Gestion des informations système, des processus et des performances dans Oracle® Solaris 11.2

Référence: E53834-02
Septembre 2014

Table des matières

Utilisation de la présente documentation

- **Présentation** : décrit les tâches permettant de gérer les informations système et les processus, ainsi que de surveiller les performances
- **Public visé** : administrateurs système qui utilisent la version Oracle Solaris 11
- **Connaissances requises** : expérience d'administration de systèmes UNIX

Bibliothèque de la documentation des produits

Les informations de dernière minute et les problèmes connus pour ce produit sont inclus dans la bibliothèque de documentation accessible à l'adresse : http://www.oracle.com/pls/topic/lookup?ctx=E56338.

Accès aux services de support Oracle

Les clients Oracle ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info ou le site http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs si vous êtes malentendant.

Commentaires en retour

Faites part de vos commentaires sur cette documentation à la page suivante : http://www.oracle.com/goto/docfeedback.

♦ ♦ ♦ CHAPITRE 1

Gestion des informations système

Ce chapitre décrit les tâches requises pour afficher et modifier les informations système les plus courantes.

Pour plus d'informations sur la gestion des ressources qui permettent d'allouer, de surveiller et de contrôler les ressources système de façon souple, reportez-vous au Chapitre 1, “ Introduction à la gestion des ressources ” du manuel “ Administration de la gestion des ressources dans Oracle Solaris 11.2 ”.

La liste suivante répertorie les informations disponibles dans ce chapitre :

Affichage des informations système

Cette section décrit les commandes qui vous permettent de modifier les informations générales sur le système.

Commandes servant à afficher les informations système

TABLEAU 1-1 Commandes d'affichage des informations système

Commande	Informations système affichées	Page de manuel
`date`	Date et heure	`date`(1)
`ID hôte`	Numéro d'ID hôte	`hostid`(1)
`isainfo`	Nombre de bits pris en charge par les applications *natives* du système en cours d'exécution, qui peut être transmis sous la forme d'un jeton aux scripts	`isainfo`(1)
`isalist`	Type de processeur	`isalist`(1)

Commande	Informations système affichées	Page de manuel
`prtconf`	Informations sur la configuration du système, la mémoire installée, les propriétés du périphérique et le nom du produit	`prtconf`(1M)
`prtdiag`	Informations de configuration et de diagnostic d'un système, y compris les éventuelles unités remplaçables sur site (FRU)	`prtdiag`(1M)
`psrinfo`	Informations sur le processeur	`psrinfo`(1M)
`uname`	Nom du système d'exploitation, version, nom de noeud, nom du matériel et type de processeur	`uname`(1)

Affichage des informations de version d'un système

Affichez le contenu du fichier `/etc/release` pour identifier la version dont vous disposez.

```
$ cat /etc/release
```

Affichage de la date et de l'heure

Pour afficher la date et l'heure actuelles en fonction de l'horloge système, utilisez la commande `date`.

L'exemple suivant illustre la sortie de la commande `hostid`.

```
$ date
Fri Jun  1 16:07:44 MDT 2012
$
```

Affichage du numéro d'ID hôte d'un système

Pour afficher le numéro d'ID hôte dans un format numérique (hexadécimal), utilisez la commande `hostid`.

L'exemple suivant illustre la sortie de la commande `hostid`.

```
$ hostid
80a5d34c
```

Affichage du type d'architecture d'un système

Utilisez la commande `isainfo` pour afficher le type d'architecture et les noms des jeux d'instructions natifs des applications prises en charge par le système d'exploitation en cours.

L'exemple de sortie suivant provient d'un système x86 :

```
$ isainfo
amd64 i386
```

Par exemple, l'exemple de sortie suivant provient d'un système basé sur SPARC, procédez comme suit :

```
$ isainfo
sparcv9 sparc
```

La commande `isainfo -v` affiche la prise en charge d'applications 32 bits et 64 bits. Par exemple, l'exemple de sortie suivant provient d'un système basé sur SPARC, procédez comme suit :

```
$ isainfo -v
64-bit sparcv9 applications
        asi_blk_init
32-bit sparc applications
        asi_blk_init v8plus div32 mul32
#
```

L'exemple suivant illustre la sortie de la commande `isainfo-v` à partir d'un système basé sur x86 :

```
$ isainfo -v
64-bit amd64 applications
        sse4.1 ssse3 ahf cx16 sse3 sse2 sse fxsr mmx cmov amd_sysc cx8 tsc fpu
32-bit i386 applications
        sse4.1 ssse3 ahf cx16 sse3 sse2 sse fxsr mmx cmov sep cx8 tsc fpu
```

Reportez-vous à la page de manuel `isainfo`(1).

Pour plus d'informations, reportez-vous à la page de manuel `isainfo`(1).

Affichage du type de processeur d'un système

Utilisez la commande `isalist` pour afficher le type de processeur d'un système.

```
The following sample output is from an x86 based system:

$ isalist
```

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

L'exemple de sortie suivant provient d'un système SPARC :

```
$ isalist
sparcv9 sparcv8plus sparcv8 sparcv8-fsmuld sparcv7 sparc sparcv9+vis sparcv9+vis2 \
sparcv8plus+vis sparcv8plus+vis2
```

Reportez-vous à la page de manuel `isalist`(1).

Affichage du nom de produit d'un système

Pour afficher le nom de produit du système, utilisez la commande `prtconf` avec l'option `-b` :

```
$ prtconf -b
```

Pour plus d'informations, reportez-vous à la page de manuel `prtconf`(1M).

L'exemple suivant illustre un exemple de sortie de la commande `prtconf -b` sur un système basé sur SPARC.

```
$ prtconf -b
name:  ORCL,SPARC-T4-2
banner-name:  SPARC T4-2
compatible: 'sun4v'
$
```

L'exemple suivant illustre un exemple de sortie de la commande `prtconf -vb` sur un système basé sur SPARC. La nouvelle option `-v` permet de définir une sortie détaillée.

```
$ prtconf -vb
name:  ORCL,SPARC-T3-4
banner-name:  SPARC T3-4
compatible: 'sun4v'
idprom: 01840014.4fa02d28.00000000.a02d28de.00000000.00000000.00000000.00000000
openprom model:  SUNW,4.33.0.b
openprom version: 'OBP 4.33.0.b 2011/05/16 16:26'
```

Affichage de la mémoire installée d'un système

Pour afficher la quantité de mémoire installée sur votre système, utilisez la commande `prtconf` avec la commande `grep Memory`. L'exemple suivant illustre la sortie sur lequel la commande `grep Memory` sélectionne la sortie de la commande `prtconf` pour afficher uniquement les informations sur la mémoire.

```
$ prtconf | grep Memory
```

```
Memory size: 523776 Megabytes
```

Affichage des valeurs des propriétés par défaut et personnalisées d'un périphérique

Pour afficher à la fois les valeurs de propriétés par défaut et personnalisées de périphériques, utilisez la commande `prtconf` avec l'option `-u`.

```
$ prtconf -u
```

La sortie de la commande `prtconf -u` affiche les propriétés par défaut et personnalisées pour tous les pilotes qui se trouvent sur le système.

Pour plus d'informations sur cette option, reportez-vous à la page de manuel `prtconf`(1M).

EXEMPLE 1-1 SPARC: Affichage des propriétés par défaut et personnalisées d'un périphérique

Cet exemple montre les propriétés par défaut et personnalisées du fichier `bge.conf`. Notez que les fichiers de configuration fournis par l'éditeur sont placés dans les répertoires `/kernel` et `/platform`, tandis que les fichiers de configuration de pilote modifiés correspondants se trouvent dans le répertoire `/etc/driver/drv`.

```
$ prtconf -u
System Configuration:  Oracle Corporation  sun4v
Memory size: 523776 Megabytes
System Peripherals (Software Nodes):

ORCL,SPARC-T3-4
    scsi_vhci, instance #0
        disk, instance #4
        disk, instance #5
        disk, instance #6
        disk, instance #8
        disk, instance #9
        disk, instance #10
        disk, instance #11
        disk, instance #12
    packages (driver not attached)
        SUNW,builtin-drivers (driver not attached)
        deblocker (driver not attached)
        disk-label (driver not attached)
        terminal-emulator (driver not attached)
        dropins (driver not attached)
        SUNW,asr (driver not attached)
        kbd-translator (driver not attached)
        obp-tftp (driver not attached)
        zfs-file-system (driver not attached)
        hsfs-file-system (driver not attached)
```

```
    chosen (driver not attached)
    openprom (driver not attached)
        client-services (driver not attached)
    options, instance #0
    aliases (driver not attached)
    memory (driver not attached)
    virtual-memory (driver not attached)
    iscsi-hba (driver not attached)
        disk, instance #0 (driver not attached)
    virtual-devices, instance #0
        flashprom (driver not attached)
        tpm, instance #0 (driver not attached)
        n2cp, instance #0
        ncp, instance #0
        random-number-generator, instance #0
        console, instance #0
        channel-devices, instance #0
            virtual-channel, instance #0
            virtual-channel, instance #1
            virtual-channel-client, instance #2
            virtual-channel-client, instance #3
            virtual-domain-service, instance #0
    cpu (driver not attached)
    cpu (driver not attached)
    cpu (driver not attached)
    cpu (driver not attached)
    cpu (driver not attached)
    cpu (driver not attached)
    cpu (driver not attached)
```

EXEMPLE 1-2 x86: Affichage des propriétés par défaut et personnalisées d'un périphérique

Cet exemple montre les propriétés par défaut et personnalisées du fichier `bge.conf`. Notez que les fichiers de configuration fournis par l'éditeur sont placés dans les répertoires `/kernel` et `/platform`, tandis que les fichiers de configuration de pilote modifiés correspondants se trouvent dans le répertoire `/etc/driver/drv`.

```
$ prtconf -u
System Configuration:  Oracle Corporation  i86pc
Memory size: 8192 Megabytes
System Peripherals (Software Nodes):

i86pc
    scsi_vhci, instance #0
    pci, instance #0
        pci10de,5e (driver not attached)
        isa, instance #0
            asy, instance #0
            motherboard (driver not attached)
            pit_beep, instance #0
        pci10de,cb84 (driver not attached)
        pci108e,cb84, instance #0
```

```
            device, instance #0
                keyboard, instance #0
                mouse, instance #1
        pci108e,cb84, instance #0
        pci-ide, instance #0
            ide, instance #0
                sd, instance #0
            ide (driver not attached)
        pci10de,5c, instance #0
            display, instance #0
        pci10de,cb84, instance #0
        pci10de,5d (driver not attached)
        pci10de,5d (driver not attached)
        pci10de,5d (driver not attached)
        pci10de,5d (driver not attached)
        pci1022,1100, instance #0
        pci1022,1101, instance #1
        pci1022,1102, instance #2
        pci1022,1103 (driver not attached)
        pci1022,1100, instance #3
        pci1022,1101, instance #4
        pci1022,1102, instance #5
        pci1022,1103 (driver not attached)
    pci, instance #1
        pci10de,5e (driver not attached)
        pci10de,cb84 (driver not attached)
        pci10de,cb84, instance #1
        pci10de,5d (driver not attached)
        pci10de,5d (driver not attached)
        pci10de,5d (driver not attached)
        pci10de,5d (driver not attached)
        pci1022,7458, instance #1
        pci1022,7459 (driver not attached)
        pci1022,7458, instance #2
            pci8086,1011, instance #0
            pci8086,1011, instance #1
            pci1000,3060, instance #0
                sd, instance #1
                sd, instance #2
        pci1022,7459 (driver not attached)
    ioapics (driver not attached)
        ioapic, instance #0 (driver not attached)
        ioapic, instance #1 (driver not attached)
    fw, instance #0
        cpu (driver not attached)
        cpu (driver not attached)
        cpu (driver not attached)
        cpu (driver not attached)
        sb, instance #1
    used-resources (driver not attached)
    iscsi, instance #0
    fcoe, instance #0
    pseudo, instance #0
    options, instance #0
```

```
    xsvc, instance #0
    vga_arbiter, instance #0
```

EXEMPLE 1-3 x86: Affichage des informations de configuration du système

L'exemple suivant illustre l'utilisation de la commande `prtconf` avec l'option `-v` sur un système x86 en vue d'identifier les périphériques de disque, à bande et DVD connectés à un système. La sortie de cette commande affiche "driver not attached" à côté des instances de périphérique pour lesquelles aucun périphérique n'existe.

```
$ prtconf -v | more
System Configuration:  Oracle Corporation  i86pc
Memory size: 8192 Megabytes
System Peripherals (Software Nodes):

i86pc
    System properties:
        name='#size-cells' type=int items=1
            value=00000002
        name='#address-cells' type=int items=1
            value=00000003
        name='relative-addressing' type=int items=1
            value=00000001
        name='MMU_PAGEOFFSET' type=int items=1
            value=00000fff
        name='MMU_PAGESIZE' type=int items=1
            value=00001000
        name='PAGESIZE' type=int items=1
            value=00001000
        name='acpi-status' type=int items=1
            value=00000013
        name='biosdev-0x81' type=byte items=588
            value=01.38.74.0e.08.1e.db.e4.fe.00.d0.ed.fe.f8.6b.04.08.d3.db.e4.fe
.
.
.
```

Pour plus d'informations, reportez-vous aux pages de manuel `driver`(4), `driver.conf`(4) et `prtconf`(1M).

