 실행을 위해 CPU를 대기 중인 메모리의 커널 스레드 수입니다. 일반적으로 이 값은 2보다 작아야 합니다. 지속적으로 높은 값은 시스템이 CPU 제한적임을 나타냅니다.
- %runocc 전달 대기열이 채워진 시간의 백분율입니다.
- swpq-sz 스왑 아웃된 프로세스의 평균 개수입니다.
- %swpocc 프로세스가 스왑 아웃된 시간의 백분율입니다.

예 13-12 대기열 작업 확인

다음 예는 sar -q 명령의 출력을 보여줍니다. %runocc 값이 높고(90% 이상) runq-sz 값이 2보다 클 경우, CPU 부하가 높고 응답 속도가 떨어집니다. 이 경우 허용할 만한 시스템 응답 속도를 얻으려면 추가 CPU가 필요할 수 있습니다.

```
# sar -q
SunOS system2 5.10 Generic_142909-13 sun4u    06/28/2010

00:00:00 runq-sz %runocc swpq-sz %swpocc
01:00:00    1.0      7      0.0      0
02:00:00    1.0      7      0.0      0
03:00:00    1.0      7      0.0      0
04:00:00    1.0      7      0.0      0
05:00:00    1.0      6      0.0      0
06:00:00    1.0      7      0.0      0

Average      1.0      7      0.0      0
```

▼ 사용되지 않은 메모리를 확인하는 방법(sar -r)

- sar -r 명령을 사용하여 현재 사용되지 않은 메모리 페이지 및 스왑 파일 디스크 블록 수를 보고합니다.

```
$ sar -r
00:00:00 freemem freeswap
01:00:00    2135    401922
```

다음 목록은 -r 옵션의 출력을 설명합니다.

```
freemem    명령으로 샘플링된 간격 동안 사용자 프로세스에서 사용할 수 있는 평균
           메모리 페이지 수입니다. 페이지 크기는 시스템에 따라 다릅니다.

freeswap   페이지 스와핑에 사용할 수 있는 512바이트 디스크 블록 수입니다.
```

예 13-13 사용되지 않은 메모리 확인(sar -r)

다음 예는 sar -r 명령의 출력을 보여줍니다.

```
$ sar -r
SunOS balmyday 5.10 s10_51 sun4u    03/18/2004

00:00:04 freemem freeswap
01:00:00    44717    1715062
02:00:01    44733    1715496
03:00:00    44715    1714746
04:00:00    44751    1715403
05:00:00    44784    1714743
06:00:00    44794    1715186
07:00:00    44793    1715159
```

```

08:00:00  44786  1714914
08:20:00  44805  1715576
08:40:01  44797  1715347
09:00:00  44761  1713948
09:20:00  44802  1715478
09:40:00  41770  1682239
10:00:00  35401  1610833
10:20:00  34295  1599141
10:40:00  33943  1598425
11:00:00  30500  1561959

Average   43312  1699242
    
```

▼ CPU 사용률을 확인하는 방법(sar -u)

- sar -u 명령을 사용하여 CPU 사용률 통계를 표시합니다.

```

$ sar -u
00:00:00  %usr  %sys  %wio  %idle
01:00:00      0     0     0    100
    
```

옵션이 없는 sar 명령은 sar -u 명령과 같습니다. 특정 시점에 프로세서는 사용 상태이거나 유휴 상태입니다. 사용 상태일 때 프로세서는 사용자 모드이거나 시스템 모드입니다. 유휴 상태일 때 프로세서는 I/O 완료를 대기 중이거나 할 일이 없어 “단순 유휴 상태”입니다.

다음 목록은 -u 옵션의 출력을 설명합니다.

%usr 프로세서가 사용자 모드인 시간의 백분율을 나열합니다.

%sys 프로세서가 시스템 모드인 시간의 백분율을 나열합니다.

%wio 프로세서가 유휴 상태이고 I/O 완료를 대기 중인 시간의 백분율을 나열합니다.

%idle 프로세서가 유휴 상태이고 I/O를 대기 중이 아닌 시간의 백분율을 나열합니다.

높은 %wio 값은 일반적으로 디스크 성능 저하가 발생함을 의미합니다.

예 13-14 CPU 사용률 확인(sar -u)

다음 예는 sar -u 명령의 출력을 나타냅니다.

```

$ sar -u

SunOS balmyday 5.10 s10_51 sun4u   03/18/2004

00:00:04  %usr  %sys  %wio  %idle
01:00:00      0     0     0    100
02:00:01      0     0     0    100
03:00:00      0     0     0    100
04:00:00      0     0     0    100
05:00:00      0     0     0    100
    
```

06:00:00	0	0	0	100
07:00:00	0	0	0	100
08:00:00	0	0	0	100
08:20:00	0	0	0	99
08:40:01	0	0	0	99
09:00:00	0	0	0	99
09:20:00	0	0	0	99
09:40:00	4	1	0	95
10:00:00	4	2	0	94
10:20:00	1	1	0	98
10:40:00	18	3	0	79
11:00:00	25	3	0	72
Average	2	0	0	98

▼ 시스템 테이블 상태를 확인하는 방법(sar -v)

- **sar -v** 명령을 사용하여 프로세스 테이블, inode 테이블, 파일 테이블 및 공유 메모리 레코드 테이블의 상태를 보고합니다.

```
$ sar -v
00:00:00 proc-sz   ov  inod-sz   ov  file-sz   ov  lock-sz
01:00:00  43/922      0 2984/4236  0  322/322   0   0/0
```

예 13-15 시스템 테이블 상태 확인(sar -v)

다음 약식 예는 sar -v 명령의 출력을 보여줍니다. 이 예는 모든 테이블이 오버플로우가 없을 만큼 큰 경우입니다. 이러한 테이블은 물리적 메모리의 양을 기준으로 모두 동적으로 할당됩니다.

```
$ sar -v
SunOS balmyday 5.10 s10_51 sun4u   03/18/2004

00:00:04 proc-sz   ov  inod-sz   ov  file-sz   ov  lock-sz
01:00:00  69/8010   0 3476/34703  0   0/0      0   0/0
02:00:01  69/8010   0 3476/34703  0   0/0      0   0/0
03:00:00  69/8010   0 3476/34703  0   0/0      0   0/0
04:00:00  69/8010   0 3494/34703  0   0/0      0   0/0
05:00:00  69/8010   0 3494/34703  0   0/0      0   0/0
06:00:00  69/8010   0 3494/34703  0   0/0      0   0/0
07:00:00  69/8010   0 3494/34703  0   0/0      0   0/0
08:00:00  69/8010   0 3494/34703  0   0/0      0   0/0
08:20:00  69/8010   0 3494/34703  0   0/0      0   0/0
08:40:01  69/8010   0 3494/34703  0   0/0      0   0/0
09:00:00  69/8010   0 3494/34703  0   0/0      0   0/0
09:20:00  69/8010   0 3494/34703  0   0/0      0   0/0
09:40:00  74/8010   0 3494/34703  0   0/0      0   0/0
10:00:00  75/8010   0 4918/34703  0   0/0      0   0/0
10:20:00  72/8010   0 4918/34703  0   0/0      0   0/0
10:40:00  71/8010   0 5018/34703  0   0/0      0   0/0
11:00:00  77/8010   0 5018/34703  0   0/0      0   0/0
```

-v 옵션의 출력은 다음 표에 설명되어 있습니다.

필드 이름	설명
proc-sz	커널에서 현재 사용되고 있거나 할당된 프로세스 항목(proc 구조) 수입니다.
inod-sz	커널에서 할당된 최대 inode 수와 비교한 메모리의 총 inode 수입니다. 이 수는 엄격한 고수위가 아닙니다. 수는 오버플로우될 수 있습니다.
file-sz	열린 시스템 파일 테이블의 크기입니다. 공간이 파일 테이블에 대해 동적으로 할당되므로 sz는 0으로 지정됩니다.
ov	각 테이블에 대한 샘플링 시점 사이에 발생하는 오버플로우입니다.
lock-sz	커널에서 현재 사용되고 있거나 할당된 공유 메모리 레코드 테이블 항목 수입니다. 공간이 공유 메모리 레코드 테이블에 대해 동적으로 할당되므로 sz는 0으로 지정됩니다.

▼ 스와핑 작업을 확인하는 방법(sar -w)

- sar -w 명령을 사용하여 스와핑 및 전환 작업을 보고합니다.

```
$ sar -w
00:00:00 swpin/s bswin/s swpot/s bswo/s pswch/s
01:00:00 0.00 0.0 0.00 0.0 22
```

다음 목록은 sar -w 명령 출력과 관련된 목표 값 및 관찰을 설명합니다.

swpin/s 초당 메모리로 LWP 전송 수입니다.

bswin/s 초당 스왑 인을 위해 전송된 블록 수입니다. /* (float)PGTOBLK(xx->cvmi.pgswapin) / sec_diff */.

swpot/s 초당 메모리에서 스왑 아웃된 평균 프로세스 수입니다. 이 수가 1보다 클 경우 메모리를 늘려야 합니다.

bswo/s 초당 스왑 아웃을 위해 전송된 블록 수입니다.

pswch/s 초당 커널 스레드 전환 수입니다.

주 - 모든 프로세스 스왑 인에는 프로세스 초기화가 포함됩니다.

예 13-16 스왑 작업 확인(sar -w)

다음 예는 sar -w 명령의 출력을 보여줍니다.

```
$ sar -w
SunOS balmyday 5.10 s10_51 sun4u    03/18/2004

00:00:04 swpin/s bswin/s swpot/s bswot/s pswch/s
01:00:00    0.00    0.0    0.00    0.0    132
02:00:01    0.00    0.0    0.00    0.0    133
03:00:00    0.00    0.0    0.00    0.0    133
04:00:00    0.00    0.0    0.00    0.0    134
05:00:00    0.00    0.0    0.00    0.0    133
06:00:00    0.00    0.0    0.00    0.0    133
07:00:00    0.00    0.0    0.00    0.0    132
08:00:00    0.00    0.0    0.00    0.0    131
08:20:00    0.00    0.0    0.00    0.0    133
08:40:01    0.00    0.0    0.00    0.0    132
09:00:00    0.00    0.0    0.00    0.0    132
09:20:00    0.00    0.0    0.00    0.0    132
09:40:00    0.00    0.0    0.00    0.0    335
10:00:00    0.00    0.0    0.00    0.0    601
10:20:00    0.00    0.0    0.00    0.0    353
10:40:00    0.00    0.0    0.00    0.0    747
11:00:00    0.00    0.0    0.00    0.0    804

Average    0.00    0.0    0.00    0.0    198
```

▼ 터미널 작업을 확인하는 방법(sar -y)

- sar -y 명령을 사용하여 터미널 장치 작업을 모니터링합니다.

```
$ sar -y
00:00:00 rawch/s canch/s outch/s rcvin/s xmtin/s madmin/s
01:00:00    0    0    0    0    0    0
```

많은 터미널 I/O가 있을 경우 이 보고서를 사용하여 잘못된 라인이 존재하는지 확인할 수 있습니다. 기록된 작업은 다음 목록에서 정의됩니다.

rawch/s 초당 입력 문자(원시 대기열)입니다.

canch/s 초당 캐논으로 처리된 입력 문자(캐논이컬 대기열)입니다.

outch/s 초당 출력 문자(출력 대기열)입니다.

rcvin/s 초당 수신기 하드웨어 인터럽트입니다.

xmtin/s 초당 전송기 하드웨어 인터럽트입니다.

madmin/s 초당 모뎀 인터럽트입니다.

초당 모뎀 인터럽트 수(mdmin/s)는 0에 가까워야 합니다. 초당 수신 및 전송 인터럽트(xmtin/s 및 rcvin/s)는 각각 수신 또는 전송 문자의 수보다 작거나 같아야 합니다. 그렇지 않은 경우 잘못된 라인을 확인합니다.

예 13-17 터미널 작업 확인(sar -y)

다음 예는 sar -y 명령의 출력을 보여줍니다.

```
$ sar -y
SunOS balmyday 5.10 s10_51 sun4u    03/18/2004

00:00:04 rawch/s  canch/s  outch/s  rcvin/s  xmtin/s  mdmin/s
01:00:00      0       0       0       0       0       0
02:00:01      0       0       0       0       0       0
03:00:00      0       0       0       0       0       0
04:00:00      0       0       0       0       0       0
05:00:00      0       0       0       0       0       0
06:00:00      0       0       0       0       0       0
07:00:00      0       0       0       0       0       0
08:00:00      0       0       0       0       0       0
08:20:00      0       0       0       0       0       0
08:40:01      0       0       0       0       0       0
09:00:00      0       0       0       0       0       0
09:20:00      0       0       0       0       0       0
09:40:00      0       0       1       0       0       0
10:00:00      0       0      37       0       0       0
10:20:00      0       0       0       0       0       0
10:40:00      0       0       3       0       0       0
11:00:00      0       0       3       0       0       0

Average      0       0       1       0       0       0
```

▼ 전체 시스템 성능을 확인하는 방법(sar -A)

- sar -A 명령을 사용하여 모든 옵션의 통계를 표시하여 전체 시스템 성능 보기를 제공합니다.

이 명령은 좀더 포괄적인 관점을 제공합니다. 단일 시점 세그먼트의 데이터보다 많은 데이터가 표시될 경우 보고서에는 평균이 포함됩니다.

자동으로 시스템 작업 데이터 수집(sar)

세 명령은 sadc, sa1 및 sa2의 시스템 작업 데이터 자동 수집에 포함됩니다.

sadc 데이터 수집 유틸리티는 시스템 작업에 대한 데이터를 정기적으로 수집하고 이진 형식의 파일(24시간 기간마다 하나씩)에 데이터를 저장합니다. sadc 명령이

정기적으로(대개 매시간마다 한번), 그리고 시스템이 다중 사용자 모드로 부트될 때마다 실행되도록 설정할 수 있습니다. 데이터 파일은 `/var/adm/sa` 디렉토리에 보관됩니다. 각 파일의 이름은 `sadd`로 지정됩니다. 여기서 `dd`는 현재 날짜입니다. 명령의 형식은 다음과 같습니다.

```
/usr/lib/sa/sadc [t n] [ofile]
```

명령은 `t`초 간격으로 `n`회 샘플링하며, 샘플링 간격은 5초보다 길어야 합니다. 그런 다음 이 명령은 이진 `ofile` 파일이나 표준 출력에 씁니다.

부트할 때 `sadc` 명령 실행

카운터가 0으로 재설정될 때부터 통계를 기록하려면 `sadc` 명령이 시스템 부트 시 실행되어야 합니다. `sadc` 명령이 부트 시 실행되도록 하기 위해 `svcadm enable system/sar:default` 명령이 레코드를 일별 데이터 파일에 씁니다.

명령 항목의 형식은 다음과 같습니다.

```
/usr/bin/su sys -c "/usr/lib/sa/sadc /var/adm/sa/sa`date +%d`"
```

sa1 스크립트를 사용하여 정기적으로 `sadc` 명령 실행

정기적인 레코드를 생성하려면 `sadc` 명령을 정기적으로 실행해야 합니다. 이를 위한 가장 간단한 방법은 `/var/spool/cron/crontabs/sys` 파일에서 다음 줄을 주석 처리하는 것입니다.

```
# 0 * * * 0-6 /usr/lib/sa/sa1
# 20,40 8-17 * * 1-5 /usr/lib/sa/sa1
# 5 18 * * 1-5 /usr/lib/sa/sa2 -s 8:00 -e 18:01 -i 1200 -A
```

`sys` crontab 항목은 다음을 수행합니다.

- 처음 두 crontab 항목은 월요일부터 금요일까지 오전 8시부터 오후 5시까지 20분마다(그렇지 않은 경우 매시간마다) 레코드가 `/var/adm/sa/sadd` 파일에 쓰여지도록 합니다.
- 세번째 항목은 월요일부터 금요일까지 매시간마다 레코드를 `/var/adm/sa/sar dd` 파일에 쓰고, 모든 `sar` 옵션을 포함합니다.

이러한 기본값은 필요에 맞게 변경할 수 있습니다.

sa2 셸 스크립트를 사용하여 보고서 생성

또 하나의 셸 스크립트인 sa2는 이진 데이터 파일 대신 보고서를 생성합니다. sa2 명령은 sar 명령을 호출하고 ASCII 출력을 보고서 파일에 씁니다.

자동 데이터 수집 설정(sar)

sar 명령은 시스템 작업 데이터 자체를 수집하거나 sadc 명령으로 생성된 일별 작업 파일에서 수집된 정보를 보고하는 데 사용할 수 있습니다.

sar 명령의 형식은 다음과 같습니다.

```
sar [-aAbcdgkmpqruvw] [-o file] t [n]
```

```
sar [-aAbcdgkmpqruvw] [-s time] [-e time] [-i sec] [-f file]
```

다음 sar 명령은 운영 체제에서 누적 작업 카운터를 t초마다 n회 샘플링합니다. t는 5초 이상이어야 합니다. 그렇지 않으면 명령 자체가 샘플에 영향을 줄 수 있습니다. 샘플링을 수행할 시간 간격을 지정해야 합니다. 그렇지 않으면 명령은 두번째 형식에 따라 작동합니다. n의 기본값은 1입니다. 다음 예는 10초로 구분된 두 번의 샘플링을 수행합니다. -o 옵션이 지정된 경우 샘플이 이진 형식으로 지정됩니다.

```
$ sar -u 10 2
```

sar 명령에 대한 기타 중요 정보에는 다음이 포함됩니다.

- 샘플링 간격이나 샘플 수가 지정되지 않으면 sar 명령은 이전에 기록된 파일에서 데이터를 추출합니다. 이 파일은 -f 옵션으로 지정된 파일이거나 기본적으로 최근의 표준 일별 작업 파일인 /var/adm/sa/sa dd입니다.
- -s 및 -e 옵션은 보고서에 대한 시작 시간과 종료 시간을 정의합니다. 시작 시간과 종료 시간의 형식은 hh[:mm[:ss]]입니다. 여기서 hh, mm 및 ss는 시, 분, 초를 나타냅니다.
- -i 옵션은 레코드 선택 사이의 간격(초)을 지정합니다. -i 옵션이 포함되지 않으면 일별 작업 파일에서 발견된 모든 간격이 보고됩니다.

다음 표는 sar 옵션 및 해당 작업을 나열합니다.

표 13-2 sar 명령에 대한 옵션

옵션	작업
-a	파일 액세스 작업을 확인합니다.
-b	버퍼 작업을 확인합니다.

표 13-2 sar 명령에 대한 옵션 (계속)

옵션	작업
-c	시스템 호출을 확인합니다.
-d	각 블록 장치에 대한 작업을 확인합니다.
-g	페이지 아웃 및 메모리 해제를 확인합니다.
-k	커널 메모리 할당을 확인합니다.
-m	프로세스간 통신을 확인합니다.
-nv	시스템 테이블 상태를 확인합니다.
-p	스왑 및 전달 작업을 확인합니다.
-q	대기열 작업을 확인합니다.
-r	사용되지 않은 메모리를 확인합니다.
-u	CPU 사용률을 확인합니다.
-w	스와핑 및 전환 볼륨을 확인합니다.
-y	터미널 작업을 확인합니다.
-A	전체 시스템 성능을 보고합니다. 모든 옵션을 입력하는 것과 같습니다.

옵션을 사용하지 않는 것은 sar 명령을 -u 옵션과 함께 사용하는 것과 같습니다.

▼ 자동 데이터 수집을 설정하는 방법

1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

2 `svcadm enable system/sar:default` 명령을 실행합니다.

이 버전의 `sadc` 명령은 카운터가 0으로 재설정될 때(부트 시) 시간을 표시하는 특수 레코드를 씁니다.

3 `/var/spool/cron/crontabs/sys crontab` 파일을 편집합니다.

주 - `crontab` 파일을 직접 편집하지 마십시오. 대신 `crontab -e` 명령을 사용하여 기존 `crontab` 파일을 변경합니다.

```
# crontab -e sys
```

4 다음 줄을 주석 처리합니다.

```
0 * * * 0-6 /usr/lib/sa/sa1
20,40 8-17 * * 1-5 /usr/lib/sa/sa1
5 18 * * 1-5 /usr/lib/sa/sa2 -s 8:00 -e 18:01 -i 1200 -A
```

자세한 내용은 [crontab\(1\)](#) 매뉴얼 페이지를 참조하십시오.

소프트웨어 문제 해결(개요)

이 장에서는 시스템 충돌 문제 해결 및 시스템 메시지 확인에 대한 정보를 비롯하여 소프트웨어 문제 해결에 대한 일반적인 개요를 제공합니다.

다음은 이 장에서 다루는 정보를 나열한 것입니다.

- 209 페이지 “문제 해결의 새로운 정보”
- 211 페이지 “소프트웨어 문제 해결 작업 검색 위치”
- 211 페이지 “시스템 충돌 문제 해결”
- 213 페이지 “시스템 충돌 문제 해결 점검 목록”

문제 해결의 새로운 정보

이 단원에서는 현재 릴리스에서 새로 구성되거나 변경된 문제 해결 정보에 대해 설명합니다.

Oracle Solaris 10 릴리스에서 새로 구성되거나 변경된 문제 해결 기능에 대한 자세한 내용은 다음을 참조하십시오.

- 210 페이지 “동적 추적 기능”
- 210 페이지 “표준 Solaris 커널 디버거로 kmdb가 kadb를 대체함”

새로운 기능의 전체 목록 및 Oracle Solaris 릴리스에 대한 설명은 [Oracle Solaris 10 8/11 새로운 기능](#)을 참조하십시오.

공통 에이전트 컨테이너 문제

Solaris 10 6/06: 공통 에이전트 컨테이너는 Oracle Solaris OS에 포함된 독립형 Java 프로그램입니다. 이 프로그램은 Java 관리 응용 프로그램용 컨테이너를 구현합니다. 공통 에이전트 컨테이너는 JMX(Java Management Extensions) 및 Java DMK(Java Dynamic Management Kit) 기반 기능용으로 설계된 관리 기반구조를 제공합니다. 소프트웨어는 SUNwcacaort 패키지를 통해 설치되며 /usr/lib/cacao 디렉토리에 상주합니다.

일반적으로 컨테이너는 표시되지 않지만 다음과 같은 두 가지 경우에 컨테이너 데몬과 상호 작용해야 할 수 있습니다.

- 다른 응용 프로그램이 공통 에이전트 컨테이너용으로 예약된 네트워크 포트를 사용하려고 시도했을 수 있습니다.
- 인증서 저장소가 손상된 경우 공통 에이전트 컨테이너 인증서 키를 재생성해야 할 수 있습니다.

해당 문제 해결 방법에 대한 자세한 내용은 [249 페이지](#) “Oracle Solaris OS에서 공통 에이전트 컨테이너 문제 해결”을 참조하십시오.

x86: 시스템 재부트 중 SMF 부트 아카이브 서비스가 실패할 수 있음

Solaris 10 1/06: GRUB 기반 부트 환경에서 시스템 충돌이 발생할 경우 시스템이 재부트될 때 SMF 서비스(`svc:/system/boot-archive:default`)가 실패할 수 있습니다. 이 문제가 발생할 경우 시스템을 재부트하고 GRUB 부트 메뉴에서 비상 안전 조치를 한 아카이브를 선택하십시오. 그런 다음 프롬프트에 따라 부트 아카이브를 재구성하십시오. 아카이브가 재구성되면 시스템을 재부트하십시오. 부트 프로세스를 계속하려는 경우 `svcadm` 명령을 사용하여 `svc:/system/boot-archive:default` 서비스를 지울 수 있습니다. GRUB 기반 부트에 대한 자세한 내용은 [시스템 관리 설명서: 기본 관리의 “GRUB를 사용하여 x86 기반 시스템 부트\(작업 맵\)”](#)를 참조하십시오.

동적 추적 기능

Oracle Solaris DTrace(동적 추적) 기능은 Solaris 커널 및 사용자 프로세스를 새로운 레벨에서 관찰할 수 있도록 해주는 포괄적인 동적 추적 기능입니다. DTrace를 사용하면 OS 커널과 사용자 프로세스가 해당 위치에 지정된 데이터를 기록(프로빙이라고 함)하도록 동적으로 지시할 수 있으므로 시스템을 이해하는 데 도움이 됩니다. 각 프로브는 새로운 D 프로그램 언어로 작성된 사용자 정의 프로그램과 연결될 수 있습니다. 모든 DTrace의 실행은 완전히 동적이며 생산 시스템의 사용자가 사용할 수 있습니다. 자세한 내용은 `dttrace(1M)` 매뉴얼 페이지 및 [Solaris Dynamic Tracing Guide](#)를 참조하십시오.

표준 Solaris 커널 디버거로 kmdb가 kadb를 대체함

표준 "원위치" Solaris 커널 디버거로 kmdb가 kadb를 대체했습니다.

kldb를 사용하면 실제 커널 디버깅에서 mdb의 모든 성능과 기능을 사용할 수 있습니다. kldb는 다음을 지원합니다.

- 디버거 명령(dcmts)
- 디버거 모듈(dmods)
- 커널 유형 데이터에 대한 액세스
- 커널 실행 제어
- 검사
- 수정

자세한 내용은 [kldb\(1\)](#) 매뉴얼 페이지를 참조하십시오. kldb를 사용하여 시스템 문제를 해결하는 단계별 지침은 [시스템 관리 설명서: 기본 관리의 “커널 디버거\(kldb\)를 사용하여 시스템을 부트하는 방법”](#) 및 [시스템 관리 설명서: 기본 관리의 “GRUB 부트 환경에서 커널 디버거\(kldb\)로 시스템을 부트하는 방법”](#)을 참조하십시오.

소프트웨어 문제 해결 작업 검색 위치

문제 해결 작업	자세한 정보
시스템 충돌 정보 관리	17 장, “시스템 충돌 정보 관리(작업)”
코어 파일 관리	16 장, “코어 파일 관리(작업)”
소프트웨어 문제(예: 재부트 실패) 및 백업 문제 해결	18 장, “기타 소프트웨어 문제 해결(작업)”
파일 액세스 문제 해결	19 장, “파일 액세스 문제 해결(작업)”
인쇄 문제 해결	System Administration Guide: Printing 의 13 장, “Troubleshooting Printing Problems in the Oracle Solaris OS (Tasks)”
UFS 파일 시스템 불일치 해결	20 장, “UFS 파일 시스템 불일치 해결(작업)”
소프트웨어 패키지 문제 해결	21 장, “소프트웨어 패키지 문제 해결(작업)”

시스템 충돌 문제 해결

Oracle Solaris OS가 실행되고 있는 시스템이 충돌할 경우 충돌 덤프 파일을 비롯하여 최대한 많은 정보를 서비스 공급자에게 제공하십시오.

시스템이 충돌할 경우 수행할 작업

다음 사항을 반드시 기억해 두어야 합니다.

1. 시스템 콘솔 메시지를 기록해 둡니다.

시스템이 충돌할 경우 시스템이 다시 실행되도록 하는 것이 급선무인 것처럼 여겨질 수 있지만, 시스템을 재부트하기 전에 콘솔 화면에서 메시지를 검사하십시오. 해당 메시지를 통해 충돌 원인을 파악할 수 있습니다. 시스템이 자동으로 재부트되고 콘솔 메시지가 화면에서 사라진 경우에도 시스템 오류 로그(/var/adm/messages 파일)에서 해당 메시지를 확인할 수 있습니다. 시스템 오류 로그 파일 확인에 대한 자세한 내용은 216 페이지 “시스템 메시지 확인 방법”을 참조하십시오.

충돌이 자주 발생하며 충돌 원인을 확인할 수 없는 경우 시스템 콘솔 또는 /var/adm/messages 파일에서 확인할 수 있는 모든 정보를 수집하여 고객 서비스 담당자가 검사할 수 있도록 하십시오. 서비스 공급자에게 제공하기 위해 수집할 문제 해결 정보의 전체 목록은 211 페이지 “시스템 충돌 문제 해결”을 참조하십시오.

시스템 충돌 후 시스템이 제대로 재부트되지 않을 경우 18 장, “기타 소프트웨어 문제 해결(작업)”을 참조하십시오.