Pour obtenir des instructions sur la création de fichiers de configuration fournis administrativement, reportez-vous auChapitre 1, “ Gestion de périphériques dans Oracle Solaris ” du manuel “ Gestion des périphériques dans Oracle Solaris 11.2 ”.

Affichage des informations de diagnostic du système

Utilisez la commande `prtdiag` pour afficher des informations de configuration et de diagnostic pour un système.

```
$ prtdiag [-v] [-l]
```

`-v` Mode verbose.

`-l` Sortie dans le journal. Si des défaillances ou des erreurs surviennent sur le système, ces informations sont uniquement consignées dans `syslogd`(1M).

EXEMPLE 1-4 SPARC: Affichage des informations de diagnostic du système

L'exemple suivant illustre la sortie de la commande `prtdiag- v` sur un système SPARC. Dans un souci de concision, l'exemple a été tronqué.

```
$ prtdiag -v | more
System Configuration:  Oracle Corporation  sun4v Sun Fire T200
Memory size: 16256 Megabytes

================================ Virtual CPUs ================================


CPU ID Frequency Implementation         Status
------ --------- ---------------------- -------
0      1200 MHz  SUNW,UltraSPARC-T1     on-line
1      1200 MHz  SUNW,UltraSPARC-T1     on-line
2      1200 MHz  SUNW,UltraSPARC-T1     on-line
3      1200 MHz  SUNW,UltraSPARC-T1     on-line
4      1200 MHz  SUNW,UltraSPARC-T1     on-line
5      1200 MHz  SUNW,UltraSPARC-T1     on-line
6      1200 MHz  SUNW,UltraSPARC-T1     on-line
.
.
.
====================== Physical Memory Configuration ======================
Segment Table:
------------------------------------------------------------
Base           Segment  Interleave  Bank     Contains
Address        Size     Factor      Size     Modules
------------------------------------------------------------
0x0            16 GB    4           2 GB     MB/CMP0/CH0/R0/D0
                                             MB/CMP0/CH0/R0/D1
                                    2 GB     MB/CMP0/CH0/R1/D0
                                             MB/CMP0/CH0/R1/D1
                                    2 GB     MB/CMP0/CH1/R0/D0
                                             MB/CMP0/CH1/R0/D1
                                    2 GB     MB/CMP0/CH1/R1/D0
.
.
System PROM revisions:
----------------------
OBP 4.30.4.d 2011/07/06 14:29

IO ASIC revisions:
```

```
-----------------
Location             Path                                  Device
                  Revision
-------------------- ------------------
IOBD/IO-BRIDGE                                    /pci@780      SUNW,sun4v-pci     0
.
.
.
```

EXEMPLE 1-5 x86: Affichage des informations de diagnostic du système

L'exemple suivant montre un exemple de sortie pour la commande `prtdiag -l` sur un système basé sur x86.

```
$ prtdiag -l
System Configuration: ... Sun Fire X4100 M2
BIOS Configuration: American Megatrends Inc. 0ABJX104 04/09/2009
BMC Configuration: IPMI 1.5 (KCS: Keyboard Controller Style)

==== Processor Sockets ====================================

Version                          Location Tag
-------------------------------- ------------------------
Dual-Core AMD Opteron(tm) Processor 2220 CPU 1
Dual-Core AMD Opteron(tm) Processor 2220 CPU 2

==== Memory Device Sockets ================================

Type        Status Set Device Locator      Bank Locator
----------- ------ --- ------------------ ----------------
unknown     empty  0   DIMM0               NODE0
unknown     empty  0   DIMM1               NODE0
DDR2        in use 0   DIMM2               NODE0
DDR2        in use 0   DIMM3               NODE0
unknown     empty  0   DIMM0               NODE1
unknown     empty  0   DIMM1               NODE1
DDR2        in use 0   DIMM2               NODE1
DDR2        in use 0   DIMM3               NODE1

==== On-Board Devices =====================================
 LSI serial-SCSI #1
 Gigabit Ethernet #1
 ATI Rage XL VGA

==== Upgradeable Slots ====================================

ID  Status    Type             Description
--- --------- ---------------- ----------------------------
1   available PCI Express      PCIExp SLOT0
2   available PCI Express      PCIExp SLOT1
3   available PCI-X            PCIX SLOT2
4   available PCI Express      PCIExp SLOT3
5   available PCI Express      PCIExp SLOT4
```

```
$
```

Identification d'informations relatives aux fonctions de chip multithreading

La commande `psrinfo` a été modifiée afin de fournir des informations sur les processeurs physiques et non plus seulement sur les processeurs virtuels. Cette fonctionnalité améliorée a été ajoutée afin de permettre l'identification des fonctions CMT (chip multithreading). La nouvelle option `-p` indique le nombre total de processeurs physiques présents dans le système. L'option `-t` affiche l'arborescence des processeurs du système et qui leur sont associées et cpu socket, les ID de noyau (core).

La commande `psrinfo-pv` permet de répertorier tous les processeurs physiques présents dans le système, ainsi que les processeurs virtuels associés à chaque processeur physique. La sortie par défaut de la commande `psrinfo` continue à afficher les informations de processeur virtuel pour un système.

Pour plus d'informations, reportez-vous à la page de manuel `psrinfo`(1M).

Affichage du type de processeur physique d'un système

Utilisez la commande `psrinfo -p` pour afficher le nombre total de processeurs physiques sur un système.

```
$ psrinfo -p
1
```

Ajoutez l'option `-v` pour également afficher des informations sur que sur le processeur virtuel associé à chaque processeur physique. Par exemple :

```
$ psrinfo -pv
The physical processor has 8 cores and 32 virtual processors (0-31)
  The core has 4 virtual processors (0-3)
  The core has 4 virtual processors (4-7)
  The core has 4 virtual processors (8-11)
  The core has 4 virtual processors (12-15)
  The core has 4 virtual processors (16-19)
  The core has 4 virtual processors (20-23)
  The core has 4 virtual processors (24-27)
  The core has 4 virtual processors (28-31)
    UltraSPARC-T1 (chipid 0, clock 1000 MHz)
```

L'exemple suivant montre un exemple de sortie pour la commande `psrinfo -pv` sur un système basé sur x86.

```
$ psrinfo -pv
The physical processor has 2 virtual processors (0 1)
  x86 (AuthenticAMD 40F13 family 15 model 65 step 3 clock 2793 MHz)
        Dual-Core AMD Opteron(tm) Processor 2220        [ Socket: F(1207) ]
The physical processor has 2 virtual processors (2 3)
  x86 (AuthenticAMD 40F13 family 15 model 65 step 3 clock 2793 MHz)
        Dual-Core AMD Opteron(tm) Processor 2220        [ Socket: F(1207) ]
```

Affichage du type de processeur virtuel d'un système

Utilisez la commande `psrinfo -v` pour afficher des informations sur le type de processeur virtuel d'un système.

```
$ psrinfo -v
```

Sur un système x86, utilisez la commande `isalist` pour afficher le type de processeur virtuel. Par exemple :

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

EXEMPLE 1-6 SPARC: Affichage du type de processeur virtuel d'un système

Cet exemple montre comment afficher les informations relatives au type de processeur virtuel d'un système SPARC.

```
$ psrinfo -v
Status of virtual processor 28 as of: 09/13/2010 14:07:47
  on-line since 04/08/2010 21:27:56.
  The sparcv9 processor operates at 1400 MHz,
        and has a sparcv9 floating point processor.
Status of virtual processor 29 as of: 09/13/2010 14:07:47
  on-line since 04/08/2010 21:27:56.
  The sparcv9 processor operates at 1400 MHz,
        and has a sparcv9 floating point processor.
```

EXEMPLE 1-7 SPARC: Affichage du processeur virtuel associé à chaque processeur physique d'un système

L'exemple suivant illustre la sortie de la commande `psrinfo` lorsqu'elle est exécutée avec les options -pv sur un serveur Oracle SPARC T4-4. La sortie affiche à la fois les informations relatives à la puce (processeur physique) et aux noyaux de l'emplacement du thread. Ces informations peuvent être utiles pour déterminer l'UC physique sur laquelle est placée un thread et la façon dont il est mappé au niveau du noyau.

```
$ psrinfo -pv
The physical processor has 8 cores and 64 virtual processors (0-63)
```

```
  The core has 8 virtual processors (0-7)
  The core has 8 virtual processors (8-15)
  The core has 8 virtual processors (16-23)
  The core has 8 virtual processors (24-31)
  The core has 8 virtual processors (32-39)
  The core has 8 virtual processors (40-47)
  The core has 8 virtual processors (48-55)
  The core has 8 virtual processors (56-63)
    SPARC-T4 (chipid 0, clock 2998 MHz)
The physical processor has 8 cores and 64 virtual processors (64-127)
  The core has 8 virtual processors (64-71)
  The core has 8 virtual processors (72-79)
  The core has 8 virtual processors (80-87)
  The core has 8 virtual processors (88-95)
  The core has 8 virtual processors (96-103)
  The core has 8 virtual processors (104-111)
  The core has 8 virtual processors (112-119)
  The core has 8 virtual processors (120-127)
    SPARC-T4 (chipid 1, clock 2998 MHz)
```

Modification des informations système

Cette section décrit les commandes qui vous permettent de modifier les informations générales sur le système.

Modification des informations système (liste des tâches)

Tâche	Instructions	Pour obtenir des instructions
Réglage manuel de la date et de l'heure d'un système	Réglez manuellement la date et l'heure du système en utilisant la syntaxe de ligne de commande `date` *mmjjHHMM[[ss]aa]*.	“Réglage manuel de la date et de l'heure d'un système” à la page 22
Définition d'un message du jour	Définissez un message du jour sur votre système en modifiant le fichier `/etc/motd` .	“Configuration d'un message du jour” à la page 22
Modification de l'identité d'un système.	Modification de l'identité d'un système à l'aide de la commande `hostname`.	“Modification de l'identité d'un système” à la page 23

▼ Réglage manuel de la date et de l'heure d'un système

1. **Connectez-vous en tant qu'administrateur.**

   Reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

2. **Indiquez la nouvelle date et heure.**

   `$ date` *mmddHHMM[[cc]yy]*

mm	Mois à deux chiffres
jj	Jour du mois à deux chiffres
HH	Heure à deux chiffres (sur 24 heures)
MM	Minutes à deux chiffres
cc	Siècle à deux chiffres
aa	Année à deux chiffres

   Pour plus d'informations, reportez-vous à la page de manuel `date`(1).