2. 디스크를 동기화하고 재부트합니다.

ok sync

시스템 충돌 후 시스템이 제대로 재부트되지 않을 경우 18 장, “기타 소프트웨어 문제 해결(작업)”을 참조하십시오.

시스템 충돌 후 시스템 충돌 덤프가 생성되었는지 여부를 확인하십시오. 시스템 충돌 덤프는 기본적으로 저장됩니다. 충돌 덤프에 대한 자세한 내용은 17 장, “시스템 충돌 정보 관리(작업)”를 참조하십시오.

문제 해결 데이터 수집

다음 질문에 대해 보면서 시스템 문제를 구분할 수 있습니다. 충돌한 시스템에 대한 문제 해결 데이터를 수집하려면 213 페이지 “시스템 충돌 문제 해결 점검 목록”을 사용하십시오.

표 14-1 시스템 충돌 데이터 식별

질문	설명
문제를 재현할 수 있습니까?	재현 가능한 테스트 사례는 어려운 문제를 디버깅하는 데 필요한 경우가 많으므로 문제를 재현할 수 있어야 합니다. 문제를 재현하면 서비스 공급자가 특수한 계층으로 커널을 작성하여 버그를 트리거, 진단 및 수정할 수 있습니다.
타사 드라이버를 사용 중입니까?	드라이버는 모두 동일한 권한을 사용하여 커널과 동일한 주소 공간에서 실행되므로 버그가 있을 경우 시스템 충돌의 원인이 될 수 있습니다.
시스템 충돌 직전에 시스템에서 어떤 작업을 수행했습니까?	시스템에서 특수한 작업을 수행(예: 새로운 스트레스 테스트 실행 또는 평소보다 부하가 높은 작업 실행) 중이었던 경우 이로 인해 충돌이 발생한 것일 수 있습니다.

질문	설명
표 14-1 시스템 충돌 데이터 식별 (계속)	
충돌 직전 특수한 콘솔 메시지가 표시되었습니까?	시스템이 실제로 충돌하기 전에 원인을 나타내는 메시지가 표시되기도 하며, 이 정보는 유용한 경우가 많습니다.
조정 매개변수를 /etc/system 파일에 추가했습니까?	조정 매개변수(예: 시스템이 확보한 것보다 많은 메모리를 할당할 수 있도록 공유 메모리 세그먼트 증가)가 시스템 충돌의 원인일 수 있습니다.
문제가 최근에 시작되었습니까?	최근에 시작된 경우 시스템 변경(예: 새 드라이버, 새 소프트웨어, 다른 작업 부하, CPU 업그레이드 또는 메모리 업그레이드)으로 인해 문제가 나타난 것일 수 있습니다.

시스템 충돌 문제 해결 점검 목록

충돌한 시스템에 대한 시스템 데이터를 수집할 때 다음 점검 목록을 사용하십시오.

항목	데이터
시스템 충돌 덤프를 사용할 수 있습니까?	
운영 체제 릴리스 및 적합한 소프트웨어 응용 프로그램 릴리스 레벨을 식별하십시오.	
시스템 하드웨어를 식별하십시오.	
sun4u 시스템의 경우 prtdiag 출력을 포함시키고, 다른 시스템의 경우 Explorer 출력을 포함시키십시오.	
패치가 설치되었습니까? 설치된 경우 showrev -p 출력을 포함시키십시오.	
문제를 재현할 수 있습니까?	
시스템에 타사 드라이버가 설치되었습니까?	
시스템 충돌 전에 시스템에서 어떤 작업을 수행했습니까?	
시스템 충돌 직전 특수한 콘솔 메시지가 표시되었습니까?	
매개변수를 /etc/system 파일에 추가했습니까?	
문제가 최근에 시작되었습니까?	

시스템 메시지 관리

이 장에서는 Oracle Solaris OS의 시스템 메시지 기능에 대해 설명합니다.

시스템 메시지 확인

시스템 메시지는 콘솔 장치에 표시됩니다. 대부분의 시스템 메시지 텍스트는 다음과 같이 표시됩니다.

[ID *msgid facility.priority*]

예를 들어, 다음과 같습니다.

```
[ID 672855 kern.notice] syncing file systems...
```

메시지가 커널에서 시작된 경우 커널 모듈 이름이 표시됩니다. 예를 들어, 다음과 같습니다.

```
Oct 1 14:07:24 mars ufs: [ID 845546 kern.notice] alloc: /: file system full
```

시스템이 충돌하면 시스템 콘솔에 다음과 같은 메시지가 표시될 수 있습니다.

```
panic: error message
```

가끔 비상 메시지 대신 다음 메시지가 표시될 수도 있습니다.

```
Watchdog reset !
```

오류 로깅 데몬(`syslogd`)이 자동으로 메시지 파일에 다양한 시스템 경고 및 오류를 기록합니다. 기본적으로 이러한 시스템 메시지는 시스템 콘솔에 표시되고 `/var/adm` 디렉토리에 저장되는 경우가 많습니다. 시스템 메시지 로깅을 설정하여 이러한 메시지가 저장되는 위치를 지정할 수 있습니다. 자세한 내용은 [218 페이지 “시스템 메시지 로깅 사용자 정의”](#)를 참조하십시오. 이러한 메시지는 시스템 문제(예: 실패 예상 장치)를 알리는 것입니다.

/var/adm 디렉토리에는 여러 메시지 파일이 들어 있습니다. 가장 최근 메시지는 /var/adm/messages 파일(및 messages.*)에 있으며 가장 오래된 메시지는 messages.3 파일에 있습니다. 특정 기간이 경과되면(일반적으로 10일마다) 새 messages 파일이 생성됩니다. messages.0 파일의 이름은 messages.1로, messages.1의 이름은 messages.2로, messages.2의 이름은 messages.3으로 바뀝니다. 현재 /var/adm/messages.3 파일은 삭제됩니다.

/var/adm 디렉토리에는 메시지, 충돌 덤프 및 기타 데이터를 포함하는 큰 파일이 저장되므로 이 디렉토리는 많은 양의 디스크 공간을 사용할 수 있습니다. /var/adm 디렉토리가 너무 커지지 않도록 하고 다음 충돌 덤프가 저장될 수 있도록 하려면 주기적으로 불필요한 파일을 제거해야 합니다. crontab 파일을 사용하여 이 작업을 자동화할 수 있습니다. 해당 작업 자동화에 대한 자세한 내용은 88 페이지 “충돌 덤프 파일 삭제 방법” 및 8 장, “시스템 작업 예약(작업)”를 참조하십시오.

▼ 시스템 메시지 확인 방법

- dmesg 명령을 사용하여 시스템 충돌 또는 재부트로 생성된 최근 메시지를 표시합니다.

```
$ dmesg
```

또는 more 명령을 사용하여 메시지 화면을 한 번에 하나씩 표시합니다.

```
$ more /var/adm/messages
```

예 15-1 시스템 메시지 확인

다음 예에서는 dmesg 명령의 출력을 보여 줍니다.

```
$ dmesg
Jan 3 08:44:41 starbug genunix: [ID 540533 kern.notice] SunOS Release 5.10 ...
Jan 3 08:44:41 starbug genunix: [ID 913631 kern.notice] Copyright 1983-2003 ...
Jan 3 08:44:41 starbug genunix: [ID 678236 kern.info] Ethernet address ...
Jan 3 08:44:41 starbug unix: [ID 389951 kern.info] mem = 131072K (0x8000000)
Jan 3 08:44:41 starbug unix: [ID 930857 kern.info] avail mem = 121888768
Jan 3 08:44:41 starbug rootnexus: [ID 466748 kern.info] root nexus = Sun Ultra 5/
10 UPA/PCI (UltraSPARC-IIi 333MHz)
Jan 3 08:44:41 starbug rootnexus: [ID 349649 kern.info] pcipsy0 at root: UPA 0x1f0x0
Jan 3 08:44:41 starbug genunix: [ID 936769 kern.info] pcipsy0 is /pci@1f,0
Jan 3 08:44:41 starbug pcipsy: [ID 370704 kern.info] PCI-device: pci@1,1, simba0
Jan 3 08:44:41 starbug genunix: [ID 936769 kern.info] simba0 is /pci@1f,0/pci@1,1
Jan 3 08:44:41 starbug pcipsy: [ID 370704 kern.info] PCI-device: pci@1, simba1
Jan 3 08:44:41 starbug genunix: [ID 936769 kern.info] simba1 is /pci@1f,0/pci@1
Jan 3 08:44:57 starbug simba: [ID 370704 kern.info] PCI-device: ide@3, uata0
Jan 3 08:44:57 starbug genunix: [ID 936769 kern.info] uata0 is /pci@1f,0/pci@1,
1/ide@3
Jan 3 08:44:57 starbug uata: [ID 114370 kern.info] dad0 at pci1095,6460
.
.
.
```

참조 자세한 내용은 [dmesg\(1M\)](#) 매뉴얼 페이지를 참조하십시오.

시스템 로그 교체

시스템 로그 파일을 교체하려면 루트 crontab 파일의 항목에서 `logadm` 명령을 사용하십시오. `/usr/lib/newsyslog` 스크립트는 더 이상 사용되지 않습니다.

시스템 로그 교체는 `/etc/logadm.conf` 파일에서 정의됩니다. 이 파일에는 `syslogd` 등의 프로세스에 대한 로그 교체 항목이 들어 있습니다. 예를 들어, `/etc/logadm.conf` 파일의 한 항목은 파일이 비어 있지 않은 경우 `/var/log/syslog` 파일이 매주 교체되도록 지정합니다. 가장 최근 `syslog` 파일은 `syslog.0`이 되고 다음 번 가장 최근 파일은 `syslog.1`이 됩니다. 여덟 개의 이전 `syslog` 로그 파일이 보관됩니다.

`/etc/logadm.conf` 파일에는 마지막 로그 교체가 발생한 시간 기록도 포함되어 있습니다.

`logadm` 명령을 사용하여 시스템 로깅을 사용자 정의하고 필요에 따라 `/etc/logadm.conf` 파일에서 로깅을 더 추가할 수 있습니다.

예를 들어, Apache 액세스 및 오류 로그를 교체하려면 다음 명령을 사용하십시오.

```
# logadm -w /var/apache/logs/access_log -s 100m
# logadm -w /var/apache/logs/error_log -s 10m
```

이 예에서 Apache `access_log` 파일은 크기가 100MB에 도달할 때 `.0`, `.1` 등의 접미어를 사용하여 교체되며, 열 개의 이전 `access_log` 파일 복사본이 보관됩니다. `error_log`는 크기가 10MB에 도달할 때 `access_log` 파일과 동일한 접미어 및 복사본 수를 사용하여 교체됩니다.

앞선 Apache 로그 교체 예에 대한 `/etc/logadm.conf` 항목은 다음과 유사하게 표시됩니다.

```
# cat /etc/logadm.conf
.
.
.
/var/apache/logs/error_log -s 10m
/var/apache/logs/access_log -s 100m
```

자세한 내용은 [logadm\(1M\)](#)을 참조하십시오.

수퍼유저 또는 동등한 역할(로그 관리 권한이 있는 역할)의 사용자로 `logadm` 명령을 사용할 수 있습니다. RBAC(역할 기반 액세스 제어)를 사용할 경우 `logadm` 명령에 대한 액세스 권한을 제공하여 비루트 사용자에게 로그 파일 유지 관리 권한을 부여할 수 있습니다.

예를 들어, 사용자 `andy`에게 `logadm` 명령을 사용할 수 있는 권한을 부여하려면 `/etc/user_attr` 파일에 다음 항목을 추가하십시오.

```
andy::::profiles=Log Management
```

또는 Solaris Management Console을 사용하여 로그 관리에 대한 역할을 설정할 수 있습니다. 역할 설정에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “[Role-Based Access Control \(Overview\)](#)”을 참조하십시오.

표 15-1 syslog.conf 메시지의 소스 기능

소스	설명
kern	커널
auth	인증
daemon	모든 데몬
mail	메일 시스템
lp	스풀링 시스템
user	사용자 프로세스

주 - /etc/syslog.conf 파일에서 활성화할 수 있는 syslog 기능 수는 무제한입니다.

표 15-2 syslog.conf 메시지의 우선 순위 레벨

우선 순위	설명
emerg	시스템 긴급
alert	즉각적인 수정이 필요한 오류
crit	심각한 오류
err	기타 오류
info	정보 메시지
debug	디버깅에 사용되는 출력
none	이 설정은 출력을 기록하지 않습니다.

▼ 시스템 메시지 로깅 사용자 정의 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 /etc/syslog.conf 파일을 편집하여 [syslog.conf\(4\)](#)에 설명된 구문에 따라 메시지 소스, 우선 순위 및 메시지 위치를 추가하거나 변경합니다.
- 3 변경 내용을 저장하여 파일을 종료합니다.

예 15-2 시스템 메시지 로깅 사용자 정의

이 `/etc/syslog.conf` `user.emerg` 기능 샘플은 사용자 긴급 메시지를 루트 및 개별 사용자에게 전송합니다.

```
user.emerg                                'root, *'
```

원격 콘솔 메시지를 사용으로 설정

다음과 같은 새로운 콘솔 기능을 통해 원격 시스템 문제 해결 성능이 향상되었습니다.

- `consadm` 명령을 사용하여 직렬 장치를 **보조**(또는 원격) 콘솔로 선택할 수 있습니다. 시스템 관리자는 `consadm` 명령을 사용하여 시스템이 실행 레벨 간에 전환될 때 `sulogin` 세션을 호스트하고 재지정된 콘솔 메시지를 표시하도록 직렬 포트를 하나 이상 구성할 수 있습니다. 이 기능을 사용하면 모뎀을 이용한 전화 접속을 통해 직렬 포트에 연결하여 콘솔 메시지를 모니터링하고 `init` 상태 전환에 참여할 수 있습니다. 자세한 내용은 `sulogin(1M)` 및 뒷부분의 단계별 절차를 참조하십시오.

보조 콘솔로 구성된 포트를 사용하여 시스템에 로그인할 수 있는 상태에서는 출력 장치에 표시되는 정보가 기본 콘솔에도 표시됩니다. 부트스크립트 또는 기타 응용 프로그램이 기본 콘솔에 대한 읽기 및 쓰기를 수행하는 경우 쓰기 출력은 모든 보조 콘솔에 표시되지만 입력 읽기는 기본 콘솔에서만 수행됩니다. 대화형 로그인 세션 중 `consadm` 명령을 사용하는 방법은 222 페이지 “대화형 로그인 세션 중 `consadm` 명령 사용”을 참조하십시오.

- 이제 콘솔 출력은 커널 및 새 의사 장치(`/dev/sysmsg`)에 기록된 `syslog` 메시지로 구성됩니다. 또한 `rc` 스크립트 시작 메시지가 `/dev/msglog`에 기록됩니다. 이전에는 이러한 메시지가 모두 `/dev/console`에 기록되었습니다.

보조 콘솔에 표시되는 스크립트 메시지를 확인하려면 콘솔 출력을 `/dev/console`로 전달하는 스크립트를 `/dev/msglog`로 변경해야 합니다. 메시지가 보조 장치로 재지정되도록 하려면 `/dev/console`을 참조하는 프로그램을 명시적으로 수정하여 `syslog()` 또는 `strlog()`를 사용하도록 해야 합니다.

- `consadm` 명령은 데몬을 실행하여 보조 콘솔 장치를 모니터링합니다. 보조 콘솔로 지정된 디스플레이 장치(반송과 연결 해제, 정지 또는 손실)가 보조 콘솔 장치 목록에서 제거되고 더 이상 활성화되지 않습니다. 하나 이상의 보조 콘솔을 사용으로 설정하면 기본 콘솔에 메시지가 표시되지 않습니다. `/dev/console`에는 메시지가 계속 표시됩니다.

실행 레벨 전환 중 보조 콘솔 메시지 사용

실행 레벨 전환 중 보조 콘솔 메시지를 사용할 때는 다음 사항을 염두에 두십시오.

- 시스템이 부트될 때 실행되는 rc 스크립트에 대한 사용자 입력이 필요한 경우 보조 콘솔에서 입력을 가져올 수 없습니다. 입력은 기본 콘솔에서 가져와야 합니다.
- 실행 레벨 간의 전환 시 슈퍼유저 암호에 대한 프롬프트를 표시하기 위해 `init`에 의해 호출되는 `sulogin` 프로그램이 슈퍼유저 암호 프롬프트를 기본 콘솔 장치와 각 보조 장치에 전송하도록 수정되었습니다.
- 시스템이 단일 사용자 모드이며 `consadm` 명령을 통해 하나 이상의 보조 콘솔이 사용으로 설정된 경우 올바른 슈퍼유저 암호를 `sulogin` 프롬프트에 제공하기 위해 첫번째 장치에서 콘솔 로그인 세션이 실행됩니다. 콘솔 장치에서 올바른 암호가 수신되면 `sulogin`이 기타 모든 콘솔 장치의 입력을 사용 안함으로 설정합니다.
- 콘솔 중 하나가 단일 사용자 권한을 사용하는 경우 기본 콘솔 및 기타 보조 콘솔에 메시지가 표시됩니다. 이 메시지는 올바른 슈퍼유저 암호를 승인하여 특정 장치가 콘솔로 지정되었음을 나타냅니다. 단일 사용자 셸이 실행되는 보조 콘솔에서 반송과 손실이 있을 경우 다음 두 가지 작업 중 하나가 발생할 수 있습니다.
 - 보조 콘솔이 실행 레벨 1의 시스템을 나타내는 경우 시스템이 계속 기본 실행 레벨로 실행됩니다.
 - 보조 콘솔이 실행 레벨 S의 시스템을 나타내는 경우 셸에서 `init s` 또는 `shutdown` 명령이 입력된 장치에 `ENTER RUN LEVEL (0-6, s or S):` 메시지가 표시됩니다. 해당 장치에 반송과가 없을 경우 반송과를 재설정하고 올바른 실행 레벨을 입력해야 합니다. `init` 또는 `shutdown` 명령은 실행 레벨 프롬프트를 다시 표시하지 않습니다.
- 직렬 포트를 사용하여 시스템에 로그인하고 다른 실행 레벨로의 전환을 위해 `init` 또는 `shutdown` 명령이 실행된 경우 이 장치가 보조 콘솔인지 여부에 관계없이 로그인 세션이 끊깁니다. 보조 콘솔 기능이 없는 릴리스에서도 이와 동일한 상황이 발생합니다.
- `consadm` 명령을 사용하여 보조 콘솔로 선택된 장치는 시스템이 재부트되거나 보조 콘솔의 선택이 해제될 때까지 보조 콘솔로 유지됩니다. 단, `consadm` 명령에는 시스템 재부트 시 장치를 보조 콘솔로 설정할 수 있는 옵션이 포함되어 있습니다. 단계별 지침은 뒷부분의 절차를 참조하십시오.

대화형 로그인 세션 중 `consadm` 명령 사용

직렬 포트에 연결된 터미널을 사용하여 시스템에 로그인한 후 `consadm` 명령을 사용하여 터미널의 콘솔 메시지를 확인하는 방식으로 대화형 로그인 세션을 실행하려면 다음 동작에 유의하십시오.

- 보조 콘솔이 활성화된 상태에서 대화형 로그인 세션에 터미널을 사용하면 콘솔 메시지가 `/dev/sysmsg` 또는 `/dev/msglog` 장치로 전송됩니다.
- 터미널에서 명령을 실행하는 동안에는 입력이 기본 콘솔(`/dev/console`)이 아닌 대화형 세션으로 전달됩니다.
- `init` 명령을 실행하여 실행 레벨을 변경하면 원격 콘솔 소프트웨어가 대화형 세션을 강제 종료하고 `sulogin` 프로그램을 실행합니다. 이 단계에서 입력은 터미널에서 가져온 것만 승인되고 콘솔 장치에서 가져온 것처럼 처리됩니다. 따라서 221 페이지 “실행 레벨 전환 중 보조 콘솔 메시지 사용”의 설명에 따라 `sulogin` 프로그램에 암호를 입력할 수 있습니다.

그런 다음 (보조) 터미널에서 올바른 암호를 입력하면 보조 콘솔이 대화형 `sulogin` 세션을 실행하고 기본 콘솔 및 기타 보조 콘솔을 잠급니다. 즉, 터미널이 시스템 콘솔로 작동합니다.

- 여기서 실행 레벨 3으로 변경하거나 다른 실행 레벨로 이동할 수 있습니다. 실행 레벨을 변경하면 `sulogin`이 다시 모든 콘솔 장치에서 실행됩니다. 종료하거나 시스템이 실행 레벨 3에 도달하도록 지정하면 모든 보조 콘솔의 입력 제공 기능이 손실됩니다. 해당 보조 콘솔은 콘솔 메시지용 디스플레이 장치로 되돌아갑니다.

시스템이 시작되면 기본 콘솔 장치에서 `rc` 스크립트에 정보를 제공해야 합니다. 시스템이 다시 시작되면 `login` 프로그램이 직렬 포트에서 실행되므로 다른 대화형 세션에 다시 로그인할 수 있습니다. 장치를 보조 콘솔로 지정한 경우 터미널의 콘솔 메시지는 계속 표시되지만 터미널의 모든 입력은 대화형 세션으로 전달됩니다.

▼ 보조(원격) 콘솔을 사용으로 설정하는 방법

`consadm` 데몬은 `consadm` 명령을 통해 보조 콘솔이 추가되기 전까지 포트 모니터링을 시작하지 않습니다. 보안 기능으로 콘솔 메시지는 반송파가 끊기거나 보조 콘솔 장치의 선택이 해제될 때까지만 재지정됩니다. 따라서 포트에서 반송파를 설정해야만 `consadm` 명령을 성공적으로 사용할 수 있습니다.

보조 콘솔을 사용으로 설정하는 방법은 `consadm(1m)` 매뉴얼 페이지를 참조하십시오.

- 1 슈퍼유저로 시스템에 로그인합니다.
- 2 보조 콘솔을 사용으로 설정합니다.
`consadm -a devicename`
- 3 현재 연결이 보조 콘솔인지 확인합니다.
`consadm`

예 15-3 보조(원격) 콘솔을 사용하여 설정

```
# consadm -a /dev/term/a
# consadm
/dev/term/a
```

▼ 보조 콘솔 목록 표시 방법

- 1 슈퍼유저로 시스템에 로그인합니다.
- 2 다음 단계 중 하나를 선택합니다.
 - a. 보조 콘솔 목록을 표시합니다.

```
# consadm
/dev/term/a
```

- b. 영구 보조 콘솔 목록을 표시합니다.

```
# consadm -p
/dev/term/b
```

▼ 시스템 재부트 시 보조(원격) 콘솔을 사용하여 설정하는 방법

- 1 슈퍼유저로 시스템에 로그인합니다.
- 2 시스템 재부트 시 보조 콘솔을 사용하여 설정합니다.


```
# consadm -a -p devicename
```

 그러면 장치가 영구 보조 콘솔 목록에 추가됩니다.
- 3 장치가 영구 보조 콘솔 목록에 추가되었는지 확인합니다.

```
# consadm
```

예 15-4 시스템 재부트 시 보조(원격) 콘솔을 사용하여 설정

```
# consadm -a -p /dev/term/a
# consadm
/dev/term/a
```

▼ 보조(원격) 콘솔을 사용 안함으로 설정하는 방법

- 1 슈퍼유저로 시스템에 로그인합니다.
- 2 다음 단계 중 하나를 선택합니다.
 - a. 보조 콘솔을 사용 안함으로 설정합니다.

```
# consadm -d devicename
```

또는
 - b. 보조 콘솔을 사용 안함으로 설정하고 영구 보조 콘솔 목록에서 제거합니다.

```
# consadm -p -d devicename
```
- 3 보조 콘솔이 사용 안함으로 설정되었는지 확인합니다.

```
# consadm
```

예 15-5 보조(원격) 콘솔을 사용 안함으로 설정

```
# consadm -d /dev/term/a  
# consadm
```

코어 파일 관리(작업)

이 장에서는 `coreadm` 명령을 사용하여 코어 파일을 관리하는 방법에 대해 설명합니다.

코어 파일 관리와 관련된 절차에 대한 자세한 내용은 225 페이지 “코어 파일 관리(작업 맵)”를 참조하십시오.

코어 파일 관리(작업 맵)

작업	설명	수행 방법
1. 현재 코어 덤프 구성을 표시합니다.	<code>coreadm</code> 명령을 사용하여 현재 코어 덤프 구성을 표시합니다.	228 페이지 “현재 코어 덤프 구성 표시 방법”
2. 코어 덤프 구성을 수정합니다.	코어 덤프 구성을 수정하여 다음 작업 중 하나를 수행합니다. 코어 파일 이름 패턴을 설정합니다. 프로세스별 코어 파일 경로를 사용으로 설정합니다. 전역 코어 파일 경로를 사용으로 설정합니다.	229 페이지 “코어 파일 이름 패턴 설정 방법” 229 페이지 “프로세스별 코어 파일 경로를 사용으로 설정하는 방법” 229 페이지 “전역 코어 파일 경로를 사용으로 설정하는 방법”
3. 코어 덤프 파일을 검사합니다.	<code>proc</code> 도구를 사용하여 코어 덤프 파일을 확인합니다.	230 페이지 “코어 파일 검사”

코어 파일 관리 개요

코어 파일은 프로세스 또는 응용 프로그램이 비정상적으로 종료될 때 생성됩니다. 코어 파일은 `coreadm` 명령으로 관리됩니다.

예를 들어, `coreadm` 명령을 사용하여 모든 프로세스 코어 파일이 단일 시스템 디렉토리에 지정되도록 시스템을 구성할 수 있습니다. 즉, 프로세스 또는 데몬이 비정상적으로 종료될 때마다 특정 디렉토리의 코어 파일을 검사하여 보다 간편하게 문제를 추적할 수 있습니다.

구성 가능한 코어 파일 경로

상호 독립적으로 사용/사용 안함으로 설정할 수 있는 두 개의 구성 가능한 새 코어 파일 경로는 다음과 같습니다.

- 프로세스별 코어 파일 경로: 기본값은 `core`이며 기본적으로 사용으로 설정되어 있습니다. 사용으로 설정할 경우 프로세스가 비정상적으로 종료될 때 프로세스별 코어 파일 경로로 인해 코어 파일이 생성됩니다. 새 프로세스는 상위 프로세스에서 프로세스별 경로를 상속합니다.

생성되는 프로세스별 코어 파일은 소유자 액세스에 대해 읽기/쓰기 권한을 가지는 프로세스 소유자가 소유합니다. 소유 사용자만 이 파일을 볼 수 있습니다.

- 전역 코어 파일 경로: 기본값은 `core`이며 기본적으로 사용 안함으로 설정되어 있습니다. 사용으로 설정할 경우 전역 코어 파일 경로를 사용하여 프로세스별 코어 파일과 콘텐츠가 동일한 추가 코어 파일이 생성됩니다.

생성되는 전역 코어 파일은 슈퍼유저 액세스에 대해서만 읽기/쓰기 권한을 가지는 슈퍼유저가 소유합니다. 권한이 없는 사용자는 이 파일을 볼 수 없습니다.

프로세스가 비정상적으로 종료될 때 기본적으로 현재 디렉토리에 코어 파일이 생성됩니다. 전역 코어 파일 경로를 사용으로 설정할 경우 프로세스가 비정상적으로 종료될 때마다 현재 작업 디렉토리와 전역 코어 파일 위치에 파일이 하나씩 생성될 수 있습니다.

기본적으로 `setuid` 프로세스는 전역 또는 프로세스별 경로를 사용하여 코어 파일을 생성하지 않습니다.

확장된 코어 파일 이름

전역 코어 파일 디렉토리를 사용으로 설정할 경우 다음 표의 설명에 따라 변수를 사용하여 코어 파일을 서로 구별할 수 있습니다.

변수 이름	변수 정의
%d	최대 MAXPATHLEN자의 실행 파일 디렉토리 이름
%f	최대 MAXCOMLEN자의 실행 파일 이름
%g	유효 그룹 ID
%m	시스템 이름(uname -m)
%n	시스템 노드 이름(uname -n)
%p	프로세스 ID
%t	시간의 십진수 값(2)
%u	유효 사용자 ID
%z	프로세스가 실행된 영역의 이름(zonename)
%%	리터럴 %

예를 들어, 전역 코어 파일 경로가 다음으로 설정된 경우

```
/var/core/core.%f.%p
```

PID가 12345인 sendmail 프로세스가 비정상적으로 종료되면 다음과 같은 코어 파일이 생성됩니다.

```
/var/core/core.sendmail.12345
```

코어 파일 이름 패턴 설정

전역, 영역 또는 프로세스별 기준으로 코어 파일 이름 패턴을 설정할 수 있습니다. 또한 시스템 재부트 시 지속되는 프로세스별 기본값을 설정할 수 있습니다.