3. **Vérifiez que vous avez bien réinitialisé la date du système à l'aide de la commande `date` sans option.**

Exemple 1-8 Réglage manuel de la date et de l'heure d'un système

L'exemple suivant montre comment utiliser la commande `date`pour définir manuellement une date et l'heure du système.

```
# date
Monday, September 13. 2010 02:00:16 PM MDT
# date 0921173404
Thu Sep 17:34:34 MST 2010
```

▼ Configuration d'un message du jour

Vous pouvez modifier le fichier de message du jour, `/etc/motd`, pour inclure des annonces ou demandes d'informations adressées à tous les utilisateurs d'un système lorsqu'ils se connectent.

Utilisez cette fonction avec parcimonie et modifiez régulièrement ce fichier pour supprimer les messages obsolètes.

1. **Prenez un rôle auquel le profil `Administrator Message Edit` a été affecté.**

   Reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

2. **Servez-vous de la commande `pfedit` pour modifier le fichier `/etc/motd` et ajoutez un message de votre choix.**

   ```
   $ pfedit /etc/motd
   ```

   Modifiez le texte pour inclure le message à afficher au cours de la connexion utilisateur. Incluez des espaces, des tabulations et des retours chariot.

3. **Vérifiez les modifications en affichant le contenu du fichier `/etc/motd`.**

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

▼ Modification de l'identité d'un système

1. **Connectez-vous en tant qu'administrateur.**

   Reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

2. **Définissez le nom de l'hôte du système.**

   ```
   # hostname name
   ```

   Les commandes `hostname` et `domainname` permettent de configurer le nom d'hôte et le nom de domainede manière permanente. Lorsque vous utilisez ces commandes, les propriétés SMF correspondantes et le service SMF associé sont également mis à jour de façon automatique.

   Pour plus d'informations, reportez-vous aux pages de manuel `hostname`(1), `domainname`(1M) et `nodename`(4).

♦ ♦ ♦ CHAPITRE 2

Gestion des processus système

Ce chapitre décrit les procédures de gestion des processus système.

Ce chapitre comprend les sections suivantes :

Processus système ne nécessitant aucune administration

Oracle Solaris 10 and Oracle Solaris 11 comprennent des processus système effectuant des tâches données mais ne nécessitant aucune administration.

Processus	Description
`fsflush`	Démon système qui vide des pages sur le disque
`init`	Processus système initial qui démarre et redémarre d'autres processus et composants SMF
`intrd`	Processus système qui surveille et équilibre la charge système due à des interruptions
`kmem_task`	Processus système qui surveille la taille de la mémoire cache
`pageout`	Processus système qui contrôle la pagination de la mémoire sur le disque
`sched`	Processus système responsable de la planification du SE et de l'échange de processus
`vm_tasks`	Processus système composé d'un thread par processeur, qui équilibre et répartit les charges de travail liées à la mémoire virtuelle sur plusieurs CPU afin d'optimiser les performances.
`zpool-`*`pool-name`*	Processus système pour chaque pool de stockage ZFS contenant les threads de tâche d'E/S destinés au pool associé

Gestion des processus système

Cette section décrit les différentes tâches de gestion des processus système.

Liste des tâches pour la gestion des processus système

Tâche	Description	Pour obtenir des instructions
Etablissement de la liste des processus	Utilisez la commande `ps` pour dresser la liste de tous les processus d'un système.	“Etablissement de la liste des processus” à la page 30
Affichage des informations sur les processus	Utilisez la commande `pgrep` pour obtenir les ID des processus dont vous souhaitez afficher plus d'informations.	“Affichage d'informations sur les processus” à la page 31
Contrôle des processus	Recherchez les processus à l'aide de la commande `pgrep`. Ensuite, utilisez la commande `pcommand` (`/proc`) appropriée pour contrôler le processus. Reportez-vous au Tableau 2-2, “Commandes de processus (`/proc`)” pour obtenir une description des commandes (`/proc`).	“Contrôle des processus” à la page 32
Arrêt d'un processus	Recherchez un processus, par nom de processus ou ID de processus. Vous pouvez utiliser la commande `pkill` ou `kill` pour mettre fin au processus.	“Arrêt d'un processus (`pkill`)” à la page 33 “Arrêt d'un processus (`kill`)” à la page 34

Commandes de gestion des processus système

Le tableau suivant décrit les commandes de gestion des processus système.

TABLEAU 2-1 Commandes de gestion des processus

Commande	Description	Page de manuel
`ps`, `pgrep`, `prstat`, `pkill`	Vérifiez l'état des processus actifs sur un système et affichez des informations détaillées sur les processus.	`ps`(1), `pgrep`(1) et`prstat`(1M)
`pkill`	Fonctionne exactement comme `pgrep` mais recherche ou signale les processus par nom ou un autre attribut et met fin au processus.	`pgrep`(1) et `pkill`(1) `kill`(1)

Commande	Description	Page de manuel
	Chaque processus concordant est signalé comme avec la commande `kill`, au lieu de voir son ID de processus imprimé.	
`pargs, preap`	Facilite le débogage des processus.	`pargs`(1) et `preap`(1)
`dispadmin`	Répertorie les stratégies de planification des processus par défaut.	`dispadmin`(1M)
`priocntl`	Affecte les processus à une classe de priorité et gère les priorités des processus.	`priocntl`(1)
`nice`	Change la priorité d'un processus de partage du temps.	`nice`(1)
`psrset`	Lie des groupes de traitement à un groupe de processeurs plutôt qu'à un seul processeur.	`psrset`(1M)

Utilisation de la commande `ps`

La commande `ps` vous permet de vérifier l'état des processus actifs sur un système et d'afficher des informations techniques sur les processus. Ces données sont utiles pour les tâches d'administration telles que la détermination des priorités des processus.

En fonction des options utilisées, la commande `ps` rapporte les informations suivantes :

- Etat actuel du processus
- ID de processus
- ID du processus parent
- ID utilisateur
- Classe de programmation
- Priorité
- Adresse du processus
- Mémoire utilisée
- Temps CPU utilisé

La liste suivante décrit quelques-uns des champs rapportés par la commande `ps`. Les champs affichés dépendent de l'option choisie. Pour une description de toutes les options disponibles, reportez-vous à la page de manuel `ps`(1).

`UID` ID utilisateur effectif du propriétaire du processus.

`PID` L'ID de processus.

`PPID` ID du processus parent.

C	Utilisation du processeur à des fins de programmation. Ce champ n'est pas affiché lorsque l'option -c est utilisée.
CLS	Classe de programmation à laquelle appartient le processus, telle qu'en temps réel, système ou partage du temps. Ce champ est inclus uniquement avec l'option -c.
PRI	Priorité de programmation du thread de noyau. Un nombre plus élevé indique une priorité plus élevée.
NI	Numéro nice du processus, qui contribue à sa priorité de programmation. Rendre un processus plus "nice" équivaut à abaisser sa priorité.
ADDR	Adresse de la structure proc.
SZ	Taille d'adresse virtuelle du processus.
WCHAN	Adresse d'un événement ou verrou pour lequel le processus est en veille.
STIME	Heure de début du processus (en heures, minutes et secondes).
TTY	Terminal à partir duquel le processus, ou son parent, a été démarré. Un point d'interrogation indique qu'il n'y a aucun terminal de contrôle.
TIME	Quantité totale du temps CPU utilisé par le processus depuis son démarrage.
CMD	Commande qui a généré le processus.

Utilisation du système de fichiers et des commandes /proc

Vous pouvez afficher des informations détaillées sur les processus du répertoire /proc à l'aide des commandes de processus. Le tableau suivant répertorie les commandes du processus /proc. Le répertoire /proc est également connu comme le système de fichiers du processus (PROCFS). Les images des processus actifs sont stockées dans le PROCFS en fonction du numéro d'identification du processus.

TABLEAU 2-2 Commandes de processus (/proc)

Commande du processus	Description
pcred	Affiche des informations d'identification du processus.
pfiles	Indique les informations fstat et fcntl relatives aux fichiers ouverts dans un processus.
pflags	Affiche les indicateurs de suivi /proc, les signaux en attente et en suspens, et d'autres informations d'état.

Commande du processus	Description
`pldd`	Répertorie les bibliothèques dynamiques liées à un processus.
`pmap`	Affiche la configuration de l'espace d'adressage de chaque processus.
`psig`	Répertorie les actions de signal et les gestionnaires de chaque processus.
`prun`	Démarre chaque processus.
`pstack`	Affiche un suivi de pile hex+symbolique pour chaque processus léger dans chaque processus.
`pstop`	Arrête chaque processus.
`ptime`	Comptabilise le temps d'un processus en utilisant la comptabilisation des micro-états.
`ptree`	Affiche les arborescences de processus contenant le processus.
`pwait`	Affiche les informations d'état une fois qu'un processus se termine.
`pwdx`	Affiche le répertoire de travail actuel d'un processus.

Pour plus d'informations, reportez-vous à la page de manuel `proc`(1) .

Les outils de processus sont similaires à certaines options de la commande `ps`, à l'exception du fait que la sortie fournie par ces commandes est plus détaillée.

En règle générale, les commandes du processus effectuent les opérations suivantes :

- Affiche plus d'informations sur les processus comme les répertoires de travail `fstat` et `fcntl` et les arborescences des processus parents et enfants
- Procure le contrôle des processus en permettant aux utilisateurs de les arrêter ou de les reprendre

Gestion des processus à l'aide des commandes de processus (`/proc`)

Vous pouvez afficher des informations techniques détaillées sur les processus ou contrôler les processus actifs à l'aide de certaines commandes de processus. Le Tableau 2-2, “Commandes de processus (`/proc`)” répertorie certaines commandes `/proc`.

Si un processus est piégé dans une boucle infinie ou si son exécution prend trop de temps, vous pouvez arrêter le processus. Pour plus d'informations sur l'arrêt des processus à l'aide de la commande `kill` ou `pkill`, reportez-vous au Chapitre 2, Gestion des processus système.

Le système de fichiers `/proc` est une hiérarchie de répertoires qui contient des sous-répertoires supplémentaires pour les informations d'état et les fonctions de contrôle.

Le système de fichiers `/proc` fournit également une fonction `xwatchpoint` qui sert à reconfigurer les autorisations de lecture/écriture sur les différentes pages de l'espace d'adressage d'un processus. Cette fonction n'a pas de restrictions et est MT-safe.

Les outils de débogage ont été modifiés de façon à utiliser la fonction `xwatchpoint`, ce qui signifie que l'ensemble du processus `xwatchpoint` est plus rapide.

Les restrictions suivantes ont été supprimées lorsque vous définissez des `xwatchpoints`à l'aide de l'outil de débogage `dbx` :

- Définition des `xwatchpoint` sur des variables locales de la pile en raison des fenêtres de registre du système SPARC.
- Définition de `xwatchpoints` sur les processus multithreaded.

Pour plus d'informations, reportez-vous aux pages de manuel `proc`(4) et`mdb`(1).

▼ Etablissement de la liste des processus

- **Utilisez la commande `ps` pour dresser la liste de tous les processus d'un système.**

```
$ ps [-efc]
```

`ps`	Affiche uniquement les processus associés à votre session de connexion.
`-ef`	Affiche des informations complètes sur tous les processus en cours d'exécution sur le système.
`-c`	Affiche les informations sur le planificateur de processus.