예를 들어, 다음 `coreadm` 명령은 기본 프로세스별 코어 파일 패턴을 설정합니다. 이 설정은 기본 코어 파일 패턴을 명시적으로 무시하지 않은 모든 프로세스에 적용됩니다. 이 설정은 시스템 재부트 시 지속됩니다.

```
# coreadm -i /var/core/core.%f.%p
```

다음 `coreadm` 명령은 모든 프로세스에 대해 프로세스별 코어 파일 이름 패턴을 설정합니다.

```
$ coreadm -p /var/core/core.%f.%p $$
```

\$\$ 기호는 현재 실행 중인 셸의 프로세스 ID에 대한 위치 표시자입니다. 모든 하위 프로세스는 프로세스별 코어 파일 이름 패턴을 상속합니다.

전역 또는 프로세스별 코어 파일 이름 패턴을 설정한 후에는 `coreadm -e` 명령을 통해 사용으로 설정해야 합니다. 자세한 내용은 다음 절차를 참조하십시오.

사용자의 `$HOME/.profile` 또는 `.login` 파일에 명령을 삽입하여 사용자의 로그인 세션 중 실행되는 모든 프로세스에 대해 코어 파일 이름 패턴을 설정할 수 있습니다.

코어 파일을 생성하도록 `setuid` 프로그램을 사용으로 설정

`coreadm` 명령으로 다음 경로를 설정하여 모든 시스템 프로세스에 대해 코어 파일을 생성하거나 프로세스별로 코어 파일을 생성하도록 `setuid` 프로그램을 사용/사용 안함으로 설정할 수 있습니다.

- 전역 `setuid` 옵션을 사용으로 설정할 경우 전역 코어 파일 경로를 통해 시스템의 모든 `setuid` 프로그램이 코어 파일을 생성할 수 있습니다.
- 프로세스별 `setuid` 옵션을 사용으로 설정할 경우 프로세스별 코어 파일 경로를 통해 특정 `setuid` 프로세스가 코어 파일을 생성할 수 있습니다.

기본적으로 두 플래그는 사용 안함으로 설정되어 있습니다. 보안상 전역 코어 파일 경로는 `/`로 시작하는 전체 경로 이름이어야 합니다. 슈퍼유저가 프로세스별 코어 파일을 사용 안함으로 설정할 경우 개별 사용자가 코어 파일을 얻을 수 없습니다.

`setuid` 코어 파일은 슈퍼유저 액세스에 대해서만 읽기/쓰기 권한을 가지는 슈퍼유저가 소유합니다. 일반 사용자가 `setuid` 코어 파일을 생성한 프로세스를 소유한 경우에도 일반 사용자는 해당 파일에 액세스할 수 없습니다.

자세한 내용은 `coreadm(1M)` 매뉴얼 페이지를 참조하십시오.

현재 코어 덤프 구성 표시 방법

옵션 없이 `coreadm` 명령을 사용하여 현재 코어 덤프 구성을 표시합니다.

```
$ coreadm
    global core file pattern:
global core file content: default
    init core file pattern: core
    init core file content: default
    global core dumps: disabled
    per-process core dumps: enabled
    global setid core dumps: disabled
per-process setid core dumps: disabled
    global core dump logging: disabled
```

▼ 코어 파일 이름 패턴 설정 방법

- 프로세스별 코어 파일을 설정할지 아니면 전역 코어 파일을 설정할지 여부를 결정하고 다음 작업 중 하나를 선택합니다.

- a. 프로세스별 파일 이름 패턴을 설정합니다.

```
$ coreadm -p $HOME/corefiles/%f.%p $$
```

- b. 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- c. 전역 파일 이름 패턴을 설정합니다.

```
# coreadm -g /var/corefiles/%f.%p
```

▼ 프로세스별 코어 파일 경로를 사용으로 설정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 프로세스별 코어 파일 경로를 사용으로 설정합니다.

```
# coreadm -e process
```

- 3 현재 프로세스 코어 파일 경로를 표시하여 구성을 확인합니다.

```
$ coreadm $$
1180: /home/kryten/corefiles/%f.%p
```

▼ 전역 코어 파일 경로를 사용으로 설정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 전역 코어 파일 경로를 사용으로 설정합니다.

```
# coreadm -e global -g /var/core/core.%f.%p
```

3 현재 프로세스 코어 파일 경로를 표시하여 구성을 확인합니다.

```
# coreadm
  global core file pattern: /var/core/core.%f.%p
  global core file content: default
  init core file pattern: core
  init core file content: default
  global core dumps: enabled
  per-process core dumps: enabled
  global setid core dumps: disabled
  per-process setid core dumps: disabled
  global core dump logging: disabled
```

코어 파일 문제 해결

오류 메시지

```
NOTICE: 'set allow_setid_core = 1' in /etc/system is obsolete
NOTICE: Use the coreadm command instead of 'allow_setid_core'
```

원인

/etc/system 파일에서 setuid 코어 파일을 허용하는 오래된 매개변수가 있습니다.

해결 방법

allow_setid_core=1을 /etc/system 파일에서 제거하십시오. 그런 다음 coreadm 명령을 사용하여 전역 setuid 코어 파일 경로를 사용으로 설정하십시오.

코어 파일 검사

일부 proc 도구가 프로세스 코어 파일과 실시간 프로세스를 검사할 수 있도록 향상되었습니다. proc 도구는 /proc 파일 시스템의 기능을 조작할 수 있는 유틸리티입니다.

이제 해당 명령에 대한 프로세스 ID를 지정하는 것과 유사한 방법으로 명령줄에 코어 파일 이름을 지정하여 /usr/proc/bin/pstack, pmap, pldd, pflags 및 pcred 도구를 코어 파일에 적용할 수 있습니다.

proc 도구로 코어 파일을 검사하는 방법은 [proc\(1\)](#)를 참조하십시오.

예 16-1 proc 도구로 코어 파일 검사

```
$ ./a.out
Segmentation Fault(coredump)
$ /usr/proc/bin/pstack ./core
core './core' of 19305: ./a.out
 000108c4 main      (1, ffbef5cc, ffbef5d4, 20800, 0, 0) + 1c
 00010880 _start    (0, 0, 0, 0, 0, 0) + b8
```

시스템 충돌 정보 관리(작업)

이 장에서는 Oracle Solaris OS에서 시스템 충돌 정보를 관리하는 방법에 대해 설명합니다.

시스템 충돌 정보 관리와 관련된 절차에 대한 자세한 내용은 232 페이지 “시스템 충돌 정보 관리(작업 맵)”를 참조하십시오.

시스템 충돌 정보 관리의 새로운 기능

이 단원에서는 Oracle Solaris 릴리스에서 시스템 리소스를 관리하기 위해 새로 추가되거나 변경된 기능에 대해 설명합니다.

빠른 충돌 덤프 기능

Oracle Solaris 10 9/10: 이 기능 향상을 통해 시스템은 보다 짧은 시간 안에 보다 적은 공간을 사용하여 충돌 덤프를 저장할 수 있게 되었습니다. 이제 충돌 덤프를 완료하는 데 필요한 시간은 플랫폼에 따라 2-10배 더 빨라졌습니다. 충돌 덤프를 `savecore` 디렉토리에 저장하는 데 필요한 디스크 공간의 양도 이와 같이 2-10배 줄어들었습니다. 충돌 덤프 파일의 생성 및 압축을 가속화하기 위해 빠른 충돌 덤프 기능은 대형 시스템에서 많이 사용되고 있지 않은 CPU를 활용합니다. 새 충돌 덤프 파일인 `vmdump.n`은 `vmcore.n` 및 `unix.n` 파일의 압축 버전입니다. 압축된 충돌 덤프는 네트워크에서 더욱 신속하게 이동한 다음 오프사이트에서 분석할 수 있습니다. `mdb` 유틸리티와 같은 도구에서 사용하려면 먼저 덤프 파일의 압축을 풀어야 합니다. 로컬 또는 원격으로 `savecore` 명령을 사용하여 덤프 파일의 압축을 풀 수 있습니다.

새 충돌 덤프 기능을 지원하기 위해 `-z` 옵션이 `dumpadm` 명령에 추가되었습니다. 이 옵션을 사용하여 덤프를 압축된 형식으로 저장할지 또는 압축되지 않은 형식으로 저장할지 지정할 수 있습니다. 기본 형식은 압축된 형식입니다.

자세한 내용은 `dumpadm(1M)` 및 `savecore(1M)` 매뉴얼 페이지를 참조하십시오.

시스템 충돌 정보 관리(작업 맵)

다음 작업 맵에서는 시스템 충돌 정보 관리에 필요한 절차를 식별합니다.

작업	설명	수행 방법
1. 현재 충돌 덤프 구성을 표시합니다.	dumpadm 명령을 사용하여 현재 충돌 덤프 구성을 표시합니다.	236 페이지 “현재 충돌 덤프 구성 표시 방법”
2. 충돌 덤프 구성을 수정합니다.	dumpadm 명령을 사용하여 덤프할 데이터의 유형, 시스템에서 전용 덤프 장치를 사용할지 여부, 충돌 덤프 파일을 저장할 디렉토리 및 충돌 덤프 파일 작성 후 사용 가능한 상태로 유지되어야 할 공간을 지정할 수 있습니다.	236 페이지 “충돌 덤프 구성 수정 방법”
3. 충돌 덤프 파일을 검사합니다.	mdb 명령을 사용하여 충돌 덤프 파일을 확인합니다.	238 페이지 “충돌 덤프 검사 방법”
4. (선택 사항) 전체 충돌 덤프 디렉토리에서 복구합니다.	이 시나리오에서는 시스템이 충돌하지만 savecore 디렉토리에 사용 가능한 공간이 없으며 중요한 시스템 충돌 덤프 정보를 저장하고자 합니다.	239 페이지 “전체 충돌 덤프 디렉토리에서 복구 방법(선택 사항)”
5. (선택 사항) 충돌 덤프 파일 저장을 사용/사용 안함으로 설정합니다.	dumpadm 명령을 사용하여 충돌 덤프 파일 저장을 사용/사용 안함으로 설정합니다. 기본적으로 충돌 덤프 파일 저장은 사용으로 설정되어 있습니다.	239 페이지 “충돌 덤프 저장을 사용/사용 안함으로 설정하는 방법”

시스템 충돌(개요)

시스템 충돌은 하드웨어 오작동, I/O 문제 및 소프트웨어 오류로 인해 발생할 수 있습니다. 시스템이 충돌하면 콘솔에 오류 메시지가 표시되고 덤프 장치에 물리적 메모리 복사본이 기록됩니다. 그런 다음 시스템이 자동으로 재부트됩니다. 시스템이 재부트될 때 savecore 명령이 실행되어 덤프 장치에서 데이터가 검색되고 저장된 충돌 덤프가 savecore 디렉토리에 기록됩니다. 저장된 충돌 덤프 파일은 지원 공급자에게 문제 진단에 도움이 되는 유용한 정보를 제공합니다.

충돌 덤프 정보는 vmdump.*n* 파일에 압축된 형식으로 기록됩니다. 여기서 *n*은 충돌 덤프를 식별하는 정수입니다. 나중에 동일한 시스템 또는 다른 시스템에서 savecore 명령을 호출하여 압축된 충돌 덤프를 이름이 unix.*n* 및 vmcore.*n*인 파일 쌍으로 확장할 수 있습니다. dumpadm 명령을 사용하여 재부트 시 충돌 덤프가 저장되는 디렉토리를 구성할 수도 있습니다.

UFS 루트 파일 시스템이 설치된 시스템의 경우 기본 덤프 장치는 해당 스왑 분할 영역으로 구성됩니다. 스왑 분할 영역은 운영 체제용 저장소를 지원하는 가상 메모리로 예약된 디스크 분할 영역입니다. 따라서 스왑 영역에는 충돌 덤프가 겹쳐 쓸 영구적인

정보가 상주하지 않습니다. Oracle Solaris ZFS 루트 파일 시스템이 설치된 시스템의 경우 전용 ZFS 볼륨이 스왑 및 덤프 영역에 사용됩니다. 자세한 내용은 233 페이지 “스왑 영역 및 덤프 장치에 대한 Oracle Solaris ZFS 지원”을 참조하십시오.

스왑 영역 및 덤프 장치에 대한 Oracle Solaris ZFS 지원

Oracle Solaris ZFS 루트 파일 시스템을 설치하거나 Oracle Solaris Live Upgrade 프로그램을 사용하여 UFS 루트 파일 시스템에서 ZFS 루트 파일 시스템으로 마이그레이션하면 스왑 및 덤프 장치가 2개의 ZFS 볼륨에 만들어집니다. 예를 들어, 기본 루트 풀 이름인 `rpool`을 사용하여 `/rpool/swap` 및 `/rpool/dump` 볼륨이 자동으로 생성됩니다. 스왑 및 덤프 볼륨의 크기를 선택한 크기로 조정할 수 있습니다. 단, 새 크기가 시스템 작동을 지원해야 합니다. 자세한 내용은 [Oracle Solaris ZFS 관리 설명서](#)의 “스왑 및 덤프 장치에 대한 ZFS 지원”을 참조하십시오.

설치 후 ZFS 스왑 장치 또는 덤프 장치를 수정해야 할 경우 이전 릴리스에서와 마찬가지로 `swap` 또는 `dumpadm` 명령을 사용하십시오.

본 문서의 덤프 장치 관리에 대한 자세한 내용은 235 페이지 “시스템 충돌 덤프 정보 관리”를 참조하십시오.

x86: GRUB 부트 환경의 시스템 충돌

GRUB 부트 환경의 x86 기반 시스템에서 시스템 충돌이 발생할 경우 다음번 시스템 재부트 시 GRUB 부트 아카이브를 관리하는 SMF 서비스(`svc:/system/boot-archive:default`)가 실패할 수 있습니다. GRUB 기반 부트에 대한 자세한 내용은 [시스템 관리 설명서: 기본 관리](#)의 “GRUB를 사용하여 x86 기반 시스템 부트(작업 맵)”를 참조하십시오.

시스템 충돌 덤프 파일

시스템 충돌 후 `savecore` 명령이 자동으로 실행되어 덤프 장치에서 충돌 덤프 정보가 검색되고 이름이 `unix.X` 및 `vmcore.X`인 파일 쌍이 작성됩니다. 여기서 `X`는 덤프 시퀀스 번호를 식별합니다. 해당 파일은 저장된 시스템 충돌 덤프 정보를 나타냅니다.

응용 프로그램이 비정상적으로 종료될 때 작성되는 사용자 응용 프로그램 이미지의 `코어` 파일과 충돌 덤프 파일이 혼동되는 경우도 있습니다.

충돌 덤프 파일은 미리 정해진 디렉토리(기본적으로 `/var/crash/hostname`)에 저장됩니다. 이전 릴리스에서는 물리적 메모리 이미지가 충돌 덤프 파일에 저장되도록 수동으로 설정하지 않은 경우 시스템이 재부트될 때 충돌 덤프 파일을 겹쳐 썼습니다. 지금은 충돌 덤프 파일 저장이 기본적으로 사용으로 설정되어 있습니다.

시스템 충돌 정보는 `dumpadm` 명령으로 관리됩니다. 자세한 내용은 234 페이지 “[dumpadm 명령](#)”을 참조하십시오.

충돌 덤프 저장

`mdb` 유틸리티를 사용하여 제어 구조, 활성 테이블, 실시간 또는 충돌한 시스템 커널의 메모리 이미지 및 커널 작동에 대한 기타 정보를 검사할 수 있습니다. `mdb`를 완전히 활용하려면 본 매뉴얼에서는 다루지 않는 커널에 대한 충분한 지식이 있어야 합니다. 이 유틸리티 사용에 대한 자세한 내용은 [mdb\(1\)](#) 매뉴얼 페이지를 참조하십시오.

또한 `savecore`로 저장되는 충돌 덤프를 고객 서비스 담당자에게 보낼 경우 시스템 충돌 원인 분석에 유용할 수 있습니다.

dumpadm 명령

`dumpadm` 명령을 사용하여 Oracle Solaris OS에서 시스템 충돌 덤프 정보를 관리할 수 있습니다.

- `dumpadm` 명령을 사용하여 운영 체제의 충돌 덤프를 구성할 수 있습니다. `dumpadm` 구성 매개변수에는 덤프 컨텐츠, 덤프 장치 및 충돌 덤프 파일이 저장되는 디렉토리가 포함됩니다.
- 덤프 데이터는 덤프 장치에 압축된 형식으로 저장됩니다. 커널 충돌 덤프 이미지는 4GB 이상일 수 있습니다. 데이터를 압축하면 덤프는 빨라지고 덤프 장치에 필요한 디스크 공간은 줄어듭니다.
- 스왑 영역이 아닌 전용 덤프 장치가 덤프 구성에 사용되는 경우 충돌 덤프 파일 저장이 백그라운드로 실행됩니다. 따라서 부트 시스템에서 `savecore` 명령이 완료될 때까지 기다리지 않고 다음 단계를 진행합니다. 대용량 메모리 시스템에서는 `savecore`가 완료되기 전에 시스템을 사용할 수 있습니다.
- `savecore` 명령으로 생성되는 시스템 충돌 덤프 파일은 기본적으로 저장됩니다.
- `savecore -L` 명령은 Oracle Solaris OS가 실행되는 충돌 덤프를 실시간으로 확인하는 데 사용할 수 있는 새로운 기능입니다. 이 명령은 잘못된 상태(예: 일시적인 성능 문제 또는 서비스 중단)가 발생할 때 메모리 스냅샷을 만들어 실행 중인 시스템 문제를 해결하는 데 사용됩니다. 시스템이 작동하고 일부 명령을 실행할 수 있을 경우 `savecore -L` 명령을 실행하여 시스템 스냅샷을 덤프 장치에 저장한 다음 충돌 덤프 파일을 `savecore` 디렉토리에 바로 기록할 수 있습니다. 시스템이 계속 실행 중이므로 전용 덤프 장치를 구성한 경우에만 `savecore -L` 명령을 사용할 수 있습니다.

다음 표에서는 `dumpadm`의 구성 매개변수에 대해 설명합니다.

덤프 매개변수	설명
덤프 장치	시스템 충돌 시 덤프 데이터를 임시로 저장하는 장치입니다. 덤프 장치가 스왑 영역이 아닐 경우 <code>savecore</code> 가 백그라운드로 실행되므로 부트 프로세스 속도가 빨라집니다.
<code>savecore</code> 디렉토리	시스템 충돌 덤프 파일을 저장하는 디렉토리입니다.
덤프 콘텐츠	덤프할 메모리 데이터의 유형입니다.
최소 사용 가능 공간	충돌 덤프 파일 저장 후 <code>savecore</code> 디렉토리에 필요한 최소 사용 가능 공간입니다. 최소 사용 가능 공간이 구성되지 않은 경우 기본값은 1MB입니다.

자세한 내용은 `dumpadm(1M)`을 참조하십시오.

덤프 구성 매개변수는 `dumpadm` 명령으로 관리됩니다.

dumpadm 명령 작동 방식

시스템 시작 시 `svc:/system/dumpadm:default` 서비스가 충돌 덤프 매개변수를 구성하기 위해 `dumpadm` 명령을 호출합니다.

`dumpadm`은 `/dev/dump` 인터페이스를 통해 덤프 장치 및 덤프 콘텐츠를 초기화합니다.

덤프 구성이 완료되면 `savecore` 스크립트가 충돌 덤프 파일 디렉토리의 위치를 찾습니다. 그런 다음 `savecore`가 호출되어 충돌 덤프가 확인되고 충돌 덤프 디렉토리에서 `minfree` 파일의 콘텐츠가 확인됩니다.

덤프 장치 및 볼륨 관리자

접근성 및 성능상 볼륨 관리 제품(예: Solaris Volume Manager)을 통해 제어되는 전용 덤프 장치는 구성하지 마십시오. 스왑 영역은 Solaris Volume Manager를 통해 제어되도록 하는 것이 좋지만 덤프 장치는 별도로 제어되도록 해야 합니다.

시스템 충돌 덤프 정보 관리

시스템 충돌 정보를 사용할 때는 다음 사항을 염두에 두어야 합니다.

- 슈퍼유저 또는 동등한 역할의 사용자로 시스템 충돌 정보에 액세스하여 해당 정보를 관리해야 합니다.
- 시스템 충돌 덤프 저장 옵션을 사용 안함으로 설정하지 마십시오. 시스템 충돌 덤프 파일은 시스템 충돌 원인을 확인할 수 있는 유용한 방법을 제공합니다.
- 고객 서비스 담당자에게 보내기 전까지는 중요한 시스템 충돌 정보를 제거하지 마십시오.

▼ 현재 충돌 덤프 구성 표시 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 현재 충돌 덤프 구성을 표시합니다.

```
# dumpadm
Dump content: kernel pages
Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/venus
Savecore enabled: yes
Saved compressed: on
```

위 출력 예는 다음을 의미합니다.

- 덤프 콘텐츠가 커널 메모리 페이지입니다.
- 커널 메모리가 스왑 장치인 /dev/dsk/c0t3d0s1에 덤프됩니다. `swap -l` 명령으로 모든 스왑 영역을 식별할 수 있습니다.
- 시스템 충돌 덤프 파일이 /var/crash/venus 디렉토리에 작성됩니다.
- 충돌 덤프 파일 저장이 사용으로 설정되어 있습니다.
- 충돌 덤프를 압축된 형식으로 저장합니다.

▼ 충돌 덤프 구성 수정 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 현재 충돌 덤프 구성을 식별합니다.

```
# dumpadm
Dump content: kernel pages
Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/pluto
Savecore enabled: yes
Save compressed: on
```

이 출력은 Oracle Solaris 10 릴리스가 실행되는 시스템의 기본 덤프 구성을 식별합니다.

- 3 충돌 덤프 구성을 수정합니다.

```
# /usr/sbin/dumpadm [-nuy] [-c content-type] [-d dump-device] [-m mink | minm | min%]  
[-s savecore-dir] [-r root-dir] [-z on | off]
```

- c *content* 덤프할 데이터의 유형을 지정합니다. 모든 커널 메모리를 덤프하려면 `kernel`을, 모든 메모리를 덤프하려면 `all`을, 커널 메모리와 충돌이 발생할 때 스레드가 실행되고 있었던 프로세스의 메모리 페이지를 덤프하려면 `curproc`를 사용합니다. 기본 덤프 콘텐츠는 커널 메모리입니다.
- d *dump-device* 시스템 충돌 시 덤프 데이터를 임시로 저장하는 장치를 지정합니다. 기본 스왑 장치가 기본적으로 사용되는 덤프 장치입니다.
- m *nnnk* | *nnnm* | *nnn%* 현재 `savecore` 디렉토리에 `minfree` 파일을 만들어 충돌 덤프 파일 저장을 위한 최소 사용 가능 디스크 공간을 지정합니다. 이 매개변수는 KB(`nnnk`), MB(`nnnm`) 또는 파일 시스템 크기 퍼센트(`nnn%`)로 지정할 수 있습니다. `savecore` 명령은 충돌 덤프 파일을 작성하기 전에 이 파일을 참조합니다. 크기를 기반으로 충돌 덤프 파일을 작성하여 사용 가능한 공간이 `minfree` 임계치 아래로 떨어지는 경우 덤프 파일이 작성되지 않고 오류 메시지가 기록됩니다. 이 시나리오 복구에 대한 자세한 내용은 239 페이지 “[전체 충돌 덤프 디렉토리에서 복구 방법\(선택 사항\)](#)”을 참조하십시오.
- n 시스템 재부트 시 `savecore`가 실행되지 않도록 지정합니다. 이 덤프 구성은 권장되지 않습니다. 시스템 충돌 정보가 스왑 장치에 기록되며 `savecore`가 사용으로 설정되지 않은 경우 시스템이 스왑을 시작할 때 충돌 덤프 정보를 겹쳐 씁니다.
- s 충돌 덤프 파일을 저장할 대체 디렉토리를 지정합니다. 기본 디렉토리는 `/var/crash/hostname`입니다. 여기서 `hostname`은 `uname -n` 명령의 출력입니다.
- u `/etc/dumpadm.conf` 파일의 콘텐츠를 기반으로 커널 덤프 구성을 강제로 업데이트합니다.
- y 재부트 시 `savecore` 명령이 자동으로 실행(이 덤프 설정의 기본값임)되도록 덤프 구성을 수정합니다.
- z *on* | *off* 재부트 시 `savecore` 명령 작동이 제어되도록 덤프 구성을 수정합니다. `on`으로 설정하면 코어 파일이 압축된 형식으로 저장됩니다. `off`로 설정하면 자동으로 충돌 덤프 파일의 압축이 풀립니다. 충돌 덤프 파일은 매우 커질 수 있으므로 충돌 덤프 파일이 압축된 형식으로 저장되면 파일 시스템 공간이 줄어들므로 기본값은 `on`입니다.

예 17-1 충돌 덤프 구성 수정

이 예에서는 모든 메모리가 전용 덤프 장치인 `/dev/dsk/c0t1d0s1`에 덤프되며 충돌 덤프 파일 저장 후 사용할 수 있어야 하는 최소 사용 가능 공간이 파일 시스템 공간의 10%입니다.

```
# dumpadm
  Dump content: kernel pages
  Dump device: /dev/dsk/c0t3d0s1 (swap)
Savecore directory: /var/crash/pluto
  Savecore enabled: yes
  Save compressed: on
# dumpadm -c all -d /dev/dsk/c0t1d0s1 -m 10%
  Dump content: all pages
  Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
  Savecore enabled: yes
  Save compressed: on
```

▼ 충돌 덤프 검사 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 `mdb` 유틸리티를 사용하여 충돌 덤프를 검사합니다.

```
# /usr/bin/mdb [-k] crashdump-file
-k          파일을 운영 체제 충돌 덤프 파일로 가정하여 커널 디버깅 모드를
           지정합니다.
crashdump-file  운영 체제 충돌 덤프 파일을 지정합니다.
```

- 3 충돌 상태 정보를 표시합니다.

```
# /usr/bin/mdb file-name
> ::status
.
.
.
> ::system
.
.
.
```

예 17-2 충돌 덤프 검사

다음 예에서는 시스템 정보를 포함하며 이 시스템의 `/etc/system` 파일에서 설정된 `tunable`을 식별하는 `mdb` 유틸리티의 샘플 출력을 보여 줍니다.

```
# /usr/bin/mdb -k unix.0
Loading modules: [ unix krtld genunix ip nfs ipc ptm ]
> ::status
debugging crash dump /dev/mem (64-bit) from ozlo
operating system: 5.10 Generic (sun4u)
> ::system
set ufs_ninode=0x9c40 [0t40000]
set ncsiz=0x4e20 [0t20000]
set pt_cnt=0x400 [0t1024]
```

▼ 전체 충돌 덤프 디렉토리에 복구 방법(선택 사항)

이 시나리오에서는 시스템이 충돌하지만 `savecore` 디렉토리에 남은 공간이 없으며 중요한 시스템 충돌 덤프 정보를 저장하고자 합니다.

- 1 시스템이 재부트되면 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
- 2 서비스 공급자에게 이미 보낸 기존 충돌 덤프 파일을 제거하여 `savecore` 디렉토리(일반적으로 `/var/crash/hostname`)를 지웁니다.
 - 또는 `savecore` 명령을 수동으로 실행하여 디스크 공간이 충분한 대체 디렉토리를 지정할 수도 있습니다.

```
# savecore [ directory ]
```

▼ 충돌 덤프 저장을 사용/사용 안함으로 설정하는 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.
역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.
- 2 시스템에서의 충돌 덤프 저장을 사용/사용 안함으로 설정합니다.

```
# dumpadm -n | -y
```

예 17-3 충돌 덤프 저장을 사용 안함으로 설정

다음 예에서는 시스템에서의 충돌 덤프 저장을 사용 안함으로 설정하는 방법을 보여 줍니다.

```
# dumpadm -n
  Dump content: all pages
  Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
  Savecore enabled: no
  Save Compressed: on
```

예 17-4 충돌 덤프 저장을 사용으로 설정

다음 예에서는 시스템에서의 충돌 덤프 저장을 사용으로 설정하는 방법을 보여 줍니다.

```
# dumpadm -y
  Dump content: all pages
  Dump device: /dev/dsk/c0t1d0s1 (dedicated)
Savecore directory: /var/crash/pluto (minfree = 77071KB)
  Savecore enabled: yes
  Save compressed: on
```

기타 소프트웨어 문제 해결(작업)

이 장에서는 가끔 발생할 수 있으며 비교적 쉽게 해결할 수 있는 기타 소프트웨어 문제에 대해 설명합니다. 기타 소프트웨어 문제 해결 과정에서는 특정 소프트웨어 응용 프로그램 또는 항목과 관련되지 않은 문제(예: 재부트 실패 및 파일 시스템 가득 참)를 해결합니다. 이러한 문제 해결은 다음 단원에서 설명됩니다.

다음은 이 장에서 다루는 정보를 나열한 것입니다.

- 241 페이지 “재부트를 실패할 경우 수행할 작업”
- 245 페이지 “x86: 시스템 재부트 중 SMF 부트 아카이브 서비스가 실패할 경우 수행할 작업”
- 246 페이지 “시스템이 정지될 경우 수행할 작업”
- 247 페이지 “파일 시스템이 가득 찬 경우 수행할 작업”
- 248 페이지 “복사 또는 복원 후 파일 ACL이 손실된 경우 수행할 작업”
- 248 페이지 “백업 문제 해결”
- 249 페이지 “Oracle Solaris OS에서 공통 에이전트 컨테이너 문제 해결”

재부트를 실패할 경우 수행할 작업

시스템이 완전히 재부트되지 않거나 재부트 후 다시 충돌하는 경우 시스템이 제대로 재부트되지 못하도록 하는 소프트웨어 또는 하드웨어 문제가 발생한 것일 수 있습니다.

시스템이 부트되지 않는 원인	문제 해결 방법
시스템이 <code>/platform/'uname -m'/kernel/unix</code> 를 찾을 수 없습니다.	SPARC 기반 시스템의 PROM에서 <code>boot-device</code> 설정을 변경해야 할 수 있습니다. 기본 부트 장치 변경에 대한 자세한 내용은 시스템 관리 설명서: 기본 관리 의 “부트 PROM을 사용하여 기본 부트 장치를 변경하는 방법”을 참조하십시오.

시스템이 부트되지 않는 원인	문제 해결 방법
<p>Oracle Solaris 10: x86 기반 시스템에 기본 부트 장치가 없습니다. 다음과 같은 메시지가 표시됩니다.</p> <p>Not a UFS filesystem.</p> <p>Solaris 10 1/06: GRUB 부트 아카이브가 손상되었습니다. 또는 SMF 부트 아카이브 서비스가 실패했습니다. <code>svcs -x</code> 명령을 실행하는 경우 오류 메시지가 표시됩니다.</p> <p><code>/etc/passwd</code> 파일에 잘못된 항목이 있습니다.</p> <p>디스크 또는 다른 장치와 관련된 하드웨어 문제가 있습니다.</p>	<p>Oracle Solaris 10: Configuration Assistant/부트 디스켓을 사용하여 시스템을 부트하고 부트할 디스크를 선택하십시오.</p> <p>Solaris 10 1/06: 비상 안전 조치를 한 아카이브를 부트하십시오.</p> <p>잘못된 <code>passwd</code> 파일 복구에 대한 자세한 내용은 시스템 관리 설명서: 기본 관리의 12 장, “Oracle Solaris 시스템 부트(작업)”을 참조하십시오.</p> <p>다음과 같이 하드웨어 연결을 확인하십시오.</p> <ul style="list-style-type: none"> ■ 장비가 연결되어 있는지 확인합니다. ■ 모든 스위치가 제대로 설정되어 있는지 확인합니다. ■ 이더넷 케이블을 비롯하여 모든 커넥터와 케이블을 확인합니다. ■ 모두 확인한 후에도 문제가 해결되지 않을 경우 시스템 전원을 껐다가 10-20초 후에 전원을 다시 켜십시오.

위 제안 조치를 통해 문제가 해결되지 않을 경우 현지 서비스 공급자에게 문의하십시오.

루트 암호를 잊은 경우 수행할 작업

루트 암호를 잊어 시스템에 로그인할 수 없는 경우 다음 작업을 수행해야 합니다.

- 키보드 중지 시퀀스를 사용하여 시스템을 중지합니다.
- **Oracle Solaris 10:** 부트 서버나 설치 서버 또는 로컬 CD-ROM에서 시스템을 부트합니다.
- 루트(/) 파일 시스템을 마운트합니다.
- `/etc/shadow` 파일에서 루트 암호를 제거합니다.
- 시스템을 재부트합니다.
- 로그인 후 루트 암호를 설정합니다.

루트 암호를 잊어 시스템에 로그인할 수 없는 경우 다음 작업을 수행해야 합니다.

- 키보드 중지 시퀀스를 사용하여 시스템을 중지합니다.
- **Solaris 10 1/06 릴리스부터:** x86 기반 시스템에서는 Solaris의 비상 안전 조치를 한 아카이브로 시스템을 부트합니다.
- **Oracle Solaris 10:** 부트 서버나 설치 서버 또는 로컬 CD-ROM에서 시스템을 부트합니다.
- 루트(/) 파일 시스템을 마운트합니다.
- /etc/shadow 파일에서 루트 암호를 제거합니다.
- 시스템을 재부트합니다.
- 로그인 후 루트 암호를 설정합니다.

이러한 절차는 **시스템 관리 설명서: 기본 관리의 12 장, “Oracle Solaris 시스템 부트(작업)”**에서 자세히 설명됩니다.

주 - 이 릴리스에서는 SPARC 기반 시스템에서 GRUB 기반 부트를 사용할 수 없습니다.

다음 예에서는 SPARC 및 x86 기반 시스템에서 잊은 루트 암호를 복구하는 방법에 대해 설명합니다.

예 18-1 SPARC: 루트 암호를 잊은 경우 수행할 작업

다음 예에서는 루트 암호를 잊은 경우 네트워크에서 부트하여 복구하는 방법에 대해 설명합니다. 이 예에서는 부트 서버를 사용할 수 있다고 가정합니다. 시스템이 재부트된 후에는 새 루트 암호를 적용해야 합니다.

```
(Use keyboard abort sequence--Press Stop A keys to stop the system)
ok boot net -s
# mount /dev/dsk/c0t3d0s0 /a
# cd /a/etc
# TERM=vt100
# export TERM
# vi shadow
(Remove root's encrypted password string)
# cd /
# umount /a
# init 6
```

예 18-2 x86: 루트 암호를 잊은 경우 GRUB 기반 부트 수행

이 예에서는 부트 서버를 사용할 수 있다고 가정합니다. 시스템이 재부트된 후에는 새 루트 암호를 적용해야 합니다.

```
GNU GRUB version 0.95 (637K lower / 3144640K upper memory)
+-----+
| be1
```

예 18-2 x86: 루트 암호를 잊은 경우 GRUB 기반 부트 수행 (계속)

```
| be1 failsafe
| be3
| be3 failsafe
| be2
| be2 failfafa
+-----+
      Use the ^ and v keys to select which entry is highlighted.
      Press enter to boot the selected OS, 'e' to edit the
      commands before booting, or 'c' for a command-line.

Searching for installed OS instances...

      An out of sync boot archive was detected on /dev/dsk/c0t0d0s0.
      The boot archive is a cache of files used during boot and
      should be kept in sync to ensure proper system operation.

      Do you wish to automatically update this boot archive? [y,n,?] n
Searching for installed OS instances...

Multiple OS instances were found. To check and mount one of them
read-write under /a, select it from the following list. To not mount
any, select 'q'.

      1 pool10:13292304648356142148      ROOT/be10
      2 rpool:14465159259155950256      ROOT/be01

Please select a device to be mounted (q for none) [?,??,q]: 1
mounting /dev/dsk/c0t0d0s0 on /a
starting shell.
.
.
.
# cd /a/etc
# vi shadow
(Remove root's encrypted password string)
# cd /
# umount /a
# reboot
```

예 18-3 x86: 루트 암호를 잊은 경우 시스템 부트

Oracle Solaris 10: 다음 예에서는 루트 암호를 잊은 경우 네트워크에서 부트하여 복구하는 방법을 보여 줍니다. 이 예에서는 부트 서버를 사용할 수 있다고 가정합니다. 시스템이 재부트된 후에는 새 루트 암호를 적용해야 합니다.

```
Press any key to reboot.
Resetting...
.
.
.
Initializing system
Please wait...
```

예 18-3 x86: 루트 암호를 잊은 경우 시스템 부트 (계속)

```

<<< Current Boot Parameters >>>
Boot path: /pci@0,0/pci-ide@7,1/ide@0/cmdk@0,0:a
Boot args:

Type   b [file-name] [boot-flags] <ENTER>   to boot with options
or     i <ENTER>                           to enter boot interpreter
or     <ENTER>                               to boot with defaults

<<< timeout in 5 seconds >>>

Select (b)oot or (i)nterpreter: b -s
SunOS Release 5.10 Version amd64-gate-2004-09-30 32-bit
Copyright (c) 1983, 2011, Oracle and/or its affiliates. All rights reserved.
Use is subject to license terms.
DEBUG enabled
Booting to milestone "milestone/single-user:default".
Hostname: venus
NIS domain name is example.com
Requesting System Maintenance Mode
SINGLE USER MODE

Root password for system maintenance (control-d to bypass): xxxxxx
Entering System Maintenance Mode
.
.
.
# mount /dev/dsk/c0t0d0s0 /a
.
.
.
# cd /a/etc
# vi shadow
(Remove root's encrypted password string)
# cd /
# umount /a
# init 6

```

x86: 시스템 재부트 중 SMF 부트 아카이브 서비스가 실패할 경우 수행할 작업

Solaris 10 1/06: 이 릴리스에서는 시스템이 충돌할 경우 시스템이 재부트될 때 부트 아카이브 SMF 서비스 `svc:/system/boot-archive:default`가 실패할 수 있습니다. 부트 아카이브 서비스가 실패한 경우 `svcs -x` 명령을 실행할 때 다음과 유사한 메시지가 표시됩니다.

```

svc:/system/boot-archive:default (check boot archive content)
State: maintenance since Fri Jun 03 10:24:52 2005
Reason: Start method exited with $SMF_EXIT_ERR_FATAL.
See: http://sun.com/msg/SMF-8000-KS
See: /etc/svc/volatile/system-boot-archive:default.log

```

Impact: 48 dependent services are not running. (Use -v for list.)

```
svc:/network/rpc/gss:default (Generic Security Service)
  State: uninitialized since Fri Jun 03 10:24:51 2005
  Reason: Restarter svc:/network/inetd:default is not running.
  See: http://sun.com/msg/SMF-8000-5H
  See: gssd(1M)
```

Impact: 10 dependent services are not running. (Use -v for list.)

```
svc:/application/print/server:default (LP print server)
  State: disabled since Fri Jun 03 10:24:51 2005
  Reason: Disabled by an administrator.
  See: http://sun.com/msg/SMF-8000-05
  See: lpsched(1M)
```

Impact: 1 dependent service is not running. (Use -v for list.)

문제를 해결하려면 다음 작업을 수행하십시오.

1. 시스템을 재부트하고 GRUB 부트 메뉴에서 비상 안전 조치를 한 아카이브 옵션을 선택합니다.
2. 프롬프트가 표시되면 `y`를 선택하여 부트 아카이브를 재구성합니다.
부트 아카이브가 재구성되면 시스템을 부트할 수 있습니다.
3. 부트를 계속하려면 다음 명령을 사용하여 SMF 부트 아카이브 서비스를 지웁니다.

```
# svcadm clear boot-archive
```

이 명령을 실행하려면 슈퍼유저 또는 동등한 권한이 있어야 합니다.

GRUB 부트 아카이브 재구성에 대한 자세한 내용은 [시스템 관리 설명서: 기본 관리의 “x86 기반 시스템을 비상 안전 모드로 부트하는 방법”](#) 및 `bootadm(1M)` 매뉴얼 페이지를 참조하십시오.

시스템이 정지될 경우 수행할 작업

일부 소프트웨어 프로세스가 멈춘 경우 시스템이 충돌하는 것이 아니라 멈추거나 정지될 수 있습니다. 다음 단계에 따라 정지된 시스템을 복구하십시오.

1. 시스템에서 윈도우 환경이 실행되고 있는지 여부를 확인하고 다음 제안 사항을 따릅니다. 이러한 제안 사항으로 문제가 해결되지 않을 경우 2단계로 이동합니다.
 - 포인터가 명령을 입력 중인 창에 있는지 확인합니다.
 - 사용자가 실수로 `Ctrl-s`를 누른 경우 `Ctrl-q`를 눌러 화면을 고정합니다. `Ctrl-s`를 누르면 전체 화면이 고정되는 것이 아니라 창만 고정됩니다. 창이 고정된 경우 다른 창을 사용해 봅니다.
 - 가능한 경우 네트워크의 다른 시스템에서 원격으로 로그인합니다. `pgrep` 명령을 사용하여 정지된 프로세스를 찾습니다. 윈도우 시스템이 정지된 것 같으면 프로세스를 식별하여 강제 종료합니다.
2. `Ctrl-\`를 눌러 실행 중인 프로그램을 강제로 "종료"하고 `core` 파일을 작성합니다.

3. Ctrl-c를 눌러 실행 중일 수 있는 프로그램을 중단합니다.
4. 원격으로 로그인하여 시스템 정지 원인이 되는 프로세스를 식별하여 강제 종료합니다.
5. 슈퍼유저 또는 동등한 역할의 사용자로 원격으로 로그인하여 시스템을 재부트합니다.
6. 시스템이 계속 응답하지 않을 경우 충돌 덤프를 강제로 실행하고 재부트합니다. 강제 충돌 덤프 및 재부트에 대한 자세한 내용은 [시스템 관리 설명서: 기본 관리의 “시스템의 충돌 덤프 및 재부트 강제 수행”](#)을 참조하십시오.
7. 시스템이 계속 응답하지 않을 경우 전원을 껐다가 1분 정도 후에 전원을 다시 켵니다.
8. 시스템이 응답하도록 문제를 해결할 수 없을 경우 현지 서비스 공급자의 도움을 받으십시오.

파일 시스템이 가득 찬 경우 수행할 작업

루트(/) 파일 시스템 또는 기타 파일 시스템이 가득 찬 경우 콘솔 창에 다음 메시지가 표시됩니다.

```
.... file system full
```

파일 시스템이 가득 차는 원인은 여러 가지입니다. 다음 단원에서는 가득 찬 파일 시스템을 복구할 수 있는 여러 가지 시나리오에 대해 설명합니다. 파일 시스템이 가득 차지 않도록 오래된 파일과 사용되지 않는 파일을 주기적으로 정리하는 방법은 [6장](#), [“디스크 사용 관리\(작업\)”](#)를 참조하십시오.

큰 파일 또는 디렉토리가 만들어져 파일 시스템이 가득 참

오류 발생 원인

사용자가 실수로 파일 또는 디렉토리를 잘못된 위치에 복사했습니다. 응용 프로그램이 충돌하고 큰 core 파일을 파일 시스템에 작성한 경우에도 이 오류가 발생합니다.

문제 해결 방법

슈퍼유저 또는 동등한 역할의 사용자로 로그인하고 특정 파일 시스템에서 `ls -tl` 명령을 사용하여 큰 파일이 새로 생성된 위치를 식별한 후 제거합니다. core 파일 제거에 대한 자세한 내용은 [87 페이지 “코어 파일 찾기 및 삭제 방법”](#)을 참조하십시오.

시스템 메모리 부족으로 인해 TMPFS 파일 시스템이 가득 참

오류 발생 원인	문제 해결 방법
TMPFS가 허용된 수를 초과하여 작성을 시도하거나 현재 프로세스 중 일부에서 많은 양의 메모리를 사용 중인 경우 이 오류가 발생할 수 있습니다.	tmpfs 관련 오류 메시지 복구에 대한 자세한 내용은 tmpfs(7FS) 매뉴얼 페이지를 참조하십시오.

복사 또는 복원 후 파일 ACL이 손실된 경우 수행할 작업

오류 발생 원인	문제 해결 방법
ACL이 있는 파일 또는 디렉토리가 /tmp 디렉토리에 복사되거나 복원되는 경우 ACL 속성이 손실됩니다. 일반적으로 /tmp 디렉토리는 UFS 파일 시스템 속성(예: ACL)을 지원하지 않는 임시 파일 시스템으로 마운트됩니다.	대신 /var/tmp 디렉토리에 파일을 복사하거나 복원합니다.

백업 문제 해결

이 단원에서는 데이터를 백업 및 복원할 때 사용할 몇 가지 기본적인 문제 해결 방법에 대해 설명합니다.

파일 시스템 백업 후 루트(/) 파일 시스템이 가득 참

파일 시스템을 백업하면 루트(/) 파일 시스템이 가득 찹니다. 매체에 어떤 항목도 작성되지 않고 `ufsdump` 명령이 두번째 볼륨의 매체를 삽입하라는 프롬프트를 표시합니다.

오류 발생 원인	문제 해결 방법
<code>-f</code> 옵션과 함께 잘못된 대상 장치 이름을 사용한 경우 <code>ufsdump</code> 명령이 루트(/) 파일 시스템의 /dev 디렉토리에 있는 파일에 항목을 작성하므로 파일 시스템이 가득 찹니다. 예를 들어, /dev/rmt/0 대신 /dev/rmt/st0을 입력한 경우 /dev/rmt/st0 백업 파일이 테이프 드라이브로 전송되는 것이 아니라 디스크에 생성됩니다.	/dev 디렉토리에서 <code>ls -tl</code> 명령을 사용하여 비정상적으로 큰 파일이 새로 생성된 위치를 식별한 후 제거합니다.

백업 명령과 복원 명령은 동일해야 함

ufsdump 명령으로 백업된 파일을 복원할 때는 ufsrestore 명령만 사용할 수 있습니다. tar 명령을 사용하여 백업한 경우 tar 명령으로 복원하십시오. ufsrestore 명령을 사용하여 다른 명령과 함께 작성된 테이프를 복원하면 테이프가 ufsdump 형식이 아님을 알리는 오류 메시지가 표시됩니다.

적합한 현재 디렉토리를 사용하는지 확인

잘못된 위치에 파일을 복원하는 경우가 많습니다. ufsdump 명령은 항상 파일 시스템 루트에 대해 상대적인 전체 파일 이름을 사용하여 파일을 복사하므로 일반적으로 ufsrestore 명령을 실행하기 전에 파일 시스템의 루트 디렉토리를 변경해야 합니다. 레벨이 보다 낮은 디렉토리로 변경할 경우 파일을 복원한 후 해당 디렉토리에 생성된 전체 파일 트리가 표시됩니다.

대화형 명령

대화형 명령을 사용하면 다음 예와 같은 ufsrestore> 프롬프트가 표시됩니다.

```
# ufsrestore ivf /dev/rmt/0
Verify volume and initialize maps
Media block size is 126
Dump date: Fri Jan 30 10:13:46 2004
Dumped from: the epoch
Level 0 dump of /export/home on starbug:/dev/dsk/c0t0d0s7
Label: none
Extract directories from tape
Initialize symbol table.
ufsrestore >
```

Oracle Solaris OS에서 공통 에이전트 컨테이너 문제 해결

이 단원에서는 공통 에이전트 컨테이너 공유 구성 요소에서 발생할 수 있는 문제를 해결합니다. 이 Oracle Solaris 릴리스에서는 Oracle Solaris OS에 공통 에이전트 컨테이너 Java 프로그램이 포함되어 있습니다. 이 프로그램은 Java 관리 응용 프로그램용 컨테이너를 구현합니다. 일반적으로 사용자에게는 컨테이너가 표시되지 않습니다.

잠재적인 문제는 다음과 같습니다.

- 포트 번호 충돌
- 슈퍼유저 암호 보안 손상

포트 번호 충돌

공통 에이전트 컨테이너에는 기본적으로 다음과 같은 포트 번호가 사용됩니다.

- JMX 포트(TCP) = 11162
- SNMPAdaptor 포트(UDP) = 11161
- 트랩용 SNMPAdaptor 포트(UDP) = 11162
- Commandstream 어댑터 포트(TCP) = 11163
- RMI 커넥터 포트(TCP) = 11164

주 - Oracle Solaris Cluster 설치 문제를 해결하는 경우 포트 지정이 다릅니다.

설치에서 이러한 포트 번호가 이미 예약된 경우 다음 절차의 설명에 따라 공통 에이전트 컨테이너에 사용되는 포트 번호를 변경하십시오.

▼ 포트 번호 확인 방법

다음 절차는 포트 확인 방법을 보여 줍니다.

- 1 수퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 공통 에이전트 컨테이너 관리 데몬을 중지합니다.

```
# /usr/sbin/cacaoadm stop
```

- 3 다음 구문을 사용하여 포트 번호를 변경합니다.

```
# /usr/sbin/cacaoadm set-param param=value
```

예를 들어, SNMPAdaptor에 사용되는 포트를 기본값인 11161에서 11165로 변경하려면 다음과 같이 입력합니다.

```
# /usr/sbin/cacaoadm set-param snmp-adaptor-port=11165
```

- 4 공통 에이전트 컨테이너 관리 데몬을 다시 시작합니다.

```
# /usr/sbin/cacaoadm start
```

수퍼유저 암호 보안 손상

Java ES가 실행되고 있는 호스트에서 보안 키를 재생성해야 할 수 있습니다. 예를 들어, 수퍼유저 암호가 노출되었거나 손상되었을 위험이 있으면 보안 키를 재생성해야

합니다. 공통 에이전트 컨테이너 서비스에서 사용되는 키는 `/etc/cacao/instances/instance-name/security` 디렉토리에 저장됩니다. 다음 작업은 Oracle Solaris OS용 보안 키 재생성 방법을 보여 줍니다.

▼ Oracle Solaris OS용 보안 키 생성 방법

- 1 슈퍼유저 또는 동등한 역할의 사용자로 로그인합니다.

역할에는 권한 부여 및 권한이 있는 명령이 포함됩니다. 역할에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 “Configuring RBAC (Task Map)”를 참조하십시오.

- 2 공통 에이전트 컨테이너 관리 데몬을 중지합니다.

```
# /usr/sbin/cacaoadm stop
```

- 3 보안 키를 재생성합니다.

```
# /usr/sbin/cacaoadm create-keys --force
```

- 4 공통 에이전트 컨테이너 관리 데몬을 다시 시작합니다.

```
# /usr/sbin/cacaoadm start
```

주 - Oracle Sun Cluster 소프트웨어의 경우 클러스터의 모든 노드에 이 변경 사항을 전파해야 합니다.

파일 액세스 문제 해결(작업)

이 장에서는 파일 액세스 문제(예: 잘못된 권한 및 검색 경로와 관련된 문제) 해결에 대한 정보를 제공합니다.

다음은 이 장에서 다루는 문제 해결 항목을 나열한 것입니다.

- 253 페이지 “검색 경로 문제 해결(Command not found)”
- 255 페이지 “파일 액세스 문제 해결”
- 256 페이지 “네트워크 액세스 문제 인식”

사용자들은 종종 이전에 사용할 수 있었던 프로그램, 파일 또는 디렉토리에 액세스할 수 없어 문제가 발생할 경우 시스템 관리자에게 도움을 요청합니다.

이와 같은 문제가 발생할 경우 항상 다음 세 가지 측면 중 하나를 조사하십시오.

- 사용자의 검색 경로가 변경되었거나 검색 경로 내 디렉토리의 순서가 부적절한 것일 수 있습니다.
- 파일 또는 디렉토리의 권한이나 소유권이 부적절한 것일 수 있습니다.
- 네트워크를 통해 액세스된 시스템의 구성이 변경된 것일 수 있습니다.

이 장에서는 이와 같은 세 가지 측면에서 각각 문제를 인식하는 방법에 대해 간략하게 설명하고 가능한 해결 방법을 제안합니다.

검색 경로 문제 해결(Command not found)

Command not found 메시지는 다음 중 하나를 나타냅니다.

- 시스템에서 명령을 사용할 수 없습니다.
- 검색 경로에 명령 디렉토리가 없습니다.

검색 경로 문제를 해결하려면 명령이 저장된 디렉토리의 경로 이름을 알아야 합니다.

잘못된 버전의 명령이 있을 경우 동일한 이름의 명령이 있는 디렉토리가 검색 경로에 포함됩니다. 이 경우 적절한 디렉토리가 검색 경로의 뒷부분에 포함되거나 아예 표시되지 않을 수 있습니다.

echo \$PATH 명령을 사용하여 현재 검색 경로를 표시할 수 있습니다. 예를 들어, 다음과 같습니다.

```
$ echo $PATH
/home/kryten/bin:/sbin:/usr/sbin:/usr/bin:/usr/dt:/usr/dist/exe
```

which 명령을 사용하여 잘못된 버전의 명령을 실행 중인지 여부를 확인할 수 있습니다. 예를 들어, 다음과 같습니다.

```
$ which acroread
/usr/doctools/bin/acroread
```

주 - which 명령은 .cshrc 파일에서 경로 정보를 찾습니다. 명령을 Bourne 또는 Korn 셸에서 실행하고 .cshrc 파일에 which 명령의 별칭이 포함되어 있을 경우 which 명령이 잘못된 결과를 제공할 수 있습니다. 정확한 결과를 얻으려면 C 셸에서 which 명령을 사용하거나 Korn 셸에서 whence 명령을 사용하십시오.

▼ 검색 경로문제 진단 및 해결 방법

- 1 현재 검색 경로를 표시하여 명령에 대한 디렉토리가 경로에 없거나 디렉토리의 철자가 잘못되었는지 확인합니다.

```
$ echo $PATH
```

- 2 다음 사항을 확인합니다.

- 검색 경로가 올바른지 여부
- 검색 경로가 다른 버전의 명령이 있는 다른 검색 경로 앞에 나열되는지 여부
- 명령이 검색 경로 중 하나에 포함되어 있는지 여부

경로를 수정해야 할 경우 3단계로 이동합니다. 그렇지 않을 경우 4단계로 이동합니다.

- 3 다음 테이블과 같이 경로를 적합한 파일에 추가합니다.

셸	파일	구문	주
Bourne 및 Korn	\$HOME/.profile	\$ PATH=\$HOME/bin:/sbin:/usr/local/bin... \$ export PATH	경로 이름은 콜론으로 구분합니다.

셸	파일	구문	주
C	\$HOME/.cshrc 또는 \$HOME/.login	<i>hostname%</i> set path=(~bin /sbin /usr/local/bin ...)	경로 이름은 공백으로 구분합니다.

4 다음과 같이 새 경로를 활성화합니다.

셸	경로를 찾을 파일	경로 활성화에 사용할 명령
Bourne 및 Korn	.profile	\$. ./profile
C	.cshrc	<i>hostname%</i> source .cshrc
	.login	<i>hostname%</i> source .login

5 새 경로를 확인합니다.

```
$ which command
```

예 19-1 검색 경로 문제 진단 및 해결

다음 예에서는 which 명령을 사용할 때 mytool 실행 파일이 검색 경로 내 디렉토리에 없음을 보여 줍니다.

```
venus% mytool
mytool: Command not found
venus% which mytool
no mytool in /sbin /usr/sbin /usr/bin /etc /home/ignatz/bin .
venus% echo $PATH
/sbin /usr/sbin /usr/bin /etc /home/ignatz/bin
venus% vi ~/.cshrc
(Add appropriate command directory to the search path)
venus% source .cshrc
venus% mytool
```

명령을 찾을 수 없는 경우 매뉴얼 페이지에서 해당 디렉토리 경로를 찾으십시오. 예를 들어, lpsched 명령(lp 프린터 데몬)을 찾을 수 없는 경우 [lpsched\(1M\)](#) 매뉴얼 페이지에서 경로가 /usr/lib/lp/lpsched임을 확인할 수 있습니다.