Exemple 2-1 Liste des processus

L'exemple suivant illustre la sortie de la commande `ps` lorsque aucune option n'est utilisée.

```
$ ps
   PID TTY      TIME COMD
  1664 pts/4    0:06 csh
  2081 pts/4    0:00 ps
```

L'exemple suivant montre la sortie de la commande `ps -ef`. Cette sortie indique que le premier processus exécuté lorsque le système s'initialise est `sched` (le swappeur) suivi du processus `init`, `pageout`, et ainsi de suite.

```
$ ps -ef
 UID   PID  PPID   C    STIME TTY         TIME CMD
    root     0     0   0 18:04:04 ?           0:15 sched
    root     5     0   0 18:04:03 ?           0:05 zpool-rpool
    root     1     0   0 18:04:05 ?           0:00 /sbin/init
    root     2     0   0 18:04:05 ?           0:00 pageout
    root     3     0   0 18:04:05 ?           2:52 fsflush
    root     6     0   0 18:04:05 ?           0:02 vmtasks
  daemon   739     1   0 19:03:58 ?           0:00 /usr/lib/nfs/nfs4cbd
```

```
    root     9     1   0 18:04:06 ?           0:14 /lib/svc/bin/svc.startd
    root    11     1   0 18:04:06 ?           0:45 /lib/svc/bin/svc.configd
  daemon   559     1   0 18:04:49 ?           0:00 /usr/sbin/rpcbind
  netcfg    47     1   0 18:04:19 ?           0:01 /lib/inet/netcfgd
   dladm    44     1   0 18:04:17 ?           0:00 /sbin/dlmgmtd
  netadm    51     1   0 18:04:22 ?           0:01 /lib/inet/ipmgmtd
    root   372   338   0 18:04:43 ?           0:00 /usr/lib/hal/hald-addon-cpufreq
    root    67     1   0 18:04:30 ?           0:02 /lib/inet/in.mpathd
    root   141     1   0 18:04:38 ?           0:00 /usr/lib/pfexecd
  netadm    89     1   0 18:04:31 ?           0:03 /lib/inet/nwamd
    root   602     1   0 18:04:50 ?           0:02 /usr/lib/inet/inetd start
    root   131     1   0 18:04:35 ?           0:01 /sbin/dhcpagent
  daemon   119     1   0 18:04:33 ?           0:00 /lib/crypto/kcfd
    root   333     1   0 18:04:41 ?           0:07 /usr/lib/hal/hald --daemon=yes
    root   370   338   0 18:04:43 ?           0:00 /usr/lib/hal/hald-addon-network-discovery
    root   159     1   0 18:04:39 ?           0:00 /usr/lib/sysevent/syseventd
    root   236     1   0 18:04:40 ?           0:00 /usr/lib/ldoms/drd
    root   535     1   0 18:04:46 ?           0:09 /usr/sbin/nscd
    root   305     1   0 18:04:40 ?           0:00 /usr/lib/zones/zonestatd
    root   326     1   0 18:04:41 ?           0:03 /usr/lib/devfsadm/devfsadmd
    root   314     1   0 18:04:40 ?           0:00 /usr/lib/dbus-daemon --system
    .
  .
  .
```

▼ Affichage d'informations sur les processus

1. Obtenez l'ID du processus dont vous souhaitez afficher plus d'informations.

```
# pgrep process
```

L'ID de processus s'affiche dans la première colonne de la sortie.

2. Affichez les informations relatives aux processus.

```
# /usr/bin/pcommand PID
```

pcommand — La commande de traitement que vous souhaitez exécuter. Tableau 2-2, “Commandes de processus (`/proc`)” répertorie et décrit ces commandes.

PID — Identifie l'ID de processus.

Exemple 2-2 Affichage des informations sur les processus

L'exemple suivant montre comment utiliser les commandes du processus pour afficher plus d'informations sur un processus `cron`.

```
# pgrep cron        Obtains the process ID for the cron process
4780
# pwdx 4780         Displays the current working directory for the cron process
```

```
4780:   /var/spool/cron/atjobs
# ptree 4780      Displays the process tree that contains the cron process
4780  /usr/sbin/cron
# pfiles 4780      Displays fstat and fcntl information
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
```

▼ Contrôle des processus

1. Obtenez l'ID du processus à contrôler.

```
# pgrep process
```

L'ID de processus apparaît dans la première colonne de la sortie.

2. Utilisez la commande de processus appropriée pour contrôler le processus.

```
# /usr/bin/pcommand PID
```

pcommand — La commande de traitement que vous souhaitez exécuter. Tableau 2-2, “Commandes de processus (`/proc`)” répertorie et décrit ces commandes.

PID — Identifie l'ID de processus.

3. Vérifiez l'état du processus.

```
# ps -ef | grep PID
```

Arrêt d'un processus (`pkill, kill`)

Il peut s'avérer nécessaire d'arrêter (interrompre) un processus qui se trouve dans une boucle infinie ou arrêter un travail volumineuses, avant qu'il ne soit terminé. Vous pouvez interrompre

un processus que vous possédez. Un superutilisateur peut interrompre tout processus du système à l'exception des processus dotés des ID 0, 1, 2, 3 et 4. L'interruption de ces processus risque de provoquer la panne du système.

Pour plus d'informations à ce sujet, reportez-vous aux pages de manuel pgrep(1), pkill(1) etkill(1).

▼ Arrêt d'un processus (pkill)

1. **Pour mettre fin au processus d'un autre utilisateur, prenez le rôle root.**

2. **Obtenez l'ID du processus à terminer.**

```
$ pgrep process
```

Par exemple :

```
$ pgrep netscape
587
566
```

L'ID de processus s'affiche dans la sortie.

Remarque - Pour obtenir des informations sur les processus d'un système Sun Ray™, utilisez les commandes suivantes :

Pour obtenir la liste de tous les processus utilisateur, procédez comme suit :

```
# ps -fu user
```

Pour localiser un de ces processus appartenant à un utilisateur, procédez comme suit :

```
# ps -fu user | grep process
```

---

3. **Mettez fin au processus.**

```
$ pkill [signal] PID
```

signal — Lorsque aucun signal n'est inclus dans la syntaxe de ligne de commande pkill, le signal utilisé par défaut est –15 (SIGKILL). L'utilisation du signal –9 (SIGTERM) avec la commande pkill garantit la fin du processus dans les plus brefs délais. Cependant, le signal –9 ne doit pas être utilisé pour arrêter certains processus, par exemple un processus de base de données ou de serveur LDAP.

PID — Représente le nom du processus à arrêter.

Astuce - Lorsque vous utilisez la commande `pkill` pour interrompre un processus, utilisez d'abord la commande proprement dite, sans y inclure une option de signal. Si le traitement ne prend pas fin après quelques minutes, utilisez la commande `pkill` avec le signal -9.

4. **Vérifiez que le processus a été interrompu.**

```
$ pgrep process
```

Le processus interrompu ne doit plus figurer dans la sortie de la commande `pgrep`.

▼ Arrêt d'un processus (`kill`)

1. **Pour mettre fin au processus d'un autre utilisateur, prenez le rôle `root`.**

2. **Obtenez l'ID du processus à arrêter.**

```
# ps -fu user
```

où *user* est le propriétaire du processus.

L'ID de processus s'affiche dans la première colonne de la sortie.

3. **Mettez fin au processus.**

```
# kill [signal-number] PID
```

signal	Lorsque aucun signal n'est inclus dans la syntaxe de ligne de commande `kill`, le signal utilisé par défaut est –15 (`SIGKILL`). L'utilisation du signal –9 (`SIGTERM`) avec la commande `kill` garantit la fin du processus dans les plus brefs délais. Cependant, le signal –9 ne doit pas être utilisé pour arrêter certains processus, par exemple un processus de base de données ou de serveur LDAP.
PID	Représente l'ID du processus à arrêter.

Astuce - Lorsque vous utilisez la commande `kill` pour arrêter un processus, utilisez d'abord la commande proprement dite, sans y inclure une option de signal. Attendez quelques minutes pour voir si le processus se termine avant d'utiliser la commande `kill` avec le signal -9.

4. **Vérifiez que le processus a été interrompu.**

```
$ ps
```

Le processus interrompu ne doit plus figurer dans la sortie de la commande `pgrep`.

Débogage d'un processus (pargs, preap)

Les commandes pargs et preap améliorent le débogage des processus. La commande pargs imprime les arguments et les variables d'environnement associées à un processus en direct ou à un dump noyau. La commande preap supprime les processus défunts (zombies). Un processus zombie n'a pas encore vu son état de sortie réclamé par son parent. Ces processus sont généralement inoffensifs mais peuvent consommer des ressources système s'ils sont nombreux. Vous pouvez utiliser les commandes pargs et preap pour examiner les processus dont vous souhaitez examiner les privilèges. Lorsque vous vous connectez en tant qu'administrateur, vous pouvez examiner les processus.

Pour plus d'informations sur l'utilisation de la commande preap, reportez-vous à la page de manuel preap(1). Pour plus d'informations sur l'utilisation de la commande pargs, reportez-vous à la page de manuel pargs(1). Reportez-vous également à la page de manuel proc(1).

EXEMPLE 2-3 Débogage d'un processus (pargs)

La commande pargs résout un problème de longue date lié à l'impossibilité d'afficher avec la commande ps tous les arguments transmis à un processus. L'exemple suivant illustre comment utiliser la commande pargs avec la commande pgrep pour afficher les arguments transmis à un processus.

```
# 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
```

L'exemple suivant illustre comment utiliser la commande pargs- e pour afficher les variables d'environnement associées à un processus.

```
$ pargs -e 6763
6763: tcsh
envp[0]: DISPLAY=:0.0
```

Affichage et gestion des informations sur les classes de processus

Vous pouvez configurer les classes de programmation des processus sur votre système et la plage de priorité de l'utilisateur pour la classe de partage du temps.

Les classes de programmation de processus possibles sont les suivantes :

- Partage équitable (`FSS`)
- Fixe (`FX`)
- Système (`SYS`)
- Interactif (`IA`)
- Temps réel (`RT`)
- Partage de temps (`TS`)
  - La priorité définie par l'utilisateur se situe dans une plage de -60 à +60.
  - La priorité d'un processus est héritée d'un processus parent. Cette priorité est désignée comme la *priorité en mode utilisateur*.
  - Le système vérifie la priorité mode utilisateur dans le tableau de paramètres de programmation du partage de temps. Ensuite, le système l'ajoute dans la priorité `nice` ou `priocntl` (fournie par l'utilisateur) et garantit une plage de 0–59 pour créer une *priorité globale*.

Affichage des informations sur les classes de processus

Cette section comprend les sections suivantes :

“Affichage des informations de priorité de traitement” à la page 36 Utilisez la commande `priocntl -l` pour afficher les classes de programmation et les plages de priorité des processus.

“Affichage de la priorité globale d'un processus” à la page 37 Utilisez la commande `ps -ecl` pour afficher la priorité globale d'un processus.

Affichage des informations de priorité de traitement

Utilisez la commande `priocntl -l` pour afficher les classes de planification du processus et les priorités.

```
$ priocntl -l
```

L'exemple suivant montre la sortie de la commande `priocntl -l`.

```
# 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
```

Affichage de la priorité globale d'un processus

Utilisez la commande `ps` pour afficher la priorité globale d'un processus.

```
$ ps -ecl
```

La priorité globale figure sous la colonne `PRI`.