파일 액세스 문제 해결

사용자가 이전에 액세스할 수 있었던 파일 또는 디렉토리에 액세스할 수 없을 경우 파일 또는 디렉토리의 권한이나 소유권이 변경된 것일 수 있습니다.

파일 및 그룹 소유권 변경

파일 및 디렉토리 소유권 변경은 다른 사용자가 수퍼유저로 파일을 편집했기 때문인 경우가 많습니다. 새 사용자에 대한 홈 디렉토리를 만들 때 사용자를 홈 디렉토리 내 dot(.) 파일의 소유자로 설정해야 합니다. "." 파일을 소유하지 않은 사용자는 고유 홈 디렉토리에 파일을 만들 수 없습니다.

그룹 소유권이 변경되거나 사용자가 속한 그룹이 /etc/group 데이터베이스에서 삭제된 경우에도 액세스 문제가 발생할 수 있습니다.

액세스 문제가 발생한 파일의 권한 또는 소유권을 변경하는 방법에 대한 자세한 내용은 [System Administration Guide: Security Services](#)의 6 장, “Controlling Access to Files (Tasks)”를 참조하십시오.

네트워크 액세스 문제 인식

사용자가 rcp 원격 복사 명령을 사용하여 네트워크를 통해 파일을 복사할 때 문제가 발생할 경우 원격 시스템의 디렉토리 및 파일이 권한 설정을 통해 액세스를 제한한 것일 수 있습니다. 원격 시스템 및 로컬 시스템이 액세스를 허용하도록 구성되지 않은 경우에도 문제가 발생할 수 있습니다.

네트워크 액세스 문제 및 AutoFS를 통해 시스템에 액세스할 때 발생하는 문제에 대한 자세한 내용은 [System Administration Guide: Network Services](#)의 “Strategies for NFS Troubleshooting”을 참조하십시오.

UFS 파일 시스템 불일치 해결(작업)

이 장에서는 fsck 오류 메시지 및 가능한 오류 메시지 해결 조치에 대해 설명합니다.

주 - Solaris 10 6/06 릴리스부터 fsck 명령을 실행할 때 표시되는 오류 메시지가 변경되었습니다. 이 장에서는 수정된 fsck 오류 메시지를 다룹니다. Solaris 10 6/06 릴리스 이상을 실행하고 있지 않은 시스템에서 fsck 명령을 실행하는 것과 관련된 정보는 “Oracle Solaris 10”이라는 레이블이 있는 절의 오류 메시지를 참조하십시오. 현재 릴리스에서 향상된 fsck에 대한 전체 세부 정보는 [System Administration Guide: Devices and File Systems](#)를 참조하십시오.

다음은 이 장에서 다루는 정보를 나열한 것입니다.

- 259 페이지 “일반 fsck 오류 메시지”
- 261 페이지 “초기화 단계 fsck 메시지”
- 264 페이지 “1단계: 블록 및 크기 메시지 확인”
- 269 페이지 “1B단계: 추가 DUPS 메시지 재검색”
- **Oracle Solaris 10:** 268 페이지 “Oracle Solaris 10: 1B단계: 추가 DUPS 메시지 재검색”
- 269 페이지 “2단계: 경로 이름 메시지 확인”
- 276 페이지 “3단계: 연결 메시지 확인”
- 278 페이지 “4단계: 참조 수 메시지 확인”
- 281 페이지 “5단계: 실린더 그룹 메시지 확인”
- **Oracle Solaris 10:** 282 페이지 “5단계: 실린더 그룹 메시지 확인”
- 283 페이지 “fsck 요약 메시지”
- **Oracle Solaris 10:** 284 페이지 “정리 단계 메시지”

fsck 명령 및 해당 명령을 사용하여 파일 시스템 무결성을 확인하는 방법에 대한 자세한 내용은 [System Administration Guide: Devices and File Systems](#)의 20 장, “Checking UFS File System Consistency (Tasks)”를 참조하십시오.

fsck 오류 메시지

일반적으로 fsck 명령은 최신 파일 시스템 변경 사항이 디스크에 기록되지 않은 상태에서 시스템이 예기치 않게 정지된 후 파일 시스템을 **다듬기** 위해 비대화형으로 실행됩니다. 다듬기는 자동으로 기본적인 파일 시스템 불일치만 수정하고 보다 심각한 오류를 복구하려고 시도하지 않습니다. 파일 시스템을 다듬는 동안 fsck 명령은 예기치 않은 정지로 인해 발생한 불일치를 수정합니다. 보다 심각한 상태가 발생한 경우 이 명령은 오류를 보고한 후 종료됩니다.

fsck 명령을 대화형으로 실행하면 발견된 각 불일치가 보고되고 사소한 오류가 수정되지만, 보다 심각한 오류가 있을 경우 불일치가 보고되고 응답 선택 프롬프트가 표시됩니다. `-y` 또는 `-n` 옵션으로 fsck 명령을 실행하면 해당 응답은 fsck 명령이 각 오류 상태에 대해 제안하는 기본 응답(예 또는 아니오)으로 미리 정의됩니다.

데이터 손실을 초래하는 해결 조치도 있습니다. 데이터 손실의 정도와 심각도는 fsck 진단 출력에서 확인할 수 있습니다.

fsck 명령은 다중 패스 파일 시스템 검사 프로그램입니다. 각 패스는 여러 메시지 세트를 사용하여 fsck 명령의 다른 단계를 호출합니다. 초기화 후 fsck 명령은 각 파일 시스템을 통해 연속 패스를 수행하면서 블록 및 크기, 경로 이름, 연결, 참조 수 및 사용 가능한 블록 맵(가능한 경우 재구성)을 확인합니다. 또한 일부 정리를 수행합니다.

UFS 버전의 fsck 명령이 수행하는 단계(패스)는 다음과 같습니다.

- 초기화
- 1단계 - 블록 및 크기 확인
- 2a단계 - 중복된 이름 확인
- 2b단계 - 경로 이름 확인
- 3단계 - 연결 확인
- 3b단계 - 새도우/ACL 확인
- 4단계 - 참조 수 확인
- 5단계 - 실린더 그룹 확인

다음 단원에서는 각 단계에서 감지될 수 있는 오류 상태와 이로 인한 메시지 및 프롬프트, 선택할 수 있는 응답에 대해 설명합니다.

두 개 이상의 단계에서 나타날 수 있는 메시지는 [259 페이지 “일반 fsck 오류 메시지”](#)에서 설명됩니다. 그 밖의 메시지는 발생하는 단계에 따라 영문자순으로 구성됩니다.

다음 표는 fsck 오류 메시지에 포함되는 몇 가지 약어를 나열한 것입니다.

표 20-1 오류 메시지 약어

약어	의미
BLK	블록 번호
DUP	중복된 블록 번호

표 20-1 오류 메시지 약어 (계속)

약어	의미
DIR	디렉토리 이름
CG	실린더 그룹
MTIME	파일이 마지막으로 수정된 시간
UNREF	참조되지 않음

변수 필드(예: inode 번호)가 포함되는 메시지도 많습니다. 본 설명서에서 변수 필드는 *inode-number*와 같이 기울임꼴로 표시됩니다. 예를 들어, 다음과 같은 화면 메시지가 있다고 가정합니다.

```
INCORRECT BLOCK COUNT I=2529
```

이는 다음과 같이 표시됩니다.

```
INCORRECT BLOCK COUNT I=inode-number
```

일반 fsck 오류 메시지

이 단원의 오류 메시지는 초기화 후 단계에서 표시될 수 있습니다. 작업을 계속할 수 있는 옵션이 제공되기는 하지만 일반적으로 해당 메시지는 치명적인 것으로 간주하는 것이 좋습니다. 해당 메시지는 심각한 시스템 오류를 나타내는 것으로 즉시 처리해야 합니다. 해당 메시지가 표시된 경우 n(아니오)을 입력하여 프로그램을 종료하십시오. 문제의 원인을 확인할 수 없을 경우 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

```
CANNOT SEEK: BLK disk-block-number (CONTINUE)
```

Oracle Solaris 10:

```
CANNOT SEEK: BLK block-number (CONTINUE)
```

원인

파일 시스템에서 지정된 *disk-block-number* 블록 번호로의 이동 요청을 실패했습니다. 이 메시지는 심각한 문제(하드웨어 오류일 수 있음)를 나타냅니다.

Oracle Solaris 10: 파일 시스템에서 지정된 *block-number* 블록 번호로의 이동 요청을 실패했습니다. 이 메시지는 심각한 문제(하드웨어 오류일 수 있음)를 나타냅니다.

파일 시스템 검사를 계속하도록 선택할 경우 fsck가 이동을 재시도하고 이동할 수 없는 섹터 번호 목록을 표시합니다. 블록이 가상 메모리 버퍼 캐시의 일부인 경우 fsck가 치명적인 I/O 오류 메시지와 함께 종료됩니다.

조치

디스크에 하드웨어 문제가 발생한 경우에는 이 문제가 계속 나타납니다. fsck를 다시 실행하여 파일 시스템을 재확인하십시오.

재확인을 실패할 경우 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

CANNOT READ: DISK BLOCK *disk-block-number*: I/O ERROR
CONTINUE?

Oracle Solaris 10:

CANNOT READ: DISK BLOCK *block-number*: I/O ERROR
CONTINUE?

원인

파일 시스템에서 지정된 *disk-block-number* 블록 번호 읽기 요청을 실패했습니다. 이 메시지는 심각한 문제(하드웨어 오류일 수 있음)를 나타냅니다.

Oracle Solaris 10: 파일 시스템에서 지정된 *block-number* 블록 번호 읽기 요청을 실패했습니다. 이 메시지는 심각한 문제(하드웨어 오류일 수 있음)를 나타냅니다.

파일 시스템 검사를 계속하도록 선택할 경우 fsck가 읽기를 재시도하고 읽을 수 없는 섹터 번호 목록을 표시합니다. 블록이 가상 메모리 버퍼 캐시의 일부인 경우 fsck가 치명적인 I/O 오류 메시지와 함께 종료됩니다. fsck가 읽기를 실패한 블록 중 하나를 다시 쓰려고 시도하면 다음 메시지가 표시됩니다.

WRITING ZERO'ED BLOCK *sector-numbers* TO DISK

조치

디스크에 하드웨어 문제가 발생한 경우에는 이 문제가 계속 나타납니다. fsck를 다시 실행하여 파일 시스템을 재확인하십시오. 재확인을 실패할 경우 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

CANNOT WRITE: BLK *disk-block-number* (CONTINUE)

Oracle Solaris 10:

CANNOT WRITE: BLK *block-number* (CONTINUE)

원인

파일 시스템에서 지정된 *disk-block-number* 블록 번호 쓰기 요청을 실패했습니다.

파일 시스템 검사를 계속하도록 선택할 경우 fsck가 쓰기를 재시도하고 쓸 수 없는 섹터 번호 목록을 표시합니다. 블록이 가상 메모리 버퍼 캐시의 일부인 경우 fsck가 치명적인 I/O 오류 메시지와 함께 종료됩니다.

Oracle Solaris 10: 파일 시스템에서 지정된 *block-number* 블록 번호 쓰기 요청을 실패했습니다.

파일 시스템 검사를 계속하도록 선택할 경우 fsck가 쓰기를 재시도하고 쓸 수 없는 섹터 번호 목록을 표시합니다. 블록이 가상 메모리 버퍼 캐시의 일부인 경우 fsck가 치명적인 I/O 오류 메시지와 함께 종료됩니다.

조치

디스크에 대한 쓰기가 보호된 것일 수 있습니다. 드라이브에서 쓰기 보호 잠금을 확인하십시오. 디스크에 하드웨어 문제가 발생한 경우 문제가 지속됩니다. fsck를 다시 실행하여 파일 시스템을 재확인하십시오. 쓰기 보호가 문제의 원인이 아니거나 재확인을 실패할 경우 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

초기화 단계 fsck 메시지

초기화 단계에서는 명령줄 구문이 확인됩니다. 파일 시스템 검사를 수행할 수 있도록 fsck는 테이블을 설정하고 파일을 엽니다.

이 단원에서 다루는 메시지는 명령줄 옵션, 메모리 요청, 파일 열기, 파일 상태, 파일 시스템 크기 검사 및 스크래치 파일 생성으로 인한 오류 상태와 관련된 것입니다. 파일 시스템을 다듬을 때 해당 초기화 오류가 발생할 경우 fsck가 종료됩니다.

Can't roll the log for *device-name*.

DISCARDING THE LOG MAY DISCARD PENDING TRANSACTIONS.
DISCARD THE LOG AND CONTINUE?

원인

fsck가 파일 시스템에서 오류가 있는지 확인하기 전에 로깅 UFS 파일 시스템의 트랜잭션 로그를 비울 수 없습니다.

조치

예를 선택하면 로그에 있었지만 파일 시스템에 적용되지 않은 파일 시스템 작업이 손실됩니다. 이 경우 fsck는 항상 실행하는 것과 동일한 검사를 실행하고 5단계에서 다음 질문을 묻습니다.

FREE BLK COUNT(S) WRONG IN SUPERBLK (SALVAGE)

이 시점에서 예를 선택하면 로그에 사용되었던 블록이 회수됩니다. 다음 번에 로깅이 사용으로 설정된 상태로 파일 시스템이 마운트되면 로그가 다시 생성됩니다.

아니오를 선택하면 로그가 보존된 후 종료되지만 파일 시스템을 마운트할 수 없습니다.

bad inode number *inode-number* to ginode

원인

inode-number inode가 존재하지 않아 내부 오류가 발생했습니다. fsck가 종료됩니다.

조치

현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

cannot alloc *size-of-block map* bytes for blockmap
cannot alloc *size-of-free map* bytes for freemap
cannot alloc *size-of-state map* bytes for statemap
cannot alloc *size-of-lncntp* bytes for lncntp

원인

내부 테이블에 대한 메모리 요청을 실패했습니다. fsck가 종료됩니다. 이 메시지는 즉시 처리해야 하는 심각한 시스템 오류를 나타냅니다. 다른 프로세스가 매우 많은 양의 시스템 리소스를 사용 중인 경우 이 상태가 발생할 수 있습니다.

조치

다른 프로세스를 강제 종료하면 문제가 해결될 수 있습니다. 그렇지 않을 경우 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

Can't open checklist file: *filename*

원인

파일 시스템의 *filename* 점검 목록 파일(일반적으로 /etc/vfstab)을 읽기 위해 열 수 없습니다. fsck가 종료됩니다.

조치

파일이 존재하며 파일의 액세스 모드가 읽기 액세스를 허용하는지 확인하십시오.

Can't open *filename*

원인

fsck가 파일 시스템의 *filename* 파일을 열 수 없습니다. 대화형으로 실행 중인 경우 fsck가 이 파일 시스템을 무시하고 계속해서 지정된 다음 파일 시스템을 확인합니다.

조치

파일 시스템의 원시 장치 파일에 대한 읽기 및 쓰기 액세스가 허용되는지 확인하십시오.

Can't stat root

원인

루트 디렉토리 통계에 대한 fsck 요청을 실패했습니다. fsck가 종료됩니다.

조치

이 메시지는 심각한 시스템 오류를 나타냅니다. 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

Can't stat *filename*

Can't make sense out of name *filename*

원인

파일 시스템의 *filename* 파일 통계에 대한 fsck 요청을 실패했습니다. 대화형으로 실행 중인 경우 fsck가 이 파일 시스템을 무시하고 계속해서 지정된 다음 파일 시스템을 확인합니다.

조치

파일 시스템이 존재하는지 확인하고 해당 액세스 모드를 확인하십시오.

filename: (NO WRITE)

원인

-n 옵션이 지정되었거나 fsck가 파일 시스템의 *filename* 파일을 쓰기 위해 열 수 없습니다. fsck가 쓰기 권한 없음 모드로 실행 중인 경우 모든 진단 메시지가 표시되지만 fsck가 수정을 시도하지 않습니다.

조치

-n이 지정되지 않은 경우 지정된 파일 유형을 확인하십시오. 정규 파일의 이름일 수 있습니다.

IMPOSSIBLE MINFREE=*percent* IN SUPERBLOCK (SET TO DEFAULT)

원인

수퍼블록 최소 공간 퍼센트가 99%보다 크거나 0%보다 작습니다.

조치

minfree 매개변수를 기본값인 10%로 설정하려면 기본 프롬프트에서 y를 입력하십시오. 오류 상태를 무시하려면 기본 프롬프트에서 n을 입력하십시오.

filename: BAD SUPER BLOCK: *message*
 USE AN ALTERNATE SUPER-BLOCK TO SUPPLY NEEDED INFORMATION;
 e.g., fsck[-f ufs] -o b=# [special ...]
 where # is the alternate superblock. See fsck_ufs(1M)

원인

수퍼블록이 손상되었습니다.

조치

다음 메시지 중 하나가 표시될 수 있습니다.

CPG OUT OF RANGE
 FRAGS PER BLOCK OR FRAGSIZE WRONG
 INODES PER GROUP OUT OF RANGE
 INOPB NONSENSICAL RELATIVE TO BSIZE
 MAGIC NUMBER WRONG
 NCG OUT OF RANGE
 NCYL IS INCONSISTENT WITH NCG*CPG
 NUMBER OF DATA BLOCKS OUT OF RANGE
 NUMBER OF DIRECTORIES OUT OF RANGE
 ROTATIONAL POSITION TABLE SIZE OUT OF RANGE
 SIZE OF CYLINDER GROUP SUMMARY AREA WRONG
 SIZE TOO LARGE
 BAD VALUES IN SUPERBLOCK

대체 수퍼블록을 사용하여 fsck를 재실행해 보십시오. 먼저 블록 32를 지정해 보는 것이 좋습니다. 슬라이스에서 newfs -N 명령을 실행하여 수퍼블록의 대체 복사본을 찾을 수 있습니다. -N 옵션을 지정해야 합니다. 그렇지 않을 경우 newfs가 기존 파일 시스템을 겹쳐줍니다.

UNDEFINED OPTIMIZATION IN SUPERBLOCK (SET TO DEFAULT)

원인

수퍼블록 최적화 매개변수가 OPT_TIME 및 OPT_SPACE가 아닙니다.

조치

파일 시스템에서의 작업 수행 시간을 최소화하려면 SET TO DEFAULT 프롬프트에서 *y*를 입력하십시오. 이 오류 상태를 무시하려면 *n*을 입력하십시오.

1단계: 블록 및 크기 메시지 확인

이 단계에서는 inode 목록을 확인하고, 다음 작업 중 발생한 오류 상태를 보고합니다.

- inode 유형 확인
- 0 링크 수 테이블 설정
- inode 블록 번호에서 잘못되거나 중복된 블록이 있는지 검사
- inode 크기 확인
- inode 형식 확인

파일 시스템을 다듬을 때 INCORRECT BLOCK COUNT, PARTIALLY TRUNCATED INODE, PARTIALLY ALLOCATED INODE 및 UNKNOWN FILE TYPE을 제외하고 이 단계에서 오류가 발생할 경우 fsck가 종료됩니다.

이러한 메시지(영문자순)는 1단계에서 발생할 수 있습니다.

block-number BAD I=*inode-number*

원인

inode-number inode에 포함된 *block-number* 블록 번호가 파일 시스템의 첫번째 데이터 블록 번호보다 작거나 파일 시스템의 마지막 블록 번호보다 큼니다. *inode-number* inode에 블록 번호가 너무 많아 파일 시스템 범위를 벗어날 경우 이 오류 상태로 인해 1단계에서 EXCESSIVE BAD BLKS 오류 메시지가 생성될 수 있습니다. 이 오류 상태는 2단계 및 4단계에서 BAD/DUP 오류 메시지를 생성합니다.

조치

해당 없음

BAD MODE: MAKE IT A FILE?

원인

제공된 inode의 상태가 모두 1s로 설정되어 있습니다. 이는 파일 시스템 손상을 나타냅니다. fsck -y가 실행된 후 이 메시지가 반복적으로 표시되지 않는 경우 물리적인 디스크 손상을 나타내는 것이 아닙니다.

조치

*y*를 입력하여 inode를 적합한 값으로 다시 초기화하십시오.

BAD STATE *state-number* TO BLKERR

원인

내부 오류로 인해 fsck 상태 맵이 손상되어 불가능한 *state-number* 값이 표시됩니다. fsck가 즉시 종료됩니다.

조치

현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

fragment-number DUP I=*inode-number*

Oracle Solaris 10:

block-number DUP I=*inode-number*

원인

inode-number inode에 포함된 *fragment-number* 블록 번호가 동일한 inode 또는 다른 inode에 의해 이미 확보되었습니다. *inode-number* inode에 동일한 inode 또는 다른 inode에 의해 확보된 블록 번호가 너무 많을 경우 이 오류 상태로 인해 1단계에서 EXCESSIVE DUP BLKS 오류 메시지가 생성될 수 있습니다. 이 오류 상태는 1B 단계를 호출하고 2단계 및 4단계에서 BAD/DUP 오류 메시지를 생성합니다.

Oracle Solaris 10: *inode-number* inode에 포함된 *block-number* 블록 번호가 동일한 inode 또는 다른 inode에 의해 이미 확보되었습니다. *inode-number* inode에 동일한 inode 또는 다른 inode에 의해 확보된 블록 번호가 너무 많을 경우 이 오류 상태로 인해 1단계에서 EXCESSIVE DUP BLKS 오류 메시지가 생성될 수 있습니다. 이 오류 상태는 1B 단계를 호출하고 2단계 및 4단계에서 BAD/DUP 오류 메시지를 생성합니다.

조치

해당 없음

DUP TABLE OVERFLOW (CONTINUE)

원인

fsck가 중복된 조각을 추적하는 데 필요한 메모리를 할당할 수 없습니다. -o p 옵션이 지정된 경우 프로그램이 종료됩니다.

Oracle Solaris 10: 중복된 블록 번호가 포함된 fsck의 내부 테이블에 더 이상 공간이 없습니다. -o p 옵션이 지정된 경우 프로그램이 종료됩니다.

조치

프로그램을 계속하려면 CONTINUE 프롬프트에서 y를 입력하십시오. 이 오류가 발생할 경우 파일 시스템 전체 검사를 실행할 수 없습니다. 중복된 다른 조각이 있을 경우 이 오류 상태가 반복됩니다. 일부 프로세스를 강제 종료한 후 스왑 공간을 늘려 사용 가능한 가상 메모리 공간을 늘리고 fsck를 다시 실행하여 파일 시스템을 재확인하십시오. 프로그램을 종료하려면 n을 입력하십시오.

Oracle Solaris 10: 프로그램을 계속하려면 CONTINUE 프롬프트에서 y를 입력하십시오. 이 오류가 발생할 경우 파일 시스템 전체 검사를 실행할 수 없습니다. 중복된 다른 블록이 있을 경우 이 오류 상태가 반복됩니다. 일부 프로세스를 강제 종료한 후 스왑 공간을 늘려 사용 가능한 가상 메모리 공간을 늘리고 fsck를 다시 실행하여 파일 시스템을 재확인하십시오. 프로그램을 종료하려면 n을 입력하십시오.

EXCESSIVE BAD FRAGMENTS I=*inode-number* (CONTINUE)

Oracle Solaris 10:

EXCESSIVE BAD BLOCKS I=inode-number (CONTINUE)

원인

조각이 너무 많으면(일반적으로 11개 이상) 디스크 주소가 잘못된 것입니다. -o p(다듬기) 옵션이 지정된 경우 프로그램이 종료됩니다.

Oracle Solaris 10: 너무 많은(일반적으로 11개 이상) 블록의 번호가 파일 시스템의 첫번째 데이터 블록 번호보다 작거나 *inode-number* inode에 연결된 파일 시스템의 마지막 블록 번호보다 큼니다. -o p(다듬기) 옵션이 지정된 경우 프로그램이 종료됩니다.

조치

프로그램을 계속하려면 CONTINUE 프롬프트에서 y를 입력하십시오. 이 오류가 발생할 경우 파일 시스템 전체 검사를 실행할 수 없습니다. fsck를 다시 실행하여 파일 시스템을 재확인해야 합니다. 프로그램을 종료하려면 n을 입력하십시오.

EXCESSIVE DUP BLKSDUPLICATE FRAGMENTS I=inode-number (CONTINUE)

Oracle Solaris 10:

EXCESSIVE DUP BLKS I=inode-number (CONTINUE)

원인

너무 많은(일반적으로 11개 이상) 조각이 동일한 inode 또는 다른 inode나 사용 가능한 목록에 의해 확보되었습니다. -o p 옵션이 지정된 경우 프로그램이 종료됩니다.

Oracle Solaris 10: 너무 많은(일반적으로 11개 이상) 블록이 동일한 inode 또는 다른 inode나 사용 가능한 목록에 의해 확보되었습니다. -o p 옵션이 지정된 경우 프로그램이 종료됩니다.

조치

프로그램을 계속하려면 CONTINUE 프롬프트에서 y를 입력하십시오. 이 오류가 발생할 경우 파일 시스템 전체 검사를 실행할 수 없습니다. fsck를 다시 실행하여 파일 시스템을 재확인해야 합니다. 프로그램을 종료하려면 n을 입력하십시오.

INCORRECT DISK BLOCK COUNT I=inode-number (*number-of-BAD-DUP-or-missing-blocks* should be *number-of-blocks-in-filesystem*) (CORRECT)**Oracle Solaris 10:**INCORRECT BLOCK COUNT I=inode-number (*number-of-BAD-DUP-or-missing-blocks* should be *number-of-blocks-in-filesystem*) (CORRECT)

원인

inode-number inode에 대한 디스크 블록 수가 잘못되었습니다. 다듬기 과정 중 fsck가 개수를 수정합니다.

Oracle Solaris 10: *inode-number* inode에 대한 블록 수가 *number-of-BAD-DUP-or-missing-blocks*개이지만 *number-of-blocks-in-filesystem*개여야 합니다. 다듬기 과정 중 fsck가 개수를 수정합니다.

조치

inode-number inode의 디스크 블록 수를 *number-of-blocks-in-file*개로 수정하려면 CORRECT 프롬프트에서 y를 입력하십시오.

Oracle Solaris 10: *inode-number* inode의 블록 수를 *number-of-blocks-in-filesystem*개로 바꾸려면 CORRECT 프롬프트에서 y를 입력하십시오. 프로그램을 종료하려면 n을 입력하십시오.

LINK COUNT TABLE OVERFLOW (CONTINUE)

원인

링크 수가 0인 할당된 inode를 포함하는 fsck에 대한 내부 테이블에 더 이상 공간이 없습니다. -o p(다듬기) 옵션이 지정된 경우 프로그램이 종료되므로 fsck를 수동으로 완료해야 합니다.

조치

프로그램을 계속하려면 CONTINUE 프롬프트에서 y를 입력하십시오. 링크 수가 0인 할당된 다른 inode가 있을 경우 이 오류 상태가 반복됩니다. 이 오류가 발생할 경우 파일 시스템 전체 검사를 실행할 수 없습니다. fsck를 다시 실행하여 파일 시스템을 재확인해야 합니다. 일부 프로세스를 강제 종료하거나 스왑 공간을 늘려 사용 가능한 가상 메모리를 늘린 후 fsck를 다시 실행하십시오. 프로그램을 종료하려면 n을 입력하십시오.

PARTIALLY ALLOCATED INODE I=*inode-number* (CLEAR)

원인

inode-number inode가 할당되거나 할당 취소되지 않았습니다. -o p(다듬기) 옵션이 지정된 경우 inode가 지워집니다.

조치

해당 콘텐츠를 0으로 설정하여 *inode-number* inode의 할당을 해제하려면 y를 입력하십시오. 그러면 2단계에서 이 inode를 가리키는 각 디렉토리 항목에 대해 UNALLOCATED 오류 상태가 발생할 수 있습니다. 이 오류 상태를 무시하려면 n을 입력하십시오. 다른 측정 단위를 사용하여 문제를 해결하려는 경우에만 n(아니오)을 선택할 수 있습니다.

PARTIALLY TRUNCATED INODE I=*inode-number* (SALVAGE)

원인

fsck가 발견한 *inode-number* inode의 크기가 할당된 조각 수보다 작습니다. 이 상태는 파일을 자르는 동안 시스템이 충돌하는 경우에만 발생합니다. 파일 시스템을 다듬을 때 fsck가 지정된 크기로의 자르기 작업을 완료합니다.

Oracle Solaris 10: fsck가 발견한 *inode-number* inode의 크기가 할당된 블록 수보다 작습니다. 이 상태는 파일을 자르는 동안 시스템이 충돌하는 경우에만 발생합니다. 파일 시스템을 다듬을 때 fsck가 지정된 크기로의 자르기 작업을 완료합니다.

조치

inode에 지정된 크기로의 자르기 작업을 완료하려면 SALVAGE 프롬프트에서 y를 입력하십시오. 이 오류 상태를 무시하려면 n을 입력하십시오.

UNKNOWN FILE TYPE I=*inode-number* (CLEAR)

원인

inode-number inode의 모드에 inode가 파이프, 문자 장치, 블록 장치, 정규 파일, 심볼릭 링크, FIFO 파일 또는 디렉토리 inode가 아닌 것으로 표시됩니다. -op 옵션이 지정된 경우 inode가 지워집니다.

Oracle Solaris 10: *inode-number* inode의 모드에 inode가 파이프, 특수 문자 inode, 특수 블록 inode, 정규 inode, 심볼릭 링크, FIFO 파일 또는 디렉토리 inode가 아닌 것으로 표시됩니다. -op 옵션이 지정된 경우 inode가 지워집니다.

조치

해당 콘텐츠를 0으로 설정하여 *inode-number* inode의 할당을 해제하려면 CLEAR 프롬프트에서 y를 입력하십시오. 그러면 2단계에서 이 inode를 가리키는 각 디렉토리 항목에 대해 UNALLOCATED 오류 상태가 발생합니다. 이 오류 상태를 무시하려면 n을 입력하십시오.

Oracle Solaris 10: 1B단계: 추가 DUPS 메시지 재검색

이 단원에서는 현재 릴리스의 1B단계 fsck 메시지를 다룹니다.

파일 시스템에 중복된 조각이 있을 경우 다음 메시지가 표시됩니다.

fragment DUP I=*inode-number*

원인

inode-number inode에 포함된 *fragment-number* 조각 번호가 동일한 inode 또는 다른 inode에 의해 이미 확보되었습니다. 이 오류 상태는 2단계에서 BAD/DUP 오류 메시지를 생성합니다. 이 오류 상태 및 1단계의 DUP 오류 상태를 검사하여 조각이 겹치는 inode를 확인할 수 있습니다. fsck 실행에서 생성되는 중복 조각 보고서를 통해 이 과정이 간소화됩니다.

조치

중복된 블록이 있을 경우 파일 시스템이 재검색되어 이전에 해당 블록을 확보한 inode가 검색됩니다.

1B단계: 추가 DUPS 메시지 재검색

이 단계에서는 Oracle Solaris 10 릴리스 및 지원되는 기타 Solaris 릴리스에 표시되는 fsck 메시지를 다룹니다.

파일 시스템에 중복된 블록이 있을 경우 다음 메시지가 표시됩니다.

```
block-number DUP I=inode-number
```

원인

inode-number inode에 포함된 *block-number* 블록 번호가 동일한 inode 또는 다른 inode에 의해 이미 확보되었습니다. 이 오류 상태는 2단계에서 BAD/DUP 오류 메시지를 생성합니다. 이 오류 상태 및 1단계의 DUP 오류 상태를 검사하여 블록이 겹치는 inode를 확인할 수 있습니다.

조치

중복된 블록이 있을 경우 파일 시스템이 재검색되어 이전에 해당 블록을 확보한 inode가 검색됩니다.

2단계: 경로 이름 메시지 확인

이 단계에서는 1단계 및 1B단계에서 발견된 잘못된 inode를 가리키는 디렉토리 항목을 제거합니다. 또한 다음으로 인한 오류 상태를 보고합니다.

- 잘못된 루트 inode 모드 및 상태
- 디렉토리 inode 포인터가 범위를 벗어남
- 디렉토리 항목이 잘못된 inode를 가리킴
- 디렉토리 무결성 검사

-o -p 옵션을 통해 파일 시스템을 다듬을 때 이 단계에서 오류가 발생할 경우 fsck가 종료됩니다. 단, 디렉토리가 블록 크기의 배수가 아니거나 블록이 중복 및 잘못되거나 inode가 범위를 벗어나거나 하드 링크가 잘못되어 오류가 발생하는 경우에는 종료되지 않습니다.

이러한 메시지(영문자순)는 2단계에서 발생할 수 있습니다.

```
BAD INODE state-number TO DESCEND
```

원인

fsck 내부 오류로 인해 파일 시스템 디렉토리 구조의 상위인 루틴에 잘못된 *state-number* 상태가 전달되었습니다. fsck가 종료됩니다.

조치

이 오류 메시지가 표시되면 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

```
BAD INODE NUMBER FOR '.' I=inode-number OWNER=UID MODE=file-mode  
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

원인

발견된 디렉토리 *inode-number*의 "." inode 번호와 *inode-number* inode 번호가 다릅니다.

조치

"." inode 번호를 *inode-number* inode 번호와 같아지도록 변경하려면 FIX 프롬프트에서 y를 입력하고, "." inode 번호를 변경하지 않은 상태로 두려면 n을 입력하십시오.

```
BAD INODE NUMBER FOR '..' I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

원인

발견된 *inode-number*의 ".." inode 번호가 *inode-number*의 상위와 다릅니다.

조치

".." inode 번호를 *inode-number*의 상위가 되도록 변경하려면 FIX 프롬프트에서 y를 입력하십시오. (루트 inode의 ".."는 자신을 가리킵니다.) ".." inode 번호를 변경하지 않은 상태로 두려면 n을 입력하십시오.

```
BAD RETURN STATE state-number FROM DESCEND
```

원인

fsck 내부 오류로 인해 파일 시스템 디렉토리 구조의 상위인 루틴에서 불가능한 *state-number* 상태가 반환되었습니다. fsck가 종료됩니다.

조치

이 메시지가 표시되면 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

```
BAD STATE state-number FOR ROOT INODE
```

원인

내부 오류로 인해 불가능한 *state-number* 상태가 루트 inode에 지정되었습니다. fsck가 종료됩니다.

조치

이 오류 메시지가 표시되면 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

```
BAD STATE state-number FOR INODE=inode-number
```

원인

내부 오류로 인해 불가능한 *state-number* 상태가 *inode-number* inode에 지정되었습니다. fsck가 종료됩니다.

조치

이 오류 메시지가 표시되면 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

```
DIRECTORY TOO SHORT I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

원인

발견된 디렉토리 *filename*의 *file-size* 크기가 최소 디렉토리 크기보다 작습니다. *UID* 소유자, *file-mode* 모드, *file-size* 크기, *modification-time* 수정 시간 및 *filename* 디렉토리 이름이 표시됩니다.

조치

디렉토리 크기를 최소 디렉토리 크기로 늘리려면 **FIX** 프롬프트에서 *y*를 입력하십시오. 이 디렉토리를 무시하려면 *n*을 입력하십시오.

DIRECTORY *filename*: LENGTH *file-size* NOT MULTIPLE OF *disk-block-size* (ADJUST)

Oracle Solaris 10:

DIRECTORY *filename*: LENGTH *file-size* NOT MULTIPLE OF *block-number* (ADJUST)

원인

발견된 디렉토리 *filename*의 *file-size* 크기가 *disk-block-size* 디렉토리 블록 크기의 배수가 아닙니다.

Oracle Solaris 10:

발견된 디렉토리 *filename*의 *file-size* 크기가 *block-number* 디렉토리 블록 크기의 배수가 아닙니다.

조치

길이를 적합한 디스크 블록 크기로 반올림하려면 *y*를 입력하십시오. **-op** 옵션을 통해 파일 시스템을 다듬을 때 **fsck**가 경고만 표시하고 디렉토리를 조정합니다. 이 상태를 무시하려면 *n*을 입력하십시오.

Oracle Solaris 10:

길이를 적합한 블록 크기로 반올림하려면 *y*를 입력하십시오. **-op** 옵션을 통해 파일 시스템을 다듬을 때 **fsck**가 경고만 표시하고 디렉토리를 조정합니다. 이 상태를 무시하려면 *n*을 입력하십시오.

DIRECTORY CORRUPTED I=*inode-number* OWNER=*UID* MODE=*file-mode*
SIZE=*file-size* MTIME=*modification-time* DIR=*filename* (SALVAGE)

원인

내부 상태가 일치하지 않는 디렉토리가 있습니다.

조치

다음 디렉토리 경계(일반적으로 512바이트 경계)까지 모든 항목을 제거하려면 **SALVAGE** 프롬프트에서 *y*를 입력하십시오. 이 과감한 조치를 수행하면 최대 42개의 항목을 제거할 수 있습니다. 다른 복구 작업을 실패한 후에만 이 조치를 수행하십시오. 다음 디렉토리 경계로 건너뛴 후 읽기를 계속하되 디렉토리를 수정하지 않으려면 *n*을 입력하십시오.

DUP/BAD I=*inode-number* OWNER=0 MODE=M SIZE=*file-size*
MTIME=*modification-time* TYPE=*filename* (REMOVE)

원인

1단계 또는 1B단계를 통해 *filename* 디렉토리 또는 파일 항목, *inode-number* inode에 중복된 조각 또는 잘못된 조각이 연결되어 있는 것으로 확인되었습니다. *UID* 소유자, *file-mode* 모드, *file-size* 크기, *modification-time* 수정 시간 및 *filename* 디렉토리 또는 파일 이름이 표시됩니다. -op(다듬기) 옵션이 지정된 경우 중복/잘못된 조각이 제거됩니다.

Oracle Solaris 10:

1단계 또는 1B단계를 통해 *filename* 디렉토리 또는 파일 항목, *inode-number* inode에 중복된 블록 또는 잘못된 블록이 연결되어 있는 것으로 확인되었습니다. *UID* 소유자, *file-mode* 모드, *file-size* 크기, *modification-time* 수정 시간 및 *filename* 디렉토리 또는 파일 이름이 표시됩니다. -op(다듬기) 옵션이 지정된 경우 중복/잘못된 블록이 제거됩니다.

조치

filename 디렉토리 또는 파일 항목을 제거하려면 REMOVE 프롬프트에서 y를 입력하십시오. 이 오류 상태를 무시하려면 n을 입력하십시오.

DUPS/BAD IN ROOT INODE (REALLOCATE)**원인**

1단계 또는 1B단계를 통해 파일 시스템의 루트 inode(inode 번호 20)에 중복된 조각 또는 잘못된 조각이 있는 것으로 확인되었습니다.

Oracle Solaris 10:

1단계 또는 1B단계를 통해 파일 시스템의 루트 inode(일반적으로 inode 번호 2)에 중복된 블록 또는 잘못된 블록이 있는 것으로 확인되었습니다.

조치

루트 inode의 기존 콘텐츠를 지운 후 재할당하려면 REALLOCATE 프롬프트에서 y를 입력하십시오. 그러면 루트 inode에서 일반적으로 발견되는 파일 및 디렉토리가 3단계에서 복구되어 lost+found 디렉토리에 삽입됩니다. 루트 할당 시도를 실패하면 CANNOT ALLOCATE ROOT INODE와 함께 fsck가 종료됩니다. CONTINUE 프롬프트를 표시하려면 n을 입력하십시오. y를 입력하여 CONTINUE 프롬프트에 응답하고 루트 inode의 DUPS/BAD 오류 상태를 무시한 후 파일 시스템 검사 실행을 계속하십시오. 루트 inode가 잘못된 경우 이로 인해 여러 가지 다른 오류 메시지가 생성될 수 있습니다. 프로그램을 종료하려면 n을 입력하십시오.

```
EXTRA '.' ENTRY I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time DIR=filename (FIX)
```

원인

발견된 디렉토리 *inode-number*에 "."에 대한 항목이 두 개 이상 있습니다.

조치

"."에 대한 추가 항목을 제거하려면 FIX 프롬프트에서 y를 입력하십시오. 디렉토리를 변경하지 않은 상태로 두려면 n을 입력하십시오.

EXTRA '..' ENTRY I=*inode-number* OWNER=*UID* MODE=*file-mode*
 SIZE=*file-size* MTIME=*modification-time* DIR=*filename* (FIX)

원인

발견된 디렉토리 *inode-number*에 ".."(상위 디렉토리)에 대한 항목이 두 개 이상 있습니다.

조치

'..'(상위 디렉토리)에 대한 추가 항목을 제거하려면 FIX 프롬프트에서 y를 입력하십시오. 디렉토리를 변경하지 않은 상태로 두려면 n을 입력하십시오.

hard-link-number IS AN EXTRANEIOUS HARD LINK TO A DIRECTORY *filename* (REMOVE)

원인

fsck를 통해 *hard-link-number* 하드 링크가 디렉토리 *filename*에 잘못 연결되어 있는 것으로 확인되었습니다. -op 옵션을 통해 다듬을 때 fsck가 잘못된 하드 링크를 무시합니다.

조치

잘못된 *hard-link-number* 항목을 삭제하려면 REMOVE 프롬프트에서 y를 입력하십시오. 이 오류 상태를 무시하려면 n을 입력하십시오.

inode-number OUT OF RANGE I=*inode-number* NAME=*filename* (REMOVE)

원인

filename 디렉토리 항목의 *inode-number* inode 번호가 inode 목록 끝보다 큼니다. -p(다듬기) 옵션이 지정된 경우 자동으로 inode가 제거됩니다.

조치

filename 디렉토리 항목을 삭제하려면 REMOVE 프롬프트에서 y를 입력하십시오. 이 오류 상태를 무시하려면 n을 입력하십시오.

MISSING '..' I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
 MTIME=*modification-time* DIR=*filename* (FIX)

원인

발견된 디렉토리 *inode-number*의 첫번째 항목("."에 대한 항목)이 할당되지 않았습니니다.

조치

inode 번호가 *inode-number*인 "."에 대한 항목을 작성하려면 FIX 프롬프트에서 y를 입력하십시오. 디렉토리를 변경하지 않은 상태로 두려면 n을 입력하십시오.

MISSING '..' I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
 MTIME=*modification-time* DIR=*filename* CANNOT FIX, FIRST ENTRY IN
 DIRECTORY CONTAINS *filename*

원인

발견된 디렉토리 *inode-number*의 첫번째 항목이 *filename*입니다. fsck가 이 문제를 해결할 수 없습니다.

조치

이 오류 메시지가 표시되면 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, INSUFFICIENT
SPACE TO ADD '..'
```

원인

발견된 *inode-number*의 첫번째 항목이 "." 항목이 아닙니다. fsck가 문제를 해결할 수 없습니다.

조치

이 오류 메시지가 표시되면 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename (FIX)
```

원인

발견된 디렉토리 *inode-number*의 두번째 항목이 할당되지 않았습니다.

조치

"."에 대한 항목(*inode* 번호: *inode-number*)을 작성하려면 FIX 프롬프트에서 y를 입력하십시오. (루트 *inode*의 "."는 자신을 가리킵니다.) 디렉토리를 변경하지 않은 상태로 두려면 n을 입력하십시오.

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, SECOND ENTRY IN
DIRECTORY CONTAINS filename
```

원인

발견된 디렉토리 *inode-number*의 두번째 항목이 *filename*입니다. fsck가 이 문제를 해결할 수 없습니다.

조치

이 오류 메시지가 표시되면 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

```
MISSING '..' I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time DIR=filename CANNOT FIX, INSUFFICIENT SPACE
TO ADD '..'
```

원인

발견된 디렉토리 *inode-number*의 두번째 항목이 ".."(상위 디렉토리)에 대한 항목이 두 개 이상 있습니다. fsck가 이 문제를 해결할 수 없습니다.

조치

이 오류 메시지가 표시되면 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

```
NAME TOO LONG filename
```

원인

너무 긴 경로 이름이 발견되었습니다. 일반적으로 이는 파일 시스템 네임스페이스에 루프가 있음을 나타냅니다. 이 오류는 권한이 있는 사용자가 디렉토리에 대한 순환 링크를 만든 경우 발생할 수 있습니다.

조치

순환 링크를 제거하십시오.

ROOT INODE UNALLOCATED (ALLOCATE)

원인

루트 inode(일반적으로 inode 번호 2)에 할당 모드 비트가 없습니다.

조치

inode 2를 루트 inode로 할당하려면 ALLOCATE 프롬프트에서 *y*를 입력하십시오. 그러면 루트 inode에서 일반적으로 발견되는 파일 및 디렉토리가 3단계에서 복구되어 *lost+found* 디렉토리에 삽입됩니다. 루트 inode 할당 시도를 실패하면 fsck가 CANNOT ALLOCATE ROOT INODE 메시지를 표시하고 종료됩니다. 프로그램을 종료하려면 *n*을 입력하십시오.

ROOT INODE NOT DIRECTORY (REALLOCATE)

원인

파일 시스템의 루트 inode(일반적으로 inode 번호 2)가 디렉토리 inode가 아닙니다.

조치

루트 inode의 기존 콘텐츠를 지운 후 재할당하려면 REALLOCATE 프롬프트에서 *y*를 입력하십시오. 그러면 루트 inode에서 일반적으로 발견되는 파일 및 디렉토리가 3단계에서 복구되어 *lost+found* 디렉토리에 삽입됩니다. 루트 inode 할당 시도를 실패하면 fsck가 CANNOT ALLOCATE ROOT INODE 메시지를 표시하고 종료됩니다. fsck가 FIX 프롬프트를 표시하도록하려면 *n*을 입력하십시오.

UNALLOCATED I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
MTIME=*modification-time* type=*filename*(REMOVE)

원인

filename 디렉토리 또는 파일 항목이 할당되지 않은 *inode-number* inode를 가리킵니다. *UID* 소유자, *file-mode* 모드, *file-size* 크기, *modification-time* 수정 시간 및 *filename* 파일 이름이 표시됩니다.

조치

filename 디렉토리 항목을 삭제하려면 REMOVE 프롬프트에서 *y*를 입력하십시오. 이 오류 상태를 무시하려면 *n*을 입력하십시오.

ZERO LENGTH DIRECTORY I=*inode-number* OWNER=*UID* MODE=*file-mode*
SIZE=*file-size* MTIME=*modification-time* DIR=*filename* (REMOVE)

원인

filename 디렉토리 항목의 *file-size* 크기가 0입니다. *UID* 소유자, *file-mode* 모드, *file-size* 크기, *modification-time* 수정 시간 및 *filename* 디렉토리 이름이 표시됩니다.

조치

filename 디렉토리 항목을 제거하려면 REMOVE 프롬프트에서 *y*를 입력하십시오. 그러면 4단계에서 BAD/DUP 오류 메시지가 표시됩니다. 이 오류 상태를 무시하려면 *n*을 입력하십시오.

3단계: 연결 메시지 확인

이 단계에서는 2단계에서 검사한 디렉토리를 확인하고 다음으로 인한 오류 상태를 보고합니다.

- 참조되지 않은 디렉토리
- 누락되거나 전체적으로 표시된 lost+found 디렉토리

이러한 메시지(영문자순)는 3단계에서 발생할 수 있습니다.

BAD INODE *state-number* TO DESCEND

원인

내부 오류로 인해 불가능한 *state-number* 상태가 파일 시스템 디렉토리 구조의 상위인 루트에 전달되었습니다. fsck가 종료됩니다.

조치

이 오류가 발생할 경우 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

DIR I=*inode-number1* CONNECTED. PARENT WAS I=*inode-number2*

원인

이는 *inode-number1* 디렉토리 *inode*가 lost+found 디렉토리에 성공적으로 연결되었음을 나타내는 권고 메시지입니다. *inode-number1* 디렉토리 *inode*의 *inode-number2* 상위 *inode*가 lost+found 디렉토리의 *inode* 번호로 바뀝니다.

조치

해당 없음

DIRECTORY *filename* LENGTH *file-size* NOT MULTIPLE OF *disk-block-size* (ADJUST)

Oracle Solaris 10:

DIRECTORY *filename* LENGTH *file-size* NOT MULTIPLE OF *block-number* (ADJUST)

원인

발견된 디렉토리 *filename*의 *file-size* 크기가 디렉토리 블록 크기 *B*의 배수가 아닙니다. (2단계에서 조정되지 않을 경우 3단계에서 이 상태가 다시 발생할 수 있습니다.)

조치

길이를 적합한 디스크 블록 크기로 반올림하려면 ADJUST 프롬프트에서 *y*를 입력하십시오. 다듬기 과정 중 fsck가 경고를 표시하고 디렉토리를 조정합니다. 이 오류 상태를 무시하려면 *n*을 입력하십시오.

Oracle Solaris 10:

길이를 적합한 블록 크기로 반올림하려면 **ADJUST** 프롬프트에서 **y**를 입력하십시오. 다듬기 과정 중 **fsck**가 경고를 표시하고 디렉토리를 조정합니다. 이 오류 상태를 무시하려면 **n**을 입력하십시오.

lost+found IS NOT A DIRECTORY (REALLOCATE)

원인

lost+found에 대한 항목이 디렉토리가 아닙니다.

조치

디렉토리 **inode**를 할당하고 이 디렉토리 **inode**를 참조하도록 **lost+found** 디렉토리를 변경하려면 **REALLOCATE** 프롬프트에서 **y**를 입력하십시오. 그러면 **lost+found** 디렉토리에 의한 이전 **inode** 참조가 지워지지 않고 참조되지 않은 **inode**로 회수되거나 이 단계의 뒷부분에서 해당 링크 수가 조정됩니다. **lost+found** 디렉토리를 만들 수 없을 경우 **SORRY. CANNOT CREATE lost+found DIRECTORY** 메시지가 표시되고 손실된 **inode** 연결 시도가 중지되어 4단계에서 **UNREF** 오류 메시지가 생성됩니다. 4단계에서 **UNREF** 오류 메시지가 생성되지 않도록 손실된 **inode** 연결 시도를 중지하려면 **n**을 입력하십시오.

NO lost+found DIRECTORY (CREATE)

원인

파일 시스템의 루트 디렉토리에 **lost+found** 디렉토리가 없습니다. 다듬기 과정 중 **fsck**가 **lost+found** 디렉토리를 만들려고 시도합니다.

조치

파일 시스템의 루트에 **lost+found** 디렉토리를 만들려면 **CREATE** 프롬프트에서 **y**를 입력하십시오. 그러면 **NO SPACE LEFT IN / (EXPAND)** 메시지가 표시될 수 있습니다. **lost+found** 디렉토리를 만들 수 없을 경우 **fsck**가 **SORRY. CANNOT CREATE lost+found DIRECTORY** 메시지를 표시하고 손실된 **inode** 연결 시도를 중지합니다. 이 오류는 4단계의 뒷부분에서 **UNREF** 오류 메시지를 생성합니다. 손실된 **inode** 연결 시도를 중지하려면 **n**을 입력하십시오.

NO SPACE LEFT IN /lost+found (EXPAND)

원인

사용 가능한 공간이 없어 파일 시스템 루트 디렉토리의 **lost+found** 디렉토리에 다른 항목을 추가할 수 없습니다. 다듬기 과정 중 **fsck**가 **lost+found** 디렉토리를 확장합니다.

조치

lost+found 디렉토리를 확장하여 새 항목에 대한 공간을 확보하려면 **EXPAND** 프롬프트에서 **y**를 입력하십시오. 확장 시도를 실패하면 **fsck**가 **SORRY. NO SPACE IN lost+found DIRECTORY**를 표시하고 **lost+found** 디렉토리에 대한 파일 연결 요청을 중지합니다. 이 오류는 4단계의 뒷부분에서 **UNREF** 오류 메시지를 생성합니다. **lost+found** 디렉토리에 불필요한 항목을 삭제하십시오. 이 오류가 발생할 경우

다듬기 옵션이 적용될 때 fsck가 종료됩니다. 손실된 inode 연결 시도를 중지하려면 n을 입력하십시오.

UNREF DIR I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
 MTIME=*modification-time* (RECONNECT)

원인

파일 시스템이 통과될 때 *inode-number* 디렉토리 inode가 디렉토리 항목에 연결되지 않았습니다. *inode-number* 디렉토리 inode의 *UID* 소유자, *file-mode* 모드, *file-size* 크기 및 *modification-time* 수정 시간이 표시됩니다. 디렉토리 크기가 0이 아닌 경우 다듬기 과정 중 fsck가 비어 있지 않은 디렉토리 inode에 다시 연결합니다. 그렇지 않은 경우 fsck가 디렉토리 inode를 지웁니다.

조치

inode-number 디렉토리 inode를 *lost+found* 디렉토리에 다시 연결하려면 RECONNECT 프롬프트에서 y를 입력하십시오. 디렉토리가 성공적으로 다시 연결되는 경우 CONNECTED 메시지가 표시됩니다. 그렇지 않은 경우 *lost+found* 오류 메시지 중 하나가 표시됩니다. 이 오류 상태를 무시하려면 n을 입력하십시오. 이 오류로 인해 4단계에서 UNREF 오류 상태가 발생합니다.

4단계: 참조 수 메시지 확인

이 단계에서는 2단계 및 3단계에서 가져온 링크 수 정보를 확인합니다. 또한 다음으로 인한 오류 상태를 보고합니다.

- 참조되지 않은 파일
- 누락되거나 전체적으로 표시된 *lost+found* 디렉토리
- 파일, 디렉토리, 심볼릭 링크 또는 특수 파일에 대해 잘못된 링크 수
- 참조되지 않은 파일, 심볼릭 링크 및 디렉토리
- 파일 및 디렉토리의 잘못되거나 중복된 조각

Oracle Solaris 10:

파일 및 디렉토리의 잘못되거나 중복된 블록

- 잘못된 총 사용 가능한 inode 수

이 단계에서 발생한 모든 오류(*lost+found* 디렉토리의 공간 부족 오류 제외)는 파일 시스템을 다듬을 때 수정할 수 있습니다.

이러한 메시지(영문자순)는 4단계에서 발생할 수 있습니다.

BAD/DUP type I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
 MTIME=*modification-time* (CLEAR)

원인

1단계 또는 1B단계를 통해 *inode-number* 파일 또는 디렉토리 *inode*에 중복된 조각 또는 잘못된 조각이 연결되어 있는 것으로 확인되었습니다. *inode-number* *inode*의 *UID* 소유자, *file-mode* 모드, *file-size* 크기 및 *modification-time* 수정 시간이 표시됩니다.

Oracle Solaris 10:

1단계 또는 1B단계를 통해 *inode-number* 파일 또는 디렉토리 *inode*에 중복된 블록 또는 잘못된 블록이 연결되어 있는 것으로 확인되었습니다. *inode-number* *inode*의 *UID* 소유자, *file-mode* 모드, *file-size* 크기 및 *modification-time* 수정 시간이 표시됩니다.

조치

해당 콘텐츠를 0으로 설정하여 *inode-number* *inode*의 할당을 해제하려면 CLEAR 프롬프트에서 y를 입력하십시오. 이 오류 상태를 무시하려면 n을 입력하십시오.

(CLEAR)

원인

바로 앞의 UNREF 오류 메시지에 언급된 *inode*에 다시 연결할 수 없습니다. 파일에 다시 연결하는 데 필요한 공간이 부족하면 fsck가 종료되므로 파일 시스템을 다듬는 중인 경우 이 메시지가 표시되지 않습니다.

조치

해당 콘텐츠를 0으로 설정하여 *inode*의 할당을 해제하려면 CLEAR 프롬프트에서 y를 입력하십시오. 앞선 오류 상태를 무시하려면 n을 입력하십시오.

```
LINK COUNT type I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size
MTIME=modification-time COUNT link-count SHOULD BE
corrected-link-count (ADJUST)
```

원인

inode-number 디렉토리 또는 파일 *inode*에 대한 링크 수가 *link-count*개이지만 *corrected-link-count*개여야 합니다. *inode-number* *inode*의 *UID* 소유자, *file-mode* 모드, *file-size* 크기 및 *modification-time* 수정 시간이 표시됩니다. -o p 옵션이 지정된 경우 참조 수가 계속 증가하지 않는 한 링크 수가 조정됩니다. 하드웨어 오류가 없을 경우 이 상태는 발생하지 않습니다. 다듬기 과정에서 참조 수가 계속 증가하는 경우 fsck가 LINK COUNT INCREASING 메시지를 표시한 후 종료됩니다.