L'exemple suivant illustre la sortie de la commande `ps -ecl`. Les valeurs de la colonne `PRI` indiquent la priorité de chaque processus.

```
$ ps -ecl
 F S    UID   PID  PPID  CLS PRI     ADDR     SZ    WCHAN TTY         TIME CMD
 1 T      0     0     0  SYS  96        ?      0          ?        0:11 sched
 1 S      0     5     0  SDC  99        ?      0        ? ?        0:01 zpool-rp
 0 S      0     1     0   TS  59        ?    688        ? ?        0:00 init
 1 S      0     2     0  SYS  98        ?      0        ? ?        0:00 pageout
 1 S      0     3     0  SYS  60        ?      0        ? ?        2:31 fsflush
 1 S      0     6     0  SDC  99        ?      0        ? ?        0:00 vmtasks
 0 S     16    56     1   TS  59        ?   1026        ? ?        0:01 ipmgmtd
 0 S      0     9     1   TS  59        ?   3480        ? ?        0:04 svc.star
 0 S      0    11     1   TS  59        ?   3480        ? ?        0:13 svc.conf
 0 S      0   162     1   TS  59        ?    533        ? ?        0:00 pfexecd
 0 S      0  1738  1730   TS  59        ?    817        ? pts/ 1      0:00 bash
 0 S      1   852     1   TS  59        ?    851        ? ?        0:17 rpcbind
 0 S     17    43     1   TS  59        ?   1096        ? ?        0:01 netcfgd
 0 S     15    47     1   TS  59        ?    765        ? ?        0:00 dlmgmtd
 0 S      0    68     1   TS  59        ?    694        ? ?        0:01 in.mpath
 0 S      1  1220     1   FX  60        ?    682        ? ?        0:00 nfs4cbd
 0 S     16    89     1   TS  59        ?   1673        ? ?        0:02 nwamd
 0 S      0   146     1   TS  59        ?    629        ? ?        0:01 dhcpagen
 0 S      1   129     1   TS  59        ?   1843        ? ?        0:00 kcfd
 0 S      1  1215     1   FX  60        ?    738        ? ?        0:00 lockd
 0 S      0   829   828   TS  59        ?    968        ? ?        0:00 hald-run
```

```
 0 S      0   361     1   TS  59        ?   1081        ? ?         0:01 devfsadm
 0 S      0   879     1   TS  59        ?   1166        ? ?         0:01 inetd
 0 O 119764  1773   880   TS  59        ?    557          cons ole    0:00 ps
 0 S      0   844   829   TS  59        ?    996        ? ?         0:00 hald-add
 0 S      0   895   866   TS  59        ?    590        ? ?         0:00 ttymon
 0 S      0   840     1   TS  59        ?    495        ? ?         0:00 cron
 0 S      0   874     1   TS  59        ?    425        ? ?         0:00 utmpd
 0 S      0  1724   956   TS  59        ?   2215        ? ?         0:00 sshd
 0 S 119764   880     9   TS  59        ?    565        ? cons ole    0:00 csh
 0 S      0   210     1   TS  59        ?   1622        ? ?         0:00 sysevent
 0 S      0   279     1   TS  59        ?    472        ? ?         0:00 iscsid
 0 S      1  1221     1   TS  59        ?   1349        ? ?         0:00 nfsmapid
 1 S      0   374     0  SDC  99        ?      0        ? ?         0:00 zpool-us
 0 S      0  1207     1   TS  59        ?   1063        ? ?         0:00 rmvolmgr
 0 S      0   828     1   TS  59        ?   1776        ? ?         0:03 hald
 0 S      0   853   829   TS  59        ?    896        ? ?         0:02 hald-add
 0 S      0   373     1   TS  59        ?    985        ? ?         0:00 picld
 0 S      0   299     1   TS  59        ?    836        ? ?         0:00 dbus-dae
 0 S  12524  1730  1725   TS  59        ?    452        ? pts/ 1      0:00 csh
 0 S      0   370     1   TS  59        ?    574        ? ?         0:00 powerd
 0 S      0   264     1   FX  60        ?    637        ? ?         0:00 zonestat
 0 S      0   866     9   TS  59        ?    555        ? ?         0:00 sac
 0 S      0   851   829   TS  59        ?    998        ? ?         0:00 hald-add
 0 S  12524  1725  1724   TS  59        ?   2732        ? ?         0:00 sshd
 0 S      1  1211     1   TS  59        ?    783        ? ?         0:00 statd
 0 S      0  1046     1   TS  59        ?   1770        ? ?         0:13 intrd
 0 S      0   889     1   TS  59        ?   1063        ? ?         0:00 syslogd
 0 S      0  1209     1   TS  59        ?    792        ? ?         0:00 in.ndpd
 0 S      0  1188  1186   TS  59        ?    951        ? ?         0:15 automoun
 0 S      0  1172   829   TS  59        ?    725        ? ?         0:00 hald-add
 0 S      0  1186     1   TS  59        ?    692        ? ?         0:00 automoun
 0 S    101  1739  1738   TS  59        ?    817        ? pts/ 1      0:00 bash
 0 S      0  1199     1   TS  59        ?   1495        ? ?         0:02 sendmail
 0 S      0   956     1   TS  59        ?   1729        ? ?         0:00 sshd
 0 S     25  1192     1   TS  59        ?   1528        ? ?         0:00 sendmail
 0 S      0   934     1   TS  59        ?   6897        ? ?         0:14 fmd
 0 S      0  1131     1   TS  59        ?   1691        ? ?         0:07 nscd
 0 S      1  1181     1   TS  59        ?    699        ? ?         0:00 ypbind
```

Liste des tâches pour la gestion des informations des classes de processus

Utilisez les procédures suivantes pour gérer vos classes de processus

Tâche	Description	Pour obtenir des instructions
Définition de la priorité d'un processus	Démarrez un processus avec un niveau de priorité spécifique à l'aide de la commande `priocntl - e-c.`	"Définition de la priorité d'un processus (`priocntl`)" à la page 39

Tâche	Description	Pour obtenir des instructions
Modification des paramètres de planification d'un processus de partage du temps	Utilisez la commande `priocntl -s -m` pour modifier les paramètres de planification d'un processus de partage du temps.	“Modification des paramètres de planification d'un processus de partage du temps (priocntl)” à la page 40
Modifiez la classe d'un processus.	Utilisez la commande `priocntl -s -c` pour modifier la classe d'un processus.	“Modification de la classe d'un processus (priocntl)” à la page 41
Modification de la priorité d'un processus	Utilisez la commande `/usr/bin/nice` avec les options appropriées pour réduire ou augmenter la priorité d'un processus.	“Modification de la priorité d'un processus (nice)” à la page 43

Modification de la priorité de planification des processus (priocntl)

La priorité de planification d'un processus est la priorité affectée par le planificateur des processus, en fonction des stratégies de planification. La commande `dispadmin` répertorie les stratégies de planification par défaut. Pour plus d'informations, reportez-vous à la page de manuel `dispadmin`(1M).

Vous pouvez utiliser la commande `priocntl` pour affecter des processus à une classe de priorité et pour gérer les priorités de processus comme le montre la procédure suivante.

▼ Définition de la priorité d'un processus (priocntl)

1. **Prenez le rôle `root`.**

   Reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

2. **Démarrez un processus avec un niveau de priorité désigné.**

   ```
   # priocntl -e -c class -m user-limit -p PRI command-name
   ```

   -e — Exécution de la commande.

   -c *class* — Spécifie la classe dans laquelle vous souhaitez exécuter le processus. Les classes valides sont `TS` (partage du temps), `RT` (temps réel), IA (interactive), FSS (partage équitable) et FX (priorité fixe).

-m *user-limit*	Lorsque vous utilisez l'option -p avec cette option, la quantité maximale dont vous pouvez augmenter ou diminuer votre priorité est également indiquée.
-p *PRI*	Permet de spécifier la priorité relative de la classe RT pour un thread en temps réel. Pour un processus de partage du temps, l'option -p vous permet de spécifier la priorité utilisateur, comprise entre -60 à +60.
command-name	Indique le nom de la commande qui sera exécutée.

3. **Vérifiez l'état du processus.**

```
# ps -ecl | grep command-name
```

Exemple 2-4 Désignation d'une priorité de processus (priocntl)

L'exemple suivant montre comment lancer la commande find avec la priorité définie par l'utilisateur la plus élevée.

```
# priocntl -e -c TS -m 60 -p 60 find . -name core -print
# ps -ecl | grep find
```

▼ Modification des paramètres de planification d'un processus de partage du temps (priocntl)

1. **Prenez le rôle root.**

   Reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

2. **Modifiez les paramètres de planification d'un processus de partage de temps en cours.**

```
# priocntl -s -m user-limit [-p user-priority] -i ID type ID list
```

-s	Permet de définir la limite supérieure de la plage de priorité utilisateur et de modifier la priorité en cours.
-m *user-limit*	Lorsque vous utilisez l'option -p, indique le niveau maximum dont vous pouvez augmenter ou diminuer la priorité.
-p *user-priority*	Permet de définir une priorité.

`-i` *ID type ID list* — Utilise une combinaison des paramètres *xidtype* et *xidlist* pour identifier le ou les processus. Le paramètre *xidtype* spécifie le type d'ID, comme l'ID de processus ou l'ID utilisateur. Utilisez le paramètre *xidlist* pour identifier une liste des ID de processus ou des ID utilisateur.

3. **Vérifiez l'état du processus.**

```
# ps -ecl | grep ID list
```

Exemple 2-5 Modification des paramètres de planification d'un processus de partage du temps (`priocntl`)

L'exemple suivant illustre comment exécuter une commande avec une tranche de temps de 500 millisecondes, une priorité de 20 dans la classe RT et une priorité globale de 120.

```
# priocntl -e -c RT -m 500 -p 20 myprog
# ps -ecl | grep myprog
```

▼ Modification de la classe d'un processus (`priocntl`)

1. **(Facultatif) Prenez le rôle `root`.**

Remarque - Vous devez prendre le rôle root ou utiliser un shell en temps réel pour modifier un processus depuis ou vers un processus en temps réel. Si, en tant qu'utilisateur root, vous affectez un processus utilisateur à la classe en temps réel, l'utilisateur ne pourra pas modifier les paramètres de programmation en temps réel en utilisant la commande `priocntl -s`.

Reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

2. **Modifiez la classe d'un processus.**

```
# priocntl -s -c class -i ID type ID list
```

`-s` — Permet de définir la limite supérieure de la plage de priorité utilisateur et de modifier la priorité en cours.

`-c` *class* — Spécifie la classe, `TS` pour le partage du temps ou `RT pour le temps réel`, que vous affectez au processus.

`-i` *ID type ID list* — Utilise une combinaison des paramètres *xidtype* et *xidlist* pour identifier le ou les processus. Le processus *xidtype* spécifie le type d'ID, par

exemple l'ID de processus ou l'ID utilisateur. Utilisez le paramètre *xidlist* pour identifier une liste des ID de processus ou des ID utilisateur.

3. **Vérifiez l'état du processus.**

```
# ps -ecl | grep ID list
```

Exemple 2-6 Modification de la classe d'un processus (priocntl)

L'exemple suivant illustre comment modifier tous les processus qui appartiennent à l'utilisateur 15249 pour les affecter aux processus en temps réel.

```
# priocntl -s -c RT -i uid 15249
# ps -ecl | grep 15249
```

Modification de la priorité d'un processus de partage du temps (nice)

La commande nice est prise en charge uniquement pour assurer la compatibilité avec les versions antérieures. La commande priocntl accroît la flexibilité dans la gestion des processus.

La priorité d'un processus est définie par les stratégies de sa classe de programmation et par son nombre *nice*. Chaque processus de partage du temps comporte une priorité globale. La priorité globale est calculée en ajoutant la priorité utilisateur, qui peut être influencée par les commandes nice et priocntl, et la priorité calculée par le système.

Le numéro de priorité d'exécution d'un processus est attribué par le système d'exploitation. Le numéro de priorité est déterminé par plusieurs facteurs, notamment la classe de programmation du processus, le temps CPU utilisé et, dans le cas d'un processus de partage du temps, son nombre nice.