조치

inode-number 디렉토리 또는 파일 *inode*의 링크 수를 *corrected-link-count*개로 바꾸려면 ADJUST 프롬프트에서 y를 입력하십시오. 이 오류 상태를 무시하려면 n을 입력하십시오.

```
lost+found IS NOT A DIRECTORY (REALLOCATE)
```

원인

lost+found에 대한 항목이 디렉토리가 아닙니다.

조치

디렉토리 inode를 할당하고 이 디렉토리 inode를 참조하도록 lost+found 디렉토리를 변경하려면 REALLOCATE 프롬프트에서 *y*를 입력하십시오. lost+found 디렉토리에 의한 이전 inode 참조는 지워지지 않습니다. 참조되지 않은 inode로 회수되거나 이 단계의 뒷부분에서 해당 링크 수가 조정됩니다. lost+found 디렉토리를 만들 수 없을 경우 SORRY. CANNOT CREATE lost+found DIRECTORY 메시지를 표시하고 손실된 inode 연결 시도를 중지합니다. 이 오류는 4단계의 뒷부분에서 UNREF 오류 메시지를 생성합니다. 손실된 inode 연결 시도를 중지하려면 *n*을 입력하십시오.

NO lost+found DIRECTORY (CREATE)

원인

파일 시스템의 루트 디렉토리에 lost+found 디렉토리가 없습니다. 다듬기 과정 중 fsck가 lost+found 디렉토리를 만들려고 시도합니다.

조치

파일 시스템의 루트에 lost+found 디렉토리를 만들려면 CREATE 프롬프트에서 *y*를 입력하십시오. lost+found 디렉토리를 만들 수 없을 경우 fsck가 SORRY. CANNOT CREATE lost+found DIRECTORY 메시지를 표시하고 손실된 inode 연결 시도를 중지합니다. 이 오류는 4단계의 뒷부분에서 UNREF 오류 메시지를 생성합니다. 손실된 inode 연결 시도를 중지하려면 *n*을 입력하십시오.

NO SPACE LEFT IN / lost+found (EXPAND)

원인

파일 시스템 루트 디렉토리의 lost+found 디렉토리에 다른 항목을 추가할 수 있는 공간이 없습니다. 다듬기 과정 중 fsck가 lost+found 디렉토리를 확장합니다.

조치

lost+found 디렉토리를 확장하여 새 항목에 대한 공간을 확보하려면 EXPAND 프롬프트에서 *y*를 입력하십시오. 확장 시도를 실패하면 fsck가 SORRY. NO SPACE IN lost+found DIRECTORY를 표시하고 lost+found 디렉토리에 대한 파일 연결 요청을 중지합니다. 이 오류는 4단계의 뒷부분에서 UNREF 오류 메시지를 생성합니다. lost+found 디렉토리에서 불필요한 항목을 삭제하십시오. 이 오류가 발생할 경우 -o p 옵션을 통해 다듬기가 적용될 때 fsck가 종료됩니다. 손실된 inode 연결 시도를 중지하려면 *n*을 입력하십시오.

UNREF FILE I=*inode-number* OWNER=*UID* MODE=*file-mode* SIZE=*file-size*
MTIME=*modification-time* (RECONNECT)

원인

파일 시스템이 통과될 때 *inode-number* 파일 inode가 디렉토리 항목에 연결되지 않았습니니다. *inode-number* inode의 *UID* 소유자, *file-mode* 모드, *file-size* 크기 및 *modification-time* 수정 시간이 표시됩니다. fsck의 다듬기 과정 중 해당 크기 또는 해당 링크 수가 0일 경우 파일이 지워지고, 그렇지 않을 경우 파일이 다시 연결됩니다.

조치

inode-number inode를 파일 시스템의 lost+found 디렉토리에 다시 연결하려면 *y*를 입력하십시오. *inode-number* inode를 lost+found 디렉토리에 연결할 때 문제가 발생할

경우 이 오류로 인해 4단계에서 `lost+found` 오류 메시지가 생성될 수 있습니다. 이 오류 상태를 무시하려면 `n`을 입력하십시오. 이 오류는 항상 4단계에서 `CLEAR` 오류 상태를 호출합니다.

```
UNREF type I=inode-number OWNER=UID MODE=file-mode SIZE=file-size
MTIME=modification-time (CLEAR)
```

원인

파일 시스템이 통과될 때 `type`이 디렉토리 또는 파일인 `inode-number inode`가 디렉토리 항목에 연결되지 않았습니다. `inode-number inode`의 `UID` 소유자, `file-mode` 모드, `file-size` 크기 및 `modification-time` 수정 시간이 표시됩니다. `fsck`의 다듬기 과정 중 해당 크기 또는 해당 링크 수가 0일 경우 파일이 지워지고, 그렇지 않을 경우 파일이 다시 연결됩니다.

조치

해당 콘텐츠를 0으로 설정하여 `inode-number inode`의 할당을 해제하려면 `CLEAR` 프롬프트에서 `y`를 입력하십시오. 이 오류 상태를 무시하려면 `n`을 입력하십시오.

```
ZERO LENGTH DIRECTORY I=inode-number OWNER=UID MODE=file-mode
SIZE=file-size MTIME=modification-time (CLEAR)
```

원인

`filename` 디렉토리 항목의 `file-size` 크기가 0입니다. `UID` 소유자, `file-mode` 모드, `file-size` 크기, `modification-time` 수정 시간 및 `filename` 디렉토리 이름이 표시됩니다.

조치

해당 콘텐츠를 0으로 설정하여 `inode-number` 디렉토리 `inode`의 할당을 해제하려면 `y`를 입력하십시오. 이 오류 상태를 무시하려면 `n`을 입력하십시오.

5단계: 실린더 그룹 메시지 확인

이 단계에서는 현재 Oracle Solaris 릴리스의 5단계 `fsck` 메시지를 다룹니다.

이 단계에서는 사용 가능한 조각 및 사용된 `inode` 맵을 확인합니다. 또한 다음으로 인한 오류 상태를 보고합니다.

- 사용된 `inode` 맵에서 할당된 `inode` 누락
- 사용 가능한 조각 맵에서 사용 가능한 조각 누락
- 사용된 `inode` 맵에 사용 가능한 `inode` 있음
- 잘못된 총 사용 가능한 조각 수
- 잘못된 총 사용된 `inode` 수

이러한 메시지(영문자순)는 5단계에서 발생할 수 있습니다.

```
FRAG BITMAP WRONG (CORRECTED)
```

원인

실린더 그룹 조각 맵에 일부 사용 가능한 조각이 누락되었습니다. 다듬기 과정 중 `fsck`가 맵을 재구성합니다.

조치

사용 가능한 조각 맵을 재구성하려면 **SALVAGE** 프롬프트에서 **y**를 입력하십시오. 이 오류 상태를 무시하려면 **n**을 입력하십시오.

CG *cg-number*: BAD MAGIC NUMBER

원인

cg-number 실린더 그룹의 매직 넘버가 잘못되었습니다. 일반적으로 이 오류는 실린더 그룹 맵이 삭제되었음을 나타냅니다. 대화형으로 실행할 때 실린더 그룹의 재구성이 필요한 것으로 표시됩니다. 파일 시스템을 다듬는 중인 경우 **fsck**가 종료됩니다.

조치

이 오류가 발생할 경우 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

CORRECT GLOBAL SUMMARY (SALVAGE)

원인

요약 정보가 잘못되었습니다. 다듬기 과정 중 **fsck**가 요약 정보를 다시 계산합니다.

조치

요약 정보를 재구성하려면 **SALVAGE** 프롬프트에서 **y**를 입력하십시오. 이 오류 상태를 무시하려면 **n**을 입력하십시오.

5단계: 실린더 그룹 메시지 확인

이 단원에서는 Solaris 10 초기 3/05 릴리스의 5단계 **fsck** 메시지를 다룹니다.

이 단계에서는 사용 가능한 블록 및 사용된 **inode** 맵을 확인합니다. 또한 다음으로 인한 오류 상태를 보고합니다.

- 사용된 **inode** 맵에서 할당된 **inode** 누락
- 사용 가능한 블록 맵에 사용 가능한 블록 누락
- 사용된 **inode** 맵에 사용 가능한 **inode** 있음
- 잘못된 총 사용 가능한 블록 수
- 잘못된 총 사용된 **inode** 수

이러한 메시지(영문자순)는 5단계에서 발생할 수 있습니다.

BLK(S) MISSING IN BIT MAPS (SALVAGE)

원인

실린더 그룹 블록 맵에 일부 사용 가능한 블록이 누락되었습니다. 다듬기 과정 중 **fsck**가 맵을 재구성합니다.

조치

사용 가능한 블록 맵을 재구성하려면 **SALVAGE** 프롬프트에서 **y**를 입력하십시오. 이 오류 상태를 무시하려면 **n**을 입력하십시오.

CG *character-for-command-option*: BAD MAGIC NUMBER

원인

character-for-command-option 실린더 그룹의 매직 넘버가 잘못되었습니다. 일반적으로 이 오류는 실린더 그룹 맵이 삭제되었음을 나타냅니다. 대화형으로 실행할 때 실린더 그룹의 재구성이 필요한 것으로 표시됩니다. 파일 시스템을 다듬는 중인 경우 fsck가 종료됩니다.

조치

이 오류가 발생할 경우 현지 서비스 공급자 또는 다른 적격자에게 문의하십시오.

FREE BLK COUNT(S) WRONG IN SUPERBLK (SALVAGE)

원인

실제 사용 가능한 블록 수가 파일 시스템 수퍼블록의 사용 가능한 블록 수와 다릅니다. *-op* 옵션이 지정된 경우 수퍼블록의 사용 가능한 블록 수가 자동으로 수정됩니다.

조치

수퍼블록의 사용 가능한 블록 정보를 재구성하려면 SALVAGE 프롬프트에서 *y*를 입력하십시오. 이 오류 상태를 무시하려면 *n*을 입력하십시오.

SUMMARY INFORMATION BAD (SALVAGE)

원인

요약 정보가 잘못되었습니다. 다듬기 과정 중 fsck가 요약 정보를 다시 계산합니다.

조치

요약 정보를 재구성하려면 SALVAGE 프롬프트에서 *y*를 입력하십시오. 이 오류 상태를 무시하려면 *n*을 입력하십시오.

fsck 요약 메시지

이 단원에서는 현재 Oracle Solaris 릴리스의 fsck 요약 메시지를 다룹니다. Solaris 10 6/06 릴리스 이상을 실행 중이 아닌 경우 정리 단계에서 해당 메시지가 표시됩니다. 자세한 내용은 284 페이지 “정리 단계 메시지”를 참조하십시오.

파일 시스템이 확인되면 몇 가지 요약 메시지가 표시됩니다.

number-of files, number-of-files
used, number-of-files free (number-of frags, number-of blocks,
percent fragmentation)

이 메시지는 확인된 파일 시스템의 파일 중 *number-of*개가 조각 크기의 블록 *number-of*개를 사용 중이며 파일 시스템에서 사용 가능한 조각 크기의 블록이 *number-of*개임을 나타냅니다. 괄호 안의 숫자는 사용 가능한 수를 사용 가능한 조각 *number-of*개, 사용 가능한 전체 크기의 블록 *number-of*개 및 조각 *percent*로 분석한 것입니다.

***** FILE SYSTEM WAS MODIFIED *****

이 메시지는 fsck가 파일 시스템을 수정했음을 나타냅니다. 이 메시지가 표시되면 fsck를 다시 실행할 필요가 없습니다. 이 메시지는 fsck의 해결 작업을 나타내는 정보 메시지입니다.

정리 단계 메시지

이 단원에서는 Oracle Solaris 10 릴리스의 fsck 정리 단계 메시지를 다룹니다. 이 릴리스의 메시지는 fsck 요약 단계의 메시지와 유사할 수 있습니다. 자세한 내용은 [283 페이지](#) “[fsck 요약 메시지](#)”를 참조하십시오.

파일 시스템이 확인되면 몇 가지 정리 기능이 수행됩니다. 정리 단계에서는 다음과 같은 상태 메시지가 표시됩니다.

```
number-of files, number-of-files
used, number-of-files free (number-of frags, number-of blocks,
percent fragmentation)
```

이 메시지는 확인된 파일 시스템의 파일 중 *number-of*개가 조각 크기의 블록 *number-of*개를 사용 중이며 파일 시스템에서 사용 가능한 조각 크기의 블록이 *number-of*개임을 나타냅니다. 괄호 안의 숫자는 사용 가능한 수를 사용 가능한 조각 *number-of*개, 사용 가능한 전체 크기의 블록 *number-of*개 및 조각 *percent*로 분석한 것입니다.

```
***** FILE SYSTEM WAS MODIFIED *****
```

이 메시지는 fsck가 파일 시스템을 수정했음을 나타냅니다. 이 파일 시스템이 마운트되었거나 현재 루트(/) 파일 시스템인 경우 재부트하십시오. 파일 시스템이 마운트된 경우 파일 시스템의 마운트를 해제하고 fsck를 다시 실행해야 합니다. 그렇지 않을 경우 fsck를 통해 완료된 작업의 실행이 핵심 테이블 복사를 통해 취소될 수 있습니다.

```
filename FILE SYSTEM STATE SET TO OKAY
```

이 메시지는 파일 시스템의 *filename* 파일이 안정적인 것으로 표시되었음을 나타냅니다. fsck -m 명령을 사용하여 파일 시스템을 확인해야 할지 여부를 결정하십시오.

```
filename FILE SYSTEM STATE NOT SET TO OKAY
```

이 메시지는 파일 시스템의 *filename* 파일이 안정적인 것으로 표시되지 않았음을 나타냅니다. fsck -m 명령을 사용하여 파일 시스템을 확인해야 할지 여부를 결정하십시오.

소프트웨어 패키지 문제 해결(작업)

이 장에서는 소프트웨어 패키지 설치 또는 제거 시 발생할 수 있는 문제에 대해 설명합니다. 특정 소프트웨어 패키지 설치 오류 단원에서는 발생할 수 있는 패키지 설치 및 관리 오류에 대해 설명하며, 일반 소프트웨어 패키지 설치 문제 단원에서는 오류 메시지에 표시되지 않을 수 있는 동작 문제에 대해 설명합니다.

다음은 이 장에서 다루는 정보를 나열한 것입니다.

- 286 페이지 “특정 소프트웨어 패키지 설치 오류”
- 286 페이지 “일반 소프트웨어 패키지 설치 문제”

소프트웨어 패키지에 대한 자세한 내용은 **시스템 관리 설명서: 기본 관리의 20 장, “소프트웨어 관리(개요)”**를 참조하십시오.

소프트웨어 패키지 심볼릭 링크 문제 해결

이전 Solaris 릴리스에서는 소프트웨어 패키지를 만들 때 `pkgmap` 파일에서 심볼릭 링크 대상을 지정할 수 있는 방법이 없었습니다. 따라서 `pkgadd` 명령으로 패키지를 추가할 때 패키지 또는 패치 관련 심볼릭 링크가 항상 심볼릭 링크 대상이 아닌 심볼릭 링크 소스로 연결되었습니다. 이로 인해 심볼릭 링크 대상을 다른 항목으로 변경해야 할 패키지 또는 패치 패키지를 업그레이드할 때 문제가 발생했습니다.

이제 패키지가 심볼릭 링크 대상을 다른 항목으로 변경해야 할 경우 `pkgadd` 명령에서 심볼릭 링크 소스가 아닌 심볼릭 링크 대상을 검사하는 것으로 기본 동작이 바뀌었습니다.

이에 따라 일부 패키지가 새 `pkgadd` 동작을 준수할 수도 있고 준수하지 않을 수도 있는 문제가 발생합니다.

`PKG_NONABI_SYMLINKS` 환경 변수를 사용하면 이전 심볼릭 링크 동작과 새 `pkgadd` 심볼릭 링크 동작 간의 전환을 수행할 수 있습니다. 이 환경 변수를 `true`로 설정하면 `pkgadd`가 심볼릭 링크 소스에 연결됩니다.

이 변수를 설정하면 `pkgadd` 명령으로 패키지를 추가하기 전에 관리자가 설정한 경우 비준수 패키지를 이전 동작으로 되돌릴 수 있습니다.

`pkgadd` 명령으로 추가된 경우 새 `pkgadd` 심볼릭 링크 동작으로 인해 기존 패키지가 실패할 수 있습니다. 이 경우 다음과 같은 오류 메시지가 표시될 수 있습니다.

```
unable to create symbolic link to <path>
```

이 문제로 인해 패키지가 설치되지 않으면 다음 작업을 수행하십시오.

1. Oracle에서 제공한 패키지일 경우 [My Oracle Support](#)를 통해 비준수 패키지 이름을 보고하십시오.
2. `PKG_NONABI_SYMLINKS` 환경 변수를 설정하고 `pkgadd` 명령으로 패키지를 다시 추가해 봅니다.

```
# PKG_NONABI_SYMLINKS=true
# export PKG_NONABI_SYMLINKS
# pkgadd pkg-name
```

특정 소프트웨어 패키지 설치 오류

WARNING: filename <not present on Read Only file system>

오류 발생 원인	문제 해결 방법
이 오류 메시지는 일부 패키지 파일을 설치할 수 없음을 나타냅니다. 일반적으로 <code>pkgadd</code> 를 사용하여 클라이언트에 패키지를 설치하는 경우 이 오류가 발생합니다. 이 경우 <code>pkgadd</code> 가 서버에서 마운트된 파일 시스템에 패키지를 설치하려고 시도하지만 <code>pkgadd</code> 에 권한이 없어 설치를 실패하는 것입니다.	패키지 설치 중 이 경고 메시지가 표시되면 서버에도 패키지를 설치해야 합니다. 자세한 내용은 시스템 관리 설명서: 기본 관리의 20 장 , “소프트웨어 관리(개요)”를 참조하십시오.

일반 소프트웨어 패키지 설치 문제

오류 발생 원인	문제 해결 방법
Solaris 2.5 릴리스 및 호환되는 버전 이전에 개발된 일부 패키지를 추가하거나 제거하는 것과 관련하여 알려진 문제가 있습니다. 이러한 패키지를 추가하거나 제거할 때 사용자 상호 작용 중 설치를 실패하거나 사용자 상호 작용을 위한 프롬프트가 표시되고 응답이 무시되는 경우가 있습니다.	다음과 같은 환경 변수를 설정한 후 패키지를 다시 추가해 보십시오. <code>NONABI_SCRIPTS=TRUE</code>

색인

A

acct.h 형식 파일, 146, 147
acctcms 명령, 140, 150
acctcom 명령, 146, 147
acctcon 명령, 134, 140, 149
acctdusg 명령, 128, 144, 148
acctprc 명령, 140
acctwtmp 명령, 127, 129, 142
active.MMDD 파일, 136, 149
active 파일, 136, 149
at.deny 파일, 119, 122
 설명, 106
at 명령, 118, 119, 122
 -l 옵션(목록), 121
 -m 옵션(배일), 119, 120
 개요, 106, 107, 118
 액세스 거부, 122
 액세스 제어, 119, 122
 개요, 106
 오류 메시지, 123
 자동 예약, 109
at 작업 파일, 118, 121
 만들기, 119, 120
 삭제, 121
 설명, 107
 위치, 107
 제출, 118
 표시, 121
atjobs 디렉토리, 109
 설명, 106

C

chargefee 스크립트, 128, 129, 144
 사용자 요금 청구, 133
ckpacct 스크립트, 129, 131, 132
closewtmp 명령, 140
cmsprev 파일, 150
Command not found 오류 메시지, 253
consadm 명령, 222-223
 보조 콘솔 목록 표시(방법), 223
 보조 콘솔을 사용 안함으로 설정, 224
 보조 콘솔을 사용으로 설정, 222-223
 시스템 재부트 시, 223
coreadm 명령, 226
 코어 덤프 구성 표시, 228
 코어 파일 관리, 226
 코어 파일 이름 패턴 설정, 229
CPU(중앙 처리 장치)
 고사용 프로세스, 177
 정보 표시
 시간 사용, 162, 177
 시간 사용량, 144
cron.allow 파일, 114, 115, 116
cron.deny 파일, 114, 115
 기본값, 115
cron 데몬, 107, 109
crontab 명령, 115
 cron 데몬, 109
 -e 옵션(편집), 110, 111
 -l 옵션(목록), 112, 113
 -r 옵션(제거), 113, 114
 /var/adm 유지 관리 및, 216
 개요, 106, 107

crontab 명령 (계속)

- 변경 사항 저장 없이 종료, 111
- 사용된 파일, 109
- 실행된 계산 스크립트, 131, 133
- 액세스 제어, 114, 115, 116
 - 개요, 106, 114, 115
 - 액세스 거부, 114, 115
 - 지정된 사용자로 액세스 제한, 116
 - 특정 사용자로 액세스 제한, 114, 115
- 예약, 109
- 오류 메시지, 117
- 일별 작업, 107

crontab 파일

- 구문, 109, 110
- 기본값, 108
- 만들기, 110, 111
- 만들기 및 편집, 105-106
- 삭제, 113, 114
- 설명, 109
- 액세스 거부, 115-116
- 위치, 108
- 제거, 113-114
- 편집, 110, 111
- 표시, 112, 113

ctacct.MMDD 파일, 140, 149

ctmp 파일, 149

D**date 명령**

- 계산 데이터, 127, 129

daytacct 파일

- runacct 스크립트, 140, 150
- /var/adm/acct/nite 디렉토리, 위치, 149
- 일별 사용량 보고서, 144

df 명령, 185, 186

- h 옵션, 77
- k 옵션(킬로바이트), 186
- t 옵션(총 블록 수), 78
- 개요, 76, 185
- 예, 77, 186

disktacct.MMDD 파일, 140

disktacct 파일, 129

disktacct 파일, 128, 140, 149

dispadmin 명령, 개요, 171

dmesg 명령, 216

dodisk 스크립트, 128

- 개요, 128, 129

- 생성된 파일, 128, 129, 140, 149

- 실행하는 crontab 항목, 133

- 주의, 128

dtmp 파일, 148

DTrace 기능, 210

du 명령, 82, 84

dumpadm, 시스템 충돌 정보 관리, 234

E**edquota 명령**

- p 옵션(프로토타입), 94

- t 옵션(시간 제한), 100

- 개별 사용자에게 대한 쿼터를 사용 안함으로 설정, 102

- 개요, 90, 91, 99

- 사용자 쿼터 설정, 94

eeeprom 명령, ttymon 터미널에 대한 변조 속도를 설정하기 위해 사용, 41

/etc/acct/holidays 파일, 132, 133

/etc/cron.d/at.deny 파일, 119, 122

/etc/cron.d/cron.allow 파일, 114, 115, 116

/etc/cron.d/cron.deny 파일, 114, 115

/etc/init.d/acct 파일, 132

/etc/syslog.conf 파일, 218

/etc/utmpx 파일, 37

/etc/vfstab 파일, 92

F

fcntl 정보, 163, 164, 166

fd2log 파일, 136, 139, 149

fee 파일, 129, 134, 140, 148

find 명령

- 오래된/비활성 파일, 84, 85

- 코어 파일, 87

- 크기 제한을 초과하는 파일 찾기, 82

fiscrptn 파일, 150

fsck 명령, 107

fstat 정보, 163, 164, 166

G

getty, 23

GRUB 기반 부트

SMF 부트 아카이브 서비스 실패 문제 해결, 210

시스템 충돌

실패한 SMF 부트 아카이브 서비스, 233

H

holidays 파일, 133

hostid 명령, 60

I

iostat 명령

개요, 183

기본 정보 표시, 184

K

klwp 구조, 156

kldb 유틸리티, 243-244, 244-245

kthread 구조, 156

L

lastdate 파일, 139, 149

lastlogin 명령, 140

lineuse 파일, “/var/adm/acct/nite/lineuse
파일”참조

localeadm 명령, 56

lock 파일, 136, 139

lock1 파일, 139

log.MMDD 파일, 149

log 파일, 149

loginlog 파일, 140, 150

ls 명령

디렉토리 크기 확인, 79

-l 옵션(크기 단위: 바이트), 80

-s 옵션(크기 단위: 블록 수), 80

-t 옵션(최신 파일), 85

LWP(Lightweight Process)

구조, 156

정의, 155

프로세스 및, 155, 156

LWP(경량 프로세스), 정보 표시, 163

M

mdb 유틸리티, 238, 239

messages.n 파일, 216

messages 파일, 212, 218

monacct 스크립트

runacct 스크립트, 130, 139

사용/생성된 파일, 150

실행 일정 잡기, 131

실행하는 crontab 항목, 132

월별 명령 요약, 144, 145

MOTD(오늘의 메시지) 기능, 70-71

motd 파일, 70-71

motd 파일, 70

N

nice 명령, 175, 177

nice 번호, 162, 175

nlsadmin 명령, 39-40

O

Oracle Solaris 프로세스 계산 및 통계 개선, 125-126

owtmpx 파일, 150

P

pacctn 파일

개요, 129, 140, 148

pacctn 파일 (계속)
 모니터링 크기, 129, 139
 표시, 146, 147

perf 파일, 205

pfiles 명령, 163, 164, 166

pflags 명령, 163, 164

pkill 명령, 164, 167

pldd 명령, 163, 164

pmadm 명령

ttymon 서비스 나열, 47

ttymon 서비스 추가, 46

ttymon 서비스를 사용 안함, 49

설명, 36

pmap 명령, 163, 164

prdaily 스크립트

runacct 스크립트, 139, 150

개요, 139

사용된 파일, 149, 150

회선 사용량 보고, 150

priocntl 명령

-c 옵션(예약 클래스 지정), 174

-i 옵션(ID 유형), 174

-l 옵션(예약 클래스 표시), 172

-m 옵션(최대/최소 우선 순위), 174

-p 옵션(우선 순위 지정), 174

-s 옵션(우선 순위 상한/우선 순위 변경), 174

개요, 171

/proc 디렉토리, 163

proc 구조, 156, 162

proc 도구, 코어 파일 검사, 230

PROCFS(프로세스 파일 시스템), 163

prtconf 명령, 60, 65

시스템의 제품 이름 표시, 65

ps 명령, 162, 165

-c 옵션(예약 클래스), 162, 177

-ecl 옵션(전역 우선 순위), 172

-ef 옵션(전체 정보), 164, 165

개요, 162

보고되는 필드, 162

psig 명령, 163, 164

pstack 명령, 163, 164

ptacctn.MMDD 파일, 141

ptime 명령, 164

ptree 명령, 163, 164, 166

pwait 명령, 164

pwdx 명령, 163, 164, 166

Q

quot 명령, 84

quotacheck 명령, 90, 95

quotaon 명령, 91, 96

quotas 파일, 90, 92

R

reboots 파일, 140, 149

repquota 명령, 97, 98-99

rm 명령, 86, 87

root crontab 파일, 128

rprrt.MMDD 파일, 130, 150

rpt.MMDD 파일, 140, 150

RS-232-C, “직렬 포트”참조

runacct 스크립트, 135, 139

monacct 스크립트, 139

prdaily 스크립트, 139, 150

개요, 130

다시 시작, 136, 139, 141

마지막 실행 시간, 149

사용/생성된 파일, 148, 150

사용자 요금 계산, 133, 144

상태, 140

실패, 136

실행 일정 잡기, 131

실행하는 crontab 항목, 139

오류 메시지, 136

오류 보호, 139

진단 파일, 139

진행 파일, 139

훼손된 파일 수정, 134, 135, 140

runaway 프로세스, 177

S

sa1 명령, 204

sa2 명령, 204, 206

- SAC, “Service Access Controller” 참조
sacadm 명령, 44
 ttymon 포트 모니터 강제 종료, 43
 ttymon 포트 모니터 시작, 44
 ttymon 포트 모니터 추가, 42
 설명, 35
sadc 명령, 204, 205, 206
sadd 파일, 205
SAF, “Service Access Facility” 참조
SAF를 사용하여 직렬 포트 관리, 작업 맵, 34
sar 명령, 188, 206
 -A 옵션(전체 성능), 204, 206
 -a 옵션(파일 액세스), 188, 189
 -b 옵션(버퍼), 189
 -c 옵션(시스템 호출), 191
 -e 옵션(종료 시간), 206
 -f 옵션(데이터를 추출할 파일), 206
 -i 옵션(간격), 206
 -m 옵션(프로세스간 통신), 196
 -p 옵션(페이지 인/페이지 폴트), 197
 -q 옵션(대기열), 198, 199
 -r 옵션(사용되지 않은 메모리), 199
 -s 옵션(시작 시간), 206
 -u 옵션(CPU 사용), 200
 -v 옵션(시스템 테이블), 201
 -y 옵션(터미널 장치), 203
 개요, 188, 206
 나열된 옵션, 206
 모든 옵션에 대한 설명, 206
Service Access Controller, 35, 36
Service Access Facility
 개요, 24, 34
 사용, 24, 34
 서비스 제어
 상태(표), 52
 연관된 프로그램(표), 35
shutacct 스크립트, 129, 130
shutdown 명령, 130
Spacctn.MMDD 파일, 140, 148
startup 명령, acct, 129
statefile 파일, 136, 140, 149
svcadm enable system/sar:default 명령, 205
sys crontab, 205
syslog.conf 파일, 218
syslogd 데몬, 215
- T**
tacct.MMDD 파일, 135-136, 140, 150
tacct 파일, 135-136, 140, 150
tacctn 파일, 150
tacctprev 파일, 150
/tmp/disktacct.MMDD 파일, 140
tmpwtmp 파일, 140, 149, 150
tty 회선, 사용량 모니터링, 142
tty 회선
 사용량 모니터링, 127, 143, 150
 잘못된 회선 문제 해결, 143
ttyadm 명령, 39
ttymon 서비스
 나열, 47
 사용, 49
 사용 안함, 49
 추가, 46
ttymon 콘솔 터미널에 대한 변조 속도 설정,
 방법, 41-42
ttymon 포트 모니터, 44
 (그림), 37
 강제 종료, 43
 시작, 44
 양방향 모뎀 서비스, 37-38
 추가, 42
turnacct switch 스크립트, 129
turnacct switch 스크립트, 140
- U**
UFS quota 명령, 91
UFS 쿼터, 98-99, 99-100
 개별 사용자에게 대해 변경, 101
 개별 사용자에게 대해 사용 안함으로 설정, 102
 명령, 97
 소프트 제한 시간
 변경, 100
 요구 사항, 91
 확인, 97
UFS 쿼터 사용, 89-90

UFS 쿼터 초기화, 91
 UFS 파일 시스템, 정보 표시, 84
 UNIX 시스템(충돌 정보), 233
 user 구조, 156
 /usr/adm/messages 파일, 212
 /usr/bin/mdb 유틸리티, 238
 /usr/proc/bin 디렉토리, 163, 164
 utmp2wtmp 명령, 140

V

/var/adm/acct 디렉토리, 149
 /var/adm/acct/fiscal 디렉토리, 149
 /var/adm/acct/nite/active 파일, 136, 139, 149
 /var/adm/acct/nite/active.MMDD 파일, 139, 149
 /var/adm/acct/nite/cms 파일, 140
 /var/adm/acct/nite/cms 파일, 149
 /var/adm/acct/nite/ctacct.MMDD 파일, 140, 149
 /var/adm/acct/nite/ctmp 파일, 149
 /var/adm/acct/nite/daycms 파일, 140, 149, 150
 /var/adm/acct/nite/daytacct 파일, “daytacct
 파일”참조
 /var/adm/acct/nite 디렉토리, 149
 /var/adm/acct/nite/disktacct 파일, 129
 /var/adm/acct/nite/disktacct 파일, 128, 130, 140,
 149
 /var/adm/acct/nite/disktacct.MMDD 파일, 140
 /var/adm/acct/nite/fd2log 파일, 136, 139, 149
 /var/adm/acct/nite/lastdate 파일, 139, 149
 /var/adm/acct/nite/lineuse 파일, 140, 149, 150
 /var/adm/acct/nite/lock 파일, 136, 139, 149
 /var/adm/acct/nite/lock1 파일, 139
 /var/adm/acct/nite/log 파일, 149
 /var/adm/acct/nite/log.MMDD 파일, 149
 /var/adm/acct/nite/owtmpx 파일, 150
 /var/adm/acct/nite/reboots 파일, 140, 149
 /var/adm/acct/nite/statefile 파일, 136, 140, 149
 /var/adm/acct/nite/tmpwtmp 파일, 140, 149, 150
 /var/adm/acct/nite/wtmp.MMDD 파일, 140, 149
 /var/adm/acct/nite/wtmperror 파일, 149
 /var/adm/acct/nite/wtmperror.MMDD 파일, 149
 /var/adm/acct/sum/cms 파일, 140
 /var/adm/acct/sum/cms 파일, 150
 /var/adm/acct/sum/cmsprev 파일, 150

/var/adm/acct/sum/daycms 파일, 150
 /var/adm/acct/sum/daycms 파일, 140
 /var/adm/acct/sum 디렉토리, 130, 149, 150
 /var/adm/acct/sum/loginlog 파일, 140, 150
 /var/adm/acct/sum/rprt.MMDD 파일, 150
 /var/adm/acct/sum/rprtMMDD 파일, 130
 /var/adm/acct/sum/rprt.MMDD 파일, 140
 /var/adm/acct/sum/tacct 파일, 140
 /var/adm/acct/sum/tacct 파일, 135-136, 150
 /var/adm/acct/sum/tacct.MMDD 파일, 140, 150
 /var/adm/acct/sum/tacctMMDD 파일, 135-136
 /var/adm/acct/sum/tacctprev 파일, 136, 150
 /var/adm/dtmp 파일, 148
 /var/adm/fee 파일, 129, 134, 140, 148
 /var/adm/messages 파일, 212, 218
 /var/adm/messages.n 파일, 216
 /var/adm/sa/sadd 파일, 205
 /var/adm/Spacctn.MMDD 파일, 140, 148
 /var/adm 디렉토리
 설명, 148
 크기 제어, 86
 원시 계산 데이터, 129
 /var/spool/cron/atjobs 디렉토리, 106, 107, 109
 /var/spool/cron/crontabs 디렉토리, 108, 109
 /var/spool/cron/crontabs/root 파일, 108, 128
 /var/spool/cron/crontabs/sys crontab, 205
 vfstab 파일, 쿼터 및, 92
 vmstat 명령
 개요, 180
 보고서의 필드, 180

W

Watchdog reset! 메시지, 215
 wtmp.MMDD 파일, 140, 149
 wtmperror.MMDD 파일, 149
 wtmperror 파일, 149
 wtmpfix 명령, 134, 140, 149
 wtmpx 파일, 135
 개요, 129, 134, 140
 일별 보고서, 142
 종료, 130
 훼손 수정, 134, 135, 140

X

x86 기반 시스템 재부트, 부트 아카이브 SMF 서비스 실패, 210

Z

ZFS, 스왑 영역 및 덤프 볼륨 요구 사항, 233

강

강제로 프로그램 종료, 246

검

검색 경로, 설정할 파일, 254

계

계산, 135, 137, 150

“사용자 요금 청구”참조

개요, 126

디스크, 128, 129

acctdusg 프로그램, 144

보고서, 141

개요, 141

마지막 로그인 보고서, 146

일별 명령 요약, 144, 150

일별 보고서(tty 회선 사용률), 142, 143

일별 사용량 보고서, 143, 144

총 명령 요약(월별), 145, 150

사용 안함, 138

사용자 요금 계산, 128

“사용자 요금 청구”참조

시작, 132

연결, 127

runacct 상태, 140

/var/adm/acct/nite/directory, 149

/var/adm/wtmpx, 143

원시 데이터, 129

유지 관리, 137

유형, 133

계산 (계속)

일별, 129, 150

“계산, 보고서”참조

단계별 요약, 130

자동으로 실행하도록 설정(방법), 132

중지, 137

파일, 148, 150

프로세스, 128, 129, 143, 144

훼손된 파일 수정

tacct 파일, 135-136

wtmpx 파일, 134, 135, 140

고

고객 서비스, 충돌 정보 전송, 212

공

공유 메모리, 프로세스 가상 메모리, 156

공통 에이전트 컨테이너

Oracle Solaris OS에서 문제 해결, 209-210

문제 해결, 249-251

공통 에이전트 컨테이너 공유, 공유 구성

요소, 249-251

공통 에이전트 컨테이너 공유 구성 요소

문제 유형

수퍼유저 암호 보안, 249-251

포트 번호 충돌, 249-251

포트 번호 지정(확인 방법), 250

기

기본값

nice 번호, 175

소프트 제한 시간, 100

오늘의 메시지, 70

쿼터, 99-100

기술 지원

충돌 덤프 분석, 234

충돌 정보 전송, 212

나

나열

- 실행 중인 프로세스, 165
- 파일 및 디렉토리, 79, 80, 85
- 프로세스, 164

네

- 네트워크, 액세스 문제 인식, 256
- 네트워크 액세스 문제 인식, 256

다

다시 시작

- runacct 스크립트, 136, 139, 141
- 프로세스, 163
- 다이얼 아웃 모뎀 서비스, 21
- 다이얼 인 모뎀 서비스, 21

단

단말기

- 잘못된 회선 문제 해결, 143
- 회선 사용량
 - /var/adm/acct/nite/lineuse 파일, 151
 - 연결 계산, 127
 - 일별 보고서, 142, 143

덤프

- 덤프 볼륨 크기, ZFS 루트 파일 시스템이 설치된 시스템의 경우, 233

도

- 도구, 시스템 성능 모니터링, 157

디

디렉토리

- 입시, 지우기, 84, 87
- 정보 표시, 79, 80, 82, 84
- 크기, 82, 84
- 프로세스에 대한 현재 작업 디렉토리, 163, 164
- 디스크 계산, “계산, 디스크” 참조
- 디스크 공간
 - 오래된/비활성 파일 찾기 및 삭제, 84, 88
- 정보 표시
 - df 명령, 185
 - 디렉토리 크기, 82, 84
 - 마운트 지점, 186
 - 사용자별 소유 디스크 공간, 84
 - 파일 크기, 79, 80, 82
 - 크기 제한을 초과하는 파일 찾기, 82
 - 큰 파일 찾기, 80, 81
- 디스크 공간의 사용자 소유권, 84
- 디스크 드라이브
 - 오래된/비활성 파일 찾기 및 삭제, 111
- 정보 표시
 - 사용 가능한 디스크 공간, 185
- 디스크 블록과 파일 제한, 차이, 90

로

- 로그 파일, 자동으로 삭제, 111
- 로그인 모니터링
 - 로그인 수, 144
 - 마지막 로그인, 140, 146, 150
 - 시간 사용량, 127, 129, 144

루

- 루트 암호, 잊음
 - SPARC, 243
 - x86, 243, 244
 - GRUB 기반 부트, 243-244
- 루트 암호 복구
 - SPARC, 243
 - x86, 243, 244
- 루트 암호 잊음
 - SPARC, 243

루트 암호 잊음 (계속)

x86, 243, 244

루틴 작업을 자동으로 실행(개요), 106

마

마지막 로그인 보고서, 146

만

만들기

at 작업, 119

at 작업, 120

crontab 파일, 110, 111

메

메모리

가상

프로세스, 156

공유

프로세스 가상 메모리, 156

정보 표시 명령, 60

정보 표시 예, 65

프로세스 구조 및, 156

명

명령, 모니터링 사용량, 149

모

모뎀, 29-30

다양한 사용 방법, 21

다이얼아웃 서비스, 21

다이얼인 서비스, 21

양방향 서비스, 21, 37-38

정의, 21

직렬 포트 도구 개요, 26

직렬 포트 도구 모뎀 템플릿, 27

문

문제 해결

tty 회선, 143

공통 에이전트 컨테이너, 209-210

공통 에이전트 컨테이너 공유 구성 요소

문제 유형, 249-251

소프트웨어 패키지 설치/제거, 285

프로세스, 177

문제 해결 작업, 검색 위치, 211

반

반복적 시스템 작업, 115

변

변경

crontab 파일, 110

개별 사용자에게 대한 UFS 쿼터, 101

날짜, 70

소프트제한 시간, 100

시스템의 호스트 이름, 71-72

예약 클래스, 174

오늘의 메시지, 70

우선 순위, 174, 175

시간 공유 프로세스, 175

변조 속도

eeprom 명령을 사용하여 설정하는 방법, 41

ttymon 터미널에 대해 설정하는 방법, 41-42

보

보안

at 명령, 119

crontab 명령, 115

보조(원격) 콘솔, 220

복

복원, 동일한 명령 사용, 249

부**부트**

- 생성된 메시지 표시, 216
- 실행 `sadc` 명령, 205
- 부트 아카이브, 재부트 시 SMF 서비스 실패, 210
- 부트 아카이브 서비스 실패
 - x86
 - GRUB 문제 해결, 245-246
- 부트 아카이브 서비스 실패 시 수행할 작업
 - x86
 - 비상 안전 조치를 한 아카이브 부트, 245-246

비

- 비상 메시지, 215

사

- 사용 안함, 시스템 계산, 138
- 사용 안함으로 설정
 - `consadm` 명령을 통한 보조 콘솔, 224
 - 개별 사용자에게 대한 쿼터, 102
- 사용으로 설정
 - `consadm` 명령을 통한 보조 콘솔, 222-223
 - 시스템 재부트 시 보조 콘솔, 223
- 사용자 로그인
 - 로그인 수, 144
 - 마지막 로그인 모니터링, 140, 146, 150
 - 시간 모니터링, 127, 140, 144
- 사용자 모드 우선 순위, 171
- 사용자 요금, 128, 129, 144
 - “사용자 요금 청구”참조
- 사용자 요금 청구, 133
 - “chargefee 스크립트”참조
- 사용자 정의
 - 시스템 메시지 로깅, 218
 - 시스템 메시지 로깅(방법), 219-220
- 사용자 쿼터, 97-98
 - 개별 사용자에게 대해 변경, 101
 - 개별 사용자에게 대해 사용 안함으로 설정, 102
- 설정, 94
- 사용자 프로세스
 - CPU 사용량, 144

사용자 프로세스(계속)

- 우선 순위, 171
- 우선 순위 변경, 175

삭**삭제**

- `at` 작업, 121
- `crontab` 파일, 113, 114
- 로그 파일, 111
- 오래된/비활성 파일, 107
- 오래된/비활성 파일 찾기 및 삭제, 84
- 임시 파일, 87
- 코어 파일, 87

상

- 상태, (runacct 스크립트), 140

새

- 새 기능, `svcadm enable system/sar:default` 명령, 205
- 새로운 기능
 - CPU 성능 카운터, 154
 - 향상된 `pfiles` 도구, 153

설

- 설정, `coreadm`으로 코어 파일 이름 패턴, 229

성**성능**

- 모니터링 도구, 157
- 보고, 188
- 시스템 작업 모니터링, 157, 188, 204
- 작업 데이터의 수동 수집, 188, 206
- 작업 데이터의 자동 수집, 204, 205
- 추적되는 작업, 157

성능 (계속)

- 파일 액세스, 188, 189
- 프로세스 관리, 155, 164, 175

소

- 소프트 제한 시간, 변경, 99-100
- 소프트웨어 패키지, 설치 문제 해결, 285

손

- 손상, 기타 시스템 정보 저장, 216

수

- 수정, 135
 - 훼손된 tacct 파일, 135-136
 - 훼손된 wtmpx 파일, 134, 135
- 수퍼유저(루트) 암호, 잊음
 - SPARC, 243
 - x86, 243, 244
- 수퍼유저 암호 보안
 - 공통 에이전트 컨테이너 공유 구성 요소 문제 해결, 249-251

스

- 스왑 볼륨 크기, ZFS 루트 파일 시스템이 설치된 시스템의 경우, 233

시

- 시간
 - CPU 사용, 162, 177
 - CPU 사용량, 144
 - 많은 양의 CPU 시간 누적 프로세스, 177
- 시간 공유 프로세스
 - 예약 매개 변수 변경, 174
 - 우선 순위
 - 개요, 171

시간 공유 프로세스, 우선 순위 (계속)

- 범위, 171
- 변경, 174, 175
- 시스템 계산, 작업 맵, 130-131
- 시스템 리소스
 - 개요, 155
 - 계산
 - 개요, 126
 - 모니터
 - UFS 쿼터, 98-99
 - 모니터링, 119
 - 계산, 137
 - 계산 시스템, 150
 - 자동, 119
 - 충돌, 218, 247
- 시스템 리소스 관리, 로드맵, 55
- 시스템 메시지
 - 로깅 사용자 정의(방법), 219-220
 - 저장소 위치 지정, 215
- 시스템 메시지 로깅(사용자 정의), 218
- 시스템 작업
 - “crontab 명령, at 명령”참조
 - 데이터의 수동 수집, 206
 - 데이터의 자동 수집, 204, 205
 - 예약
 - 반복적 작업, 107, 108
 - 일회성 작업, 107, 118
 - 자동으로 예약, 106
 - 추적된 작업 목록, 157
- 시스템 작업 실행 자동화, 106
- 시스템 충돌 문제 해결
 - GRUB
 - 재부트 시 부트 아카이브 서비스 실패, 233
 - 시스템 충돌 정보, dumpadm을 사용하여 관리, 234
 - 시스템 충돌 정보 관리, dumpadm 사용, 234
 - 시스템의 물리적 프로세서 유형 표시, psrinfo
 - p, 66
 - 시스템의 제품 이름, prtconf 명령으로 표시, 65

실

- 실시간 프로세스, 클래스 변경, 174
- 실패한 SMF 부트 아카이브 서비스, GRUB 기반 부트 문제 해결, 233

실패한 SMF 부트 아카이브 서비스 문제 해결
x86
GRUB 비상 안전 조치를 한 아카이브, 245-246
실패한 x86 기반 시스템 재부트, SMF 부트 아카이브
서비스, 210

알

알림 메시지 우선 순위(syslogd의 경우), 219

암

암호 보안 충돌, 슈퍼유저, 공통 에이전트
컨테이너, 249-251

양

양방향 모뎀 서비스, 21, 37-38

어

어댑터 보드(직렬 포트), 22

연

연결 계산, “계산, 연결”참조

영

영숫자 터미널, “터미널”참조

예

예약
“crontab 명령, at 명령”참조
반복적 시스템 작업, 107, 108
일회성 시스템 작업, 107, 118
예약 클래스, 171

예약 클래스(계속)

변경, 174
우선 순위 레벨, 171
우선 순위 레벨 및, 174
우선 순위 변경, 174, 175
정보 표시, 162, 172
지정, 173

오

오류 메시지
at 명령, 123
crontab 명령, 117
runacct 스크립트, 136
로그 파일, 212, 215
로깅 사용자 정의, 218
소스, 218
우선 순위, 219
저장소 위치 지정, 215, 218
충돌 관련, 215
충돌 메시지, 216

요

요구 사항, UFS 쿼터, 91
요금, 사용자, 129, 133
요금(사용자), 144

우

우선 순위(프로세스)
개요, 171, 175
변경, 174, 175
시간 공유 프로세스, 174, 175
사용자 모드 우선 순위, 171
예약 클래스 및, 174
전역
정의, 171
표시, 172
정보 표시, 162, 172
지정, 173, 174

원

원격 인쇄, 사용자 요금 계산, 133

월

월별 명령 요약, 145

월별 작업(crontab으로 예약), 107

응

응용 프로그램 스택드, 155, 156

이

이유 레코드, 프로세스 계산, 130

인

인쇄, 사용자 요금 계산, 133

일

일별 계산, “계산, 일별”참조

일별 작업(crontab으로 예약), 107

임

임시 디렉토리, 84, 87

자

자동 시스템 작업 데이터 수집, 204, 205

자동 시스템 작업 보고, 204, 206

자동 시스템 작업 실행

 단일 작업, 118, 119, 122

 반복적 작업, 115, 116

자동으로 쿼터 켜기, 90

재

재부트

 /var/adm/wtmpx 파일, 129

 연결 계산, 127

 일별 보고서, 142

 충돌 후 실패, 241-242

전

전역 우선 순위

 정의, 171

 표시, 172

전역 코어 파일 경로, coreadm으로 설정, 226

전원 순환, 247

전원 장애 복구, 142

전체 충돌 덤프 디렉토리에서 복구, 239

제

제거, crontab 파일, 113-114

제어

 at 명령에 대한 액세스, 106

 at 명령에 액세스, 119, 122

 crontab 명령에 대한 액세스

 개요, 106

 crontab 명령에 액세스, 115, 116

 프로세스, 166-167

제품 이름 정보 표시, prtconf 명령, 65

조

조정, 일별 명령 요약, 144

종

종료

 강제로 프로그램 종료, 246

 모니터링, 129, 130, 142

주

주별 작업(crontab으로 예약), 107
주소 공간 맵, 163,164

중

중지
시스템 계산, 137
일시적으로 프로세스, 163

직

직렬 포트
어댑터 보드, 22
정의, 22

찾

찾기
및 오래된/비활성 파일 삭제
“삭제”참조
크기 제한을 초과하는 파일, 82
큰 파일, 80,81

총

총 명령 요약, 145,150

최

최대값
nice 번호, 175
최대 크기를 초과하는 파일 찾기, 82
최소값,nice 번호, 175

추

추적 플래그, 164

총

총돌, 218,247
고객 서비스 및, 212,234
생성된 시스템 정보 표시, 215,239
수행할 절차, 212,247
재부트 실패 전 작업, 241-242
총돌 덤프 검사, 238,239
총돌 덤프 정보 저장, 233
총돌 덤프 디렉토리, 전체에서 복구, 239
총돌 덤프 정보 저장, 234

칩

칩 다중 스레드 기능을 확인하는 psrinfo 명령 옵션,
psrinfo -p, 56

커

커널 스레드
구조, 156,162
예약 및, 162

코

코어 덤프 구성,coreadm으로 표시, 228
코어 파일
coreadm으로 관리, 226
코어 파일
proc 도구로 검사, 230
코어 파일
자동으로 삭제, 118
코어 파일
찾기 및 삭제, 87
코어 파일 검사,proc 도구 사용, 230
코어 파일 이름 패턴,coreadm으로 설정, 227

콘

콘솔
보조
시스템 재부트 시 사용으로 설정, 223

콘솔 터미널, 변조 속도를 설정하는 방법, 41-42
 콘솔 터미널 변조 속도, eeprom 명령을 사용하여
 설정, 41

퀴

쿼터

개요, 89
 다중 사용자에게 대한 프로토타입, 94
 변경, 99
 사용자
 개별 사용자에게 대해 변경, 101
 설정, 94
 초과 확인, 97
 사용자 쿼터 초과 확인, 97
 설정, 90
 소프트 제한 설정, 90
 일관성 확인, 95
 정보 표시, 97
 제거, 99
 초과 확인, 97-98
 초기화, 95
 크기, 90
 크기, 예, 96
 쿼터 크기 및 끄기, 91
 표시, 97-98
 하드 제한 설정, 90
 확인, 90, 97, 101
 쿼터, UFS
 사용, 89-90
 소프트 제한 기본값 변경, 99-100
 초기화, 91
 파일 시스템에 대한 확인, 98-99
 쿼터 끄기, 91
 쿼터 초기화, 95
 쿼터 크기, 91
 쿼터 크기, 예, 96

크

크기

디렉토리, 82, 84
 파일, 79, 80, 82, 84

큰

큰 파일, 81

터

터미널, 29
 영숫자, 21
 유형 구별, 21
 정의, 21
 직렬 포트 도구 개요, 26
 직렬 포트 도구 항목 설명, 26
 프로세스 제어, 162
 터미널 및 모뎀 설정, 작업 맵, 25-26

툴

툴

프로세스, 164
 프로세스 정보 표시, 163

파

파일

fstat 및 fcntl 정보 표시, 163, 164, 166
 검색 경로 설정, 254
 계산, 148, 150
 사용량 모니터링, 128, 144
 삭제
 “삭제”참조
 액세스 작업 확인, 188, 189
 정보 표시
 나열, 79, 80
 크기, 79, 80, 82, 84
 크기, 79, 80, 82, 84
 크기 제한을 초과하는 파일 찾기, 82
 크기 표시, 79-80
 훼손 수정
 wtmpx 파일, 140
 파일 또는 그룹 소유권, 파일 액세스 문제 해결, 256
 파일 시스템
 디스크 공간 사용, 185
 마운트 지점, 186

파일 시스템 (계속)

복원, 133, 144

편

편집

crontab 파일, 110, 111

포

포트, 30-31

상태(표), 53

정의, 21

초기화 프로세스, 37

포트 모니터

ttypmon 및 listen(정의), 22, 38-40

상태(표), 52

정의, 22

포트 번호 충돌

공통 에이전트 컨테이너 공유 구성 요소

문제 해결, 249-251

포트 번호(확인 방법)

공통 에이전트 컨테이너 공유 구성 요소

cacao, 250

표

표시

acct.h 형식 파일, 146, 147

at 작업, 121

coreadm으로 코어 덤프 구성, 228

crontab 파일, 112, 113

LWP 정보, 163

pacctn 파일, 146, 147

날짜 및 시간, 66

디렉토리 정보, 79, 80, 82

부트 메시지, 216

시스템 작업 정보, 188, 206

시스템 정보

명령, 60, 66

시스템에 설치된 메모리, 65

연결된 라이브러리, 163, 164

표시 (계속)

예약 클래스 정보, 162, 172

우선 순위 정보, 162, 172

충돌 정보, 215, 239

쿼터, 97-98

쿼터 정보, 90, 97

파일 시스템 정보, 84

파일 정보

du 명령 사용, 82

최신 파일 나열, 85

파일 크기, 79, 80

파일 크기, 79-80

프로세스 정보(방법), 165-166

호스트 ID, 64-65

프

프로그램

강제로 실행 종료, 246

디스크 종속성, 189

중단, 247

프로그램 중단, 247

프로세스

nice 번호, 162, 175, 177

proc 툴 명령, 163

proc 툴 명령을 사용하여 정보 표시, 164

proc 툴을 사용하여 정보 표시, 163

runaway, 177

계산 유틸리티, 128, 129, 143, 144

구조, 156, 162

다시 시작, 163

문제 해결, 177

스택 추적, 164

신호 작업, 164

연결된 라이브러리, 163, 164

열린 파일에 대한 fstat 및 fcntl 정보, 163, 164, 166

예약 클래스, 171

변경, 174

우선 순위 레벨, 171

우선 순위 레벨 및, 174

우선 순위 변경, 174, 175

정보 표시, 162, 172

지정, 173

프로세스 (계속)

- 용어, 155, 156
- 우선 순위, 175
 - 개요, 171, 175
 - 변경, 174, 175
 - 사용자 모드 우선 순위, 171
 - 시간 공유 프로세스 우선 순위 변경, 174, 175
 - 예약 클래스, 171
 - 예약 클래스 및, 174
 - 전역 우선 순위, 171, 172
 - 정보 표시, 162, 172
 - 지정, 173, 174
- 응용 프로그램 스레드 및, 155, 156
- 일시적으로 중지, 163
- 정보 표시, 162
 - acctcom 명령, 146, 147
 - LWP, 163
 - priocntl 명령, 172
 - ps 명령, 162, 165, 172
 - 사용 불가능 프로세스, 146
 - 실행 중인 프로세스 나열, 165
 - 일별 사용량 보고서, 143, 144
 - 프로세스 나열, 164
- 정보 표시(방법), 165-166
- 정의, 155
- 제어, 166-167
- 종료, 164, 167
- 주소 공간 맵, 163, 164
- 추적 플래그, 163, 164
- 툴 명령, 164
- 트리, 163, 164, 166
- 현재 작업 디렉토리, 163, 164, 166
- 프로세스 계산, 128, 129, 143, 144
 - 이유 레코드, 130
- 프로세스 종료, 164, 167
- 프로세스 파일 시스템(PROCFS), 163
- 프로세스별 코어 파일 경로, coreadm으로 설정, 226

확

- 확인
 - UFS 쿼터, 97
 - 쿼터, 101

활

- 활성 파일, 139

회

- 회선 규칙, 37
- 회선 사용량
 - /var/adm/acct/nite/lineuse 파일, 151
 - 연결 계산, 127
 - 일별 보고서, 142
- 회선 사용량 모니터링, 143

호

- 호스트 이름, 변경, 71-72