Chaque processus de partage du temps commence avec un nombre nice par défaut, qu'il hérite de son processus parent. Le nombre nice est indiqué dans la colonne NI du rapport ps.

Un utilisateur peut diminuer la priorité d'un processus en augmentant sa priorité utilisateur. Cependant, seul le superutilisateur peut réduire un nombre nice pour augmenter la priorité d'un processus. Cette restriction empêche les utilisateurs d'augmenter les priorités de leurs propres processus, ce qui monopolise une plus grande part de la CPU.

Les nombres nice sont compris entre 0 et +39, où 0 représente la priorité la plus élevée. La valeur par défaut nice de chaque processus de partage du temps est de 20. Deux versions de la commande sont disponibles : la version standard, /usr/bin/nice et la commande intégrée au shell C.

Modification de la priorité d'un processus (`nice`)

En tant qu'utilisateur, vous pouvez diminuer la priorité d'un processus. Connectez-vous en tant qu'administrateur pour augmenter ou diminuer la priorité d'un processus.

- En tant qu'utilisateur, réduisez la priorité d'une commande en augmentant le nombre `nice`.

  La commande `nice` suivante exécute *command-name* avec une priorité inférieure en augmentant la valeur du nombre nice de 5 unités.

  ```
  $ /usr/bin/nice -5 command-name
  ```

  Dans cette commande, le signe moins indique que ce qui suit est une option. Cette commande peut également être définie comme suit :

  ```
  $ /usr/bin/nice -n 5 command-name
  ```

  La commande `nice` suivante réduit la priorité de *command-name* en augmentant le nombre nice de l'incrément par défaut de 10 unités, mais pas au-delà de la valeur maximale de 39.

  ```
  $ /usr/bin/nice command-name
  ```

- En tant que superutilisateur, augmentez ou diminuez la priorité d'une commande en modifiant le nombre `nice`.

  La commande suivante`nice` augmente la priorité de *command-name* en réduisant le nombre nice de 10 unités. Il n'est pas abaissé sous la valeur minimale de 0.

  ```
  # /usr/bin/nice --10 command-name
  ```

  Dans la commande ci-dessus, le premier signe moins indique que ce qui suit est une option. Le deuxième signe moins indique un nombre négatif.

  La commande suivante`nice` réduit la priorité de *command-name* en augmentant le nombre nice de 5 unités. Il ne dépasse pas la valeur maximale de 39.

  ```
  # /usr/bin/nice -5 command-name
  ```

Pour plus d'informations, reportez-vous à la page de manuel `nice`(1).

Résolution des problèmes liés aux processus système

Des problèmes de processus qui pourraient se poser à vous sont, entre autres :

- Recherchez plusieurs travaux identiques détenus par le même utilisateur. Ce problème peut se produire en raison d'un script qui démarre de nombreux travaux de fond sans attendre la fin de l'un d'entre eux.

- Cherchez un processus qui a accumulé une vaste quantité de temps de l'unité centrale. Vous pouvez identifier ce problème en vérifiant le champ `TIME` dans la sortie `ps`. Cette valeur peut indiquer que le processus se trouve dans une boucle infinie.
- Cherchez un processus en cours d'exécution avec une priorité trop élevée. Utilisez la commande `ps -c` pour vérifier le champ `CLS`, qui affiche la classe de programmation de chaque processus. Un processus en cours d'exécution en temps réel (`RT`) peut monopoliser le CPU. Ou recherchez un processus de partage du temps (`TS`) avec un nombre `nice` élevé. Un administrateur peut avoir augmenté la priorité d'un processus. L'administrateur système peut diminuer la priorité à l'aide de la commande `nice`.
- Cherchez un processus hors de contrôle qui augmente progressivement le temps de l'unité centrale utilisé. Vous pouvez identifier ce problème en examinant l'heure de démarrage du processus (`STIME`) et en observant le cumul de temps CPU (`TIME`) pendant un certain temps.

♦ ♦ ♦ CHAPITRE 3

Surveillance des performances système

L'obtention de bonnes performances à partir d'un ordinateur ou d'un réseau est une partie importante de l'administration du système. Ce chapitre décrit certains facteurs qui contribuent à la gestion des performances des systèmes informatiques sous votre responsabilité. Ce chapitre décrit les procédures de surveillance des performances du système à l'aide des commandes `vmstat`, `iostat`, `df` et `sar`.

Ce chapitre comprend les sections suivantes :

Différentes sources d'informations à propos de la surveillance des performances système

Tâches relatives aux performances du système	Pour en savoir plus
Gestion des processus	Chapitre 2, Gestion des processus système
Contrôle des performances du système	Chapitre 3, Surveillance des performances système
Modification des paramètres réglables	“ Manuel de référence des paramètres réglables d’Oracle Solaris 11.2 ”
Gestion des tâches relatives aux performances du système	Chapitre 2, “ A propos des projets et tâches ” du manuel “ Administration de la gestion des ressources dans Oracle Solaris 11.2 ”
Gestion des processus avec les planificateurs FX et FS	Chapitre 8, “ A propos de l’ordonnanceur de partage équitable ” du manuel “ Administration de la gestion des ressources dans Oracle Solaris 11.2 ”

Gestion des performances à l'aide d'Oracle Enterprise Manager Ops Center

Si vous avez besoin de gérer, d'analyser et d'améliorer la performance des systèmes d'exploitation physiques et virtuels, serveurs virtuels et périphériques de stockage au sein d'un centre de données plutôt que de surveiller les performances dans un système spécifique, vous pouvez utiliser les solutions complètes de gestion système dans Oracle Enterprise Manager Ops Center.

La fonctionnalité de surveillance d'Enterprise Manager Ops Center fournit des informations détaillées sur les systèmes d'exploitation et zones de votre centre de données surveillés. Vous pouvez utiliser ces informations pour évaluer les performances, identifier les problèmes et effectuer un réglage. Des analyses sont disponibles pour le système d'exploitation Oracle Solaris, Linux et les technologies de virtualisation SE, y compris Oracle Solaris Zones, Oracle VM Server for SPARC et Oracle VM Server for x86 guests.

Pour plus d'informations, reportez-vous à la http://www.oracle.com/pls/topic/lookup?ctx=oc122.

A propos des ressources système ayant une incidence sur les performances du système

Les performances d'un système informatique dépendent de la façon dont le système utilise et alloue ses ressources. Surveillez régulièrement les performances du système afin de connaître son comportement dans des conditions normales d'utilisation. Vous devez avoir une bonne idée de ce qu'il faut attendre et être capable de reconnaître un problème lorsqu'il se produit.

Les ressources système qui affectent les performances sont les suivantes :

Unité de calcul centrale (CPU)	La CPU traite les instructions en les extrayant de la mémoire de l'ordinateur et en les exécutant.
Périphériques d'entrée/sortie (E/S)	Les périphériques d'E/S transfèrent les informations à l'intérieur et à l'extérieur de l'ordinateur. Il peut s'agir d'un terminal et d'un clavier, d'une unité de disque ou d'une imprimante.
Mémoire	La mémoire physique (ou principale) représente la quantité de mémoire vive (RAM) du système.

Le Chapitre 3, Surveillance des performances système décrit les outils qui affichent les statistiques sur l'activité et les performances du système.

A propos des processus et performances du système

Voici quelques termes liés aux processus :

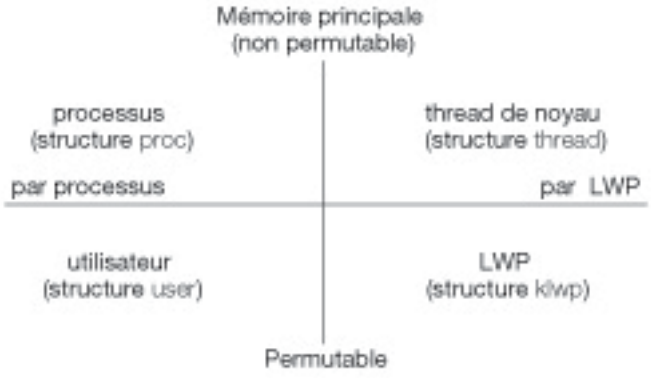
Processus	N'importe quelle activité ou travail du système. Chaque fois que vous initialisez un système, exécutez une commande ou démarrez une application, le système active un ou plusieurs processus.
Processus léger (LWP)	CPU virtuelle ou ressource d'exécution. Les LWP sont planifiés par le noyau afin d'exploiter les ressources CPU disponibles en fonction de leur classe de programmation et de leur priorité. Un LWP contient des informations permutables et thread de noyau qui contient des informations à conserver en permanence en mémoire.
Thread d'application	Série d'instructions dotée d'une pile séparée qui peut s'exécuter indépendamment dans l'espace d'adressage d'un utilisateur. Les threads d'application peuvent faire l'objet d'un multiplexage au-dessus des LWP.

Un processus peut être composé de plusieurs LWP et threads d'application. Le noyau planifie une structure de noyau-thread, qui représente l'entité de planification dans l'environnement Oracle Solaris. Voici différentes structures de processus :

`proc`	Contient des informations qui concernent l'ensemble du processus et doivent se trouver en permanence dans la mémoire principale.
`kthread`	Contient des informations qui concernent un LWP et doivent se trouver en permanence dans la mémoire principale.
`user`	Contient les informations permutables "par processus".
`klwp`	Contient les informations permutables "par processus LWP".

La figure ci-dessous illustre les relations entre ces structures de processus.

FIGURE 3-1 Relations entre les structures de processus



La plupart des ressources de processus sont disponibles pour tous les threads du processus. Presque toute la mémoire virtuelle du processus est partagée. Un changement de données partagées par un thread est disponible pour les autres threads du processus.

A propos de la surveillance des performances du système

Lorsque votre ordinateur fonctionne, les compteurs du système d'exploitation sont incrémentés afin de suivre les différentes activités du système.

Les activités du système qui font l'objet d'un suivi sont les suivantes :

- Utilisation de l'unité de calcul centrale (CPU)
- Utilisation de la mémoire tampon
- Activité d'entrée/sortie (E/S) des disques et bandes
- Activité des périphériques terminaux
- Activité d'appel système
- Changement de contexte
- Accès aux fichiers
- Activité de la file d'attente
- Tables du noyau

- Communication interprocessus
- Pageur
- Mémoire libre et espace de swap
- Allocation de mémoire du noyau (KMA)

Outils de surveillance

Oracle Solaris fournit plusieurs outils qui facilitent le suivi des performances du système.

TABLEAU 3-1 Outils de surveillance des performances

Commande	Description	Pour en savoir plus
Commandes `cpustat` et `cputrack`	Surveille les performances d'un système ou d'un processus à l'aide des compteurs de performances CPU.	`cpustat`(1M) et `cputrack`(1)
Commandes `netstat` et `nfsstat`	Affiche des informations sur les performances réseau.	`netstat`(1M) et `nfsstat`(1M)
Commandes `ps` et `prstat`	Affiche des informations sur les processus actifs.	Chapitre 2, Gestion des processus système
Commandes `sar` et `sadc`	Collecte des données et les consigne dans des rapports sur l'activité du système.	Chapitre 3, Surveillance des performances système
Commande `swap`	Affiche des informations sur l'espace de swap disponible sur votre système.	Chapitre 3, “ Extension de l’espace de swap ” du manuel “ Gestion des systèmes de fichiers dans Oracle Solaris 11.2 ”
Commandes `vmstat` et `iostat`	Récapitule les données d'activité du système, telles que les données statistiques de mémoire virtuelle, l'utilisation du disque et l'activité CPU.	Chapitre 3, Surveillance des performances système
Commandes `kstat` et `mpstat`	Examine les statistiques disponibles du noyau, ou `kstats`, sur le système et génère un rapport sur les statistiques qui correspondent aux critères spécifiés sur la ligne de commande. La commande `mpstat` génère un rapport sur les statistiques du processus sous forme de tableau.	Pages de manuel `kstat`(1M) et `mpstat`(1M).

Affichage des informations sur les performances du système

Cette section décrit les tâches de contrôle et l'affichage des informations sur les performances du système.

Affichage des statistiques de mémoire virtuelle

Vous pouvez utiliser la commande `vmstat` pour générer des rapports sur les statistiques de mémoire virtuelle et sur les informations sur les événements système telles que la charge CPU, la pagination, le nombre de changements de contexte, les interruptions de périphérique et les appels système. La commande `vmstat` permet également d'afficher les statistiques sur la permutation, la purge du cache et les interruptions.

TABLEAU 3-2 Sortie de la commande `vmstat`

catégorie	Nom du champ	Description
`procs`		
	`r`	Nombre de threads de noyau dans la file d'attente de répartition
	`b`	Nombre de threads de noyau bloqués qui sont en attente de ressources
	`w`	Nombre de LWP extraits du swap qui attendent la fin du traitement des ressources
`memory`		Indique l'utilisation de la mémoire réelle et virtuelle
	`swap`	Espace de swap disponible
	`free`	Taille de la liste d'espaces libres
`page`		Indique les défauts de page et l'activité de pagination, en unités par seconde
	`re`	Pages récupérées
	`mf`	Erreurs mineures et majeures
	`pi`	Kilo-octets chargés
	`po`	Kilo-octets renvoyés
	`fr`	Kilo-octets libérés
	`de`	Mémoire anticipée requise par les processus récemment introduits dans le swap
	`sr`	Pages analysées par le démon `page` qui ne sont pas en cours d'utilisation. Si `sr` est différent de zéro, le démon `page` a été exécuté.
`disk`		Indique le nombre d'opérations sur disque par seconde, en affichant les données d'un maximum de quatre disques
`faults`		Indique le taux d'interruption/déroutement par seconde
	`in`	Interruptions par seconde
	`sy`	Appels système par seconde
	`cs`	Taux de changement de contexte CPU
`cpu`		Indique l'utilisation du temps CPU
	`us`	USER TIME
	`sy`	Heure système
	`id`	Temps d'inactivité

Pour une description plus détaillée de cette commande, reportez-vous à la page de manuel vmstat(1M).

Affichage des statistiques de mémoire virtuelle (vmstat)

Pour afficher les statistiques de mémoire virtuelle, utilisez la commande vmstat avec un intervalle de temps (en secondes).

```
$ vmstat n
```

où *n* représente l'intervalle de création des rapports (en secondes).

L'exemple suivant illustre l'affichage vmstat des statistiques recueillies à des intervalles de cinq secondes :

```
$ 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  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
```

Affichage des informations sur les événements système (vmstat -s)

Exécutez la commande vmstat -s pour afficher le nombre d'événements système survenus depuis la dernière initialisation du système.

```
$ vmstat -s
        0 swap ins
        0 swap outs
        0 pages swapped in
        0 pages swapped out
   522586 total address trans. faults taken
    17006 page ins
       25 page outs
    23361 pages paged in
       28 pages paged out
    45594 total reclaims
```

```
   45592 reclaims from free list
       0 micro (hat) faults
  522586 minor (as) faults
   16189 major faults
   98241 copy-on-write faults
  137280 zero fill page faults
   45052 pages examined by the clock daemon
       0 revolutions of the clock hand
      26 pages freed by the clock daemon
    2857 forks
      78 vforks
    1647 execs
34673885 cpu context switches
65943468 device interrupts
  711250 traps
63957605 system calls
 3523925 total name lookups (cache hits 99%)
   92590 user   cpu
   65952 system cpu
16085832 idle   cpu
    7450 wait   cpu
```

Affichage des statistiques de swap (`vmstat -S`)

Exécutez `vmstat -S` pour afficher les statistiques de permutation.

```
$ vmstat -S
kthr      memory            page            disk          faults      cpu
 r b w   swap  free  si  so pi po fr de sr dd f0 s1 --   in   sy   cs us sy id
 0 0 0 862608 364792  0   0  1  0  0  0  0  0  0  0  0  406  394  213  1  0 99
```

Les champs des statistiques de permutation sont décrits dans la liste suivante. Pourune description des autres champs, reportez-vous au Tableau 3-2, “Sortie de la commande `vmstat`”.

`si` — Nombre moyen de processus légers (LWP) qui sont introduits dans le swap par seconde

`so` — Nombre de processus complets qui sont extraits du swap

Remarque - La commande `vmstat` tronque la sortie des champs `si` et `so`. Utilisez la commande `sar` pour afficher une comptabilisation plus précise des statistiques de swap.

Affichage des interruptions par périphérique (`vmstat -i`)

Exécutez la commande `vmstat -i` pour afficher le nombre d'interruptions par périphérique.

L'exemple suivant illustre la sortie de la commande `vmstat -i`.

```
$ 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
```

Affichage des informations sur l'utilisation des disques

Utilisez la commande `iostat` pour générer des rapports statistiques sur l'entrée et la sortie des disques et fournir des mesures du débit, de l'utilisation, des longueurs de file d'attente, des taux de transaction et de la durée de service. Pour une description plus détaillée de cette commande, reportez-vous à la page de manuel `iostat`(1M).

Affichage des informations sur l'utilisation des disques (`iostat`)

Vous pouvez afficher les informations sur l'utilisation des disques en utilisant la commande `iostat` avec un intervalle de temps (en secondes).

```
$ 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
```

La première ligne de la sortie présente les statistiques depuis la dernière initialisation du système. Chaque ligne suivante présente les statistiques de l'intervalle. La valeur par défaut permet d'afficher les statistiques du terminal (`tty`), des disques (`fd` et `sd`), et de la CPU (`cpu`).

L'exemple suivant présente les statistiques de disque collectées toutes les cinq secondes.

```
$ 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
```

Le tableau ci-dessous décrit les champs contenus dans la sortie de la commande `iostat`*n*.

Type de dispositif	Nom du champ	Description
Terminal		
	`tin`	Nombre de caractères dans la file d'attente d'entrée du terminal
	`tout`	Nombre de caractères de la file d'attente de sortie du terminal
Disk (Disque)		
	`bps`	Blocs par seconde
	`tps`	Transactions par seconde
	`serv`	Temps de service moyen (en millisecondes)
UC		
	`us`	En mode utilisateur
	`sy`	En mode système
	`wt`	En attente d'E/S
	`id`	Inactif

Affichage des statistiques de disque étendues (`iostat -xtc`)

Exécutez la commande `iostat &minus;xt` pour afficher les statistiques de disque étendues.

```
$ iostat &minus;xt
device     r/s    w/s   kr/s   kw/s wait actv wsvc_t asvc_t  %w  %b  tin tout
blkdev0    0.0    0.0    0.0    0.0  0.0  0.0    0.0    0.0   0   0    0    1
sd0        0.1   19.3    1.4   92.4  0.0  0.0    0.2    1.6   0   1
sd1        0.0    0.0    0.0    0.0  0.0  0.0    0.0    0.0   0   0
nfs9       0.0    0.0    0.0    0.0  0.0  0.0    0.0    1.0   0   0
nfs10      0.0    0.0    0.0    0.0  0.0  0.0    0.0    7.6   0   0
nfs11      0.0    0.0    0.0    0.0  0.0  0.0    0.0   15.6   0   0
nfs12      0.3    0.0    1.9    0.0  0.0  0.0    0.0   30.5   0   1
```

La commande `iostat &minus;xt` affiche une ligne de sortie pour chaque disque. Les champs sont les suivants :

`r/s`	Lectures par seconde
`w/s`	Ecritures par seconde
`kr/s`	Kilo-octets lus par seconde
`kw/s`	Kilo-octets écrits par seconde
`wait`	Nombre moyen de transactions en attente de service (longueur de la file d'attente)
`actv`	Nombre moyen de transactions en cours de service
`svc_t`	Temps de service moyen (en millisecondes)
`%w`	Pourcentage de temps pendant lequel la file d'attente n'est pas vide
`%b`	Pourcentage de temps pendant lequel le disque est occupé

Affichage des statistiques de l'espace disque (`df`)

Utilisez la commande `df` pour afficher l'espace disponible sur chaque disque monté. L'espace disque *utilisable* indiqué par `df` ne reflète que 90 % de la capacité totale, puisque les statistiques des rapports prennent en considération 10 % au-dessus de l'espace total disponible. Cette *marge* reste normalement vide pour améliorer les performances.

Le pourcentage d'espace disque réellement indiqué par la commande `df` est l'espace utilisé divisé par l'espace utilisable.

Si le système de fichiers dépasse 90 % de la capacité, vous pouvez transférer des fichiers vers un disque encore disponible à l'aide de la commande `cp`. Vous pouvez également transférer des fichiers sur une bande à l'aide des commandes `tar` ou `cpio`. Vous pouvez aussi supprimer les fichiers.

Pour une description plus détaillée de cette commande, reportez-vous à la page de manuel `df`(1M).

Affichage des informations sur l'espace disque (`df -k`)

Utilisez la commande `df -k` pour afficher les informations sur l'espace disque (en kilo-octets).

```
$ df -k
Filesystem            kbytes    used   avail capacity  Mounted on
/dev/dsk/c0t3d0s0     192807   40231  133296    24%    /
```

EXEMPLE 3-1 Affichage des informations sur le système de fichiers

L'exemple suivant montre la sortie de la commande `df -k` pour un système SPARC.

```
$ df -k
Filesystem           1024-blocks        Used   Available Capacity  Mounted on
rpool/ROOT/solaris-161   191987712     6004395   140577816     5%    /
/devices                       0           0           0     0%    /devices
/dev                           0           0           0     0%    /dev
ctfs                           0           0           0     0%    /system/contract
proc                           0           0           0     0%    /proc
mnttab                         0           0           0     0%    /etc/mnttab
swap                     4184236         496     4183740     1%    /system/volatile
objfs                          0           0           0     0%    /system/object
sharefs                        0           0           0     0%    /etc/dfs/sharetab
/usr/lib/libc/libc_hwcap1.so.1   146582211     6004395   140577816     5%    /lib/libc.so.1
fd                             0           0           0     0%    /dev/fd
swap                     4183784          60     4183724     1%    /tmp
rpool/export           191987712          35   140577816     1%    /export
rpool/export/home      191987712          32   140577816     1%    /export/home
rpool/export/home/123   191987712    13108813   140577816     9%    /export/home/123
rpool/export/repo      191987712    11187204   140577816     8%    /export/repo
rpool/export/repo2010_11   191987712          31   140577816     1%    /export/repo2010_11
rpool                  191987712     5238974   140577816     4%    /rpool
/export/home/123      153686630    13108813   140577816     9%    /home/123
```

Les champs de sortie de la commande `df -k` sont les suivants :

`1024-blocks` Taille totale de l'espace utilisable dans le système de fichiers

`Used` Quantité d'espace utilisé

`Available` Quantité d'espace disque disponible pour une utilisation

`Capacity` Quantité d'espace utilisé, sous forme de pourcentage de la capacité totale

`Mounted on` Point de montage

EXEMPLE 3-2 Affichage des informations du système de fichiers à l'aide de la commande `df` utilisée sans option

Lorsque la commande `df` est utilisée sans opérande ou option, elle fournit des informations sur tous les systèmes de fichiers montés, comme indiqué dans l'exemple suivant :

```
$ df
/                  (rpool/ROOT/solaris):100715496 blocks 100715496 files
```

```
/devices           (/devices          ):       0 blocks        0 files
/dev               (/dev              ):       0 blocks        0 files
/system/contract   (ctfs              ):       0 blocks 2147483601 files
/proc              (proc              ):       0 blocks    29946 files
/etc/mnttab        (mnttab            ):       0 blocks        0 files
/system/volatile   (swap              ):42257568 blocks  2276112 files
/system/object     (objfs             ):       0 blocks 2147483441 files
/etc/dfs/sharetab  (sharefs           ):       0 blocks 2147483646 files
/dev/fd            (fd                ):       0 blocks        0 files
/tmp               (swap              ):42257568 blocks  2276112 files
/export            (rpool/export      ):100715496 blocks 100715496 files
/export/home       (rpool/export/home ):100715496 blocks 100715496 files
/export/home/admin (rpool/export/home/admin):100715496 blocks 100715496 files
/rpool             (rpool             ):100715496 blocks 100715496 files
/export/repo2010_11(rpool/export/repo2010_11):281155639 blocks 281155639 files
/rpool             (rpool             ):281155639 blocks 281155639 files
```

Surveillance des activités du système

Cette section décrit les activités de surveillance des activités du système.

Surveillance des activités du système (sar)

Utilisez la commande `sar` pour effectuer les tâches suivantes :

- Organiser et visualiser les données sur l'activité du système.
- Accéder aux données de l'activité système sur demande spéciale.
- Générer des rapports automatiques pour mesurer et contrôler les performances du système, ainsi que des rapports sur demande spéciale afin d'identifier les problèmes de performance. Pour plus d'informations sur la configuration de la commande `sar` afin qu'elle s'exécute sur votre système, ainsi qu'une description de ces outils, reportez-vous à la section “Collecte automatique des données sur l'activité du système (sar)” à la page 74.

Pour une description plus détaillée de cette commande, reportez-vous à la page de manuel `sar`(1).

Vérification de l'accès aux fichiers (sar -a)

Affichez les statistiques des opérations d'accès aux fichiers à l'aide de la commande `sar -a`.

```
$ sar -a

SunOS t2k-brm-24 5.10 Generic_144500-10 sun4v    ...
```

```
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
```

Les routines suivantes du système d'exploitation font l'objet des rapports de la commande `sar -a` :

`iget/s` Nombre de demandes effectuées pour les inodes qui ne se trouvaient pas dans le cache de recherche de nom de répertoire (DNLC).

`namei/s` Nombre de recherches de chemin d'accès au système de fichiers par seconde. Si `namei` ne trouve pas un nom de répertoire dans le DNLC, il appelle `iget` afin d'obtenir l'inode d'un fichier ou d'un répertoire. Par conséquent, la plupart des champs `igets` sont le résultat d'échecs de DNLC.

`dirbk/s` Nombre de lectures de bloc de répertoire par seconde.

Plus les valeurs indiquées des routines du système d'exploitation sont élevées, plus le noyau passe de temps à accéder aux fichiers utilisateurs. La durée reflète l'intensité avec laquelle les programmes et applications utilisent les systèmes de fichiers. L'option `-a` permet de visualiser le degré de dépendance aux disques d'une application.

Vérification de l'activité du tampon (`sar -b`)

Affichez les statistiques sur l'activité du tampon avec la commande `sar -b`.

Le tampon sert à mettre en cache les métadonnées. Les métadonnées comprennent les inodes, les blocs de groupes de cylindres et les blocs indirects.

```
$ 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
```

Le tableau suivant décrit les activités du tampon affichées par l'option `-b`.

Nom du champ	Description
`bread/s`	Nombre moyen de lectures par seconde qui sont soumises au cache du tampon à partir du disque
`lread/s`	Nombre moyen de lectures logiques par seconde à partir du cache du tampon
`%rcache`	Fraction des lectures logiques qui se trouvent dans le cache du tampon (100 % moins le rapport `bread/s` sur `lread/s`)
`bwrit/s`	Nombre moyen de blocs physiques (512 octets) qui sont écrits à partir du cache du tampon sur le disque, par seconde
`lwrit/s`	Nombre moyen d'écritures logiques sur le cache du tampon, par seconde
`%wcache`	Fraction des écritures logiques qui se trouvent dans le cache du tampon (100 % moins le rapport `bwrit/s` sur `lwrit/s`)
`pread/s`	Nombre moyen de lectures physiques, par seconde, qui utilisent les interfaces de périphérique de caractère
`pwrit/s`	Nombre moyen de demandes d'écriture physique, par seconde, qui utilisent les interfaces de périphérique de caractère

Les entrées les plus importantes sont les rapports de succès de `%rcache` et `%wcache`. Ces entrées mesurent l'efficacité de la mise en mémoire tampon du système. Si `%rcache` est inférieur à 90 % ou si `%wcache` est inférieur à 65 %, il est possible d'améliorer les performances en augmentant l'espace du tampon.

EXEMPLE 3-3 Vérification de l'activité du tampon (`sar -b`)

L'exemple suivant de la sortie de la commande `sar -b` montre que les tampons `%rcache` et `%wcache` ne sont pas à l'origine des ralentissements. Toutes les données se trouvent dans les limites acceptables.

```
$ sar -b

SunOS t2k-brm-24 5.10 Generic_144500-10 sun4v    ...

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
```

Vérification des statistiques d'appel système (`sar -c`)

Affichez les statistiques d'appel système en utilisant la commande `sar -c`.

```
$ 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
```

La liste suivante décrit les catégories d'appel système signalées par l'option `-c`. En règle générale, les opérations de lecture et d'écriture représentent environ la moitié du nombre total d'appels système. Cependant, le pourcentage varie fortement en fonction des activités effectuées par le système.

`scall/s` Nombre de tous les types d'appels système par seconde, soit généralement environ 30 par seconde sur un système avec quatre à six utilisateurs.

`sread/s` Nombre d'appels système `read` par seconde.

`swrit/s` Nombre d'appels système `write` par seconde.

`fork/s` Nombre d'appels système `fork` par seconde, soit généralement environ 0,5 par seconde sur un système avec quatre à six utilisateurs. Ce nombre augmente si les scripts shell sont en cours d'exécution.

`exec/s` Nombre d'appels système `exec` par seconde. Si `exec/s` divisé par `fork/s` est supérieur à 3, recherchez les variables `PATH` inefficaces.

`rchar/s` Nombre de caractères (octets) transférés par les appels système `read` par seconde.

`wchar/s` Nombre de caractères (octets) transférés par les appels système `write` par seconde.

EXEMPLE 3-4 Vérification des statistiques d'appel système (`sar -c`)

L'exemple suivant illustre la sortie de la commande `sar -c`.

```
$ sar -c

SunOS balmy 5.10 Generic_144500-10 sun4v ...
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
```

Vérification de l'activité du disque (`sar -d`)

Affichez les statistiques sur l'activité du disque avec la commande `sar -d`.

```
$ sar -d

00:00:00   device        %busy   avque   r+w/s  blks/s  avwait  avserv
```

La liste suivante décrit les activités du périphérique de disque qui sont signalées par l'option `-d`.

`device` Nom du périphérique de disque surveillé.

`%busy` Durée pendant laquelle le périphérique a été occupé à traiter une demande de transfert.

`avque` Nombre moyen de demandes pendant la période où le périphérique était occupé à traiter une demande de transfert.

`r+w/s` Nombre de transferts de lecture et d'écriture vers le périphérique, par seconde.

`blks/s` Nombre de blocs de 512 octets transférés vers le périphérique, par seconde.

`avwait` Durée moyenne (en millisecondes) pendant laquelle les demandes de transfert restent dans la file d'attente. Cette durée est mesurée uniquement lorsque la file d'attente est occupée.

`avserv` Durée moyenne (en millisecondes) requise par le périphérique pour terminer une demande de transfert. Pour les disques, cette valeur

comprend les temps de recherche, de latence de rotation et de transfert des données.

EXEMPLE 3-5 Vérification de l'activité du disque

Cet exemple abrégé illustre la sortie de la commande `sar -d`.

```
$ sar -d

SunOS balmy 5.10 Generic_144500-10 sun4v    ...

12:36:32   device        %busy   avque   r+w/s  blks/s  avwait  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
```

Notez que la longueur des files d'attente et le temps d'attente sont mesurés lorsqu'une demande se trouve dans la file d'attente. Si la valeur `%busy` est petite, la longueur des files d'attente et des délais de service représente probablement les efforts périodiques du système pour garantir l'écriture rapide des blocs modifiés sur le disque.

Vérification du renvoi de page et de la mémoire (`sar -g`)

Utilisez la commande `sar -g` pour afficher les moyennes des activités de libération de mémoire et de renvoi de page.

```
$ 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
```

La sortie affichée par la commande `sar -g` permet de savoir si un ajout de mémoire est nécessaire. Utilisez la commande `ps -elf` pour afficher le nombre de cycles utilisés par le

démon page. Un nombre élevé de cycles, combiné avec des valeurs élevées pour les champs pgfree/s et pgscan/s, indique une insuffisance de mémoire.

La commande sar -g indique également si les inodes sont recyclées trop rapidement et entraînent une perte de pages réutilisables.

La liste suivante décrit la sortie de l'option -g.

pgout/s — Nombre de demandes de renvoi de page par seconde.

ppgout/s — Nombre réel de pages renvoyées, par seconde. Une seule demande de renvoi de page peut impliquer le renvoi de plusieurs pages.

pgfree/s — Nombre de pages placées sur la liste d'espaces libres, par seconde.

pgscan/s — Nombre de pages analysées par le démon page, par seconde. Si cette valeur est élevée, le démon page consacre beaucoup de temps à chercher de la mémoire libre. Cette situation implique la nécessité d'ajouter de la mémoire.

%ufs_ipf — Le pourcentage d'inodes ufs déduit de la liste d'espaces libres par iget associés à des pages réutilisables. Ces pages sont vidées et ne peuvent pas être récupérées par les processus. Par conséquent, ce champ représente le pourcentage de igets avec des pages vides. Une valeur élevée indique que la liste libre d'inodes est liée à une page et que le nombre d'inodes ufs peut avoir besoin d'être augmenté.

EXEMPLE 3-6 Vérification du renvoi de page et de la mémoire (sar -g)

L'exemple suivant illustre la sortie de la commande sar -g.

```
$ sar -g

SunOS balmy 5.10 Generic_144500-10 sun4v    ...

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
```

Vérification de l'allocation de mémoire du noyau

L'allocation de mémoire du noyau (KMA) permet à un sous-système du noyau d'allouer et de libérer de la mémoire, en fonction des besoins.

Au lieu d'allouer de manière statique la quantité maximale de mémoire possiblement requise sous la charge de pointe, la KMA divise les demandes de mémoire en trois catégories :

- Petite taille (moins de 256 octets)
- Grande taille (512 octets de 4 Ko)
- Surdimensionnée (supérieure à 4 Ko)

La KMA conserve deux pools de mémoire pour satisfaire les demandes de petite et grande taille. L'allocation de mémoire satisfait les demandes surdimensionnées à partir du programme d'allocation de pages système.

La commande `sar -k` s'avère utile si vous vérifiez un système qui sert à écrire des pilotes ou des STREAMS qui utilisent les ressources KMA. Tout pilote ou module qui utilise des ressources KMA, mais ne retourne pas spécifiquement les ressources avant d'être arrêté, peut créer une fuite de mémoire. Une fuite de mémoire entraîne l'augmentation de la quantité de mémoire allouée par KMA au fil du temps. Par conséquent, si les champs `alloc` de la commande `sar -k` augmentent progressivement au fil du temps, il est possible qu'il y ait une fuite de mémoire. Les échecs de demande indiquent également une fuite de mémoire. Si un problème de ce type se produit, c'est probablement à cause d'une fuite de mémoire que KMA ne peut pas réserver ni allouer la mémoire.

S'il apparaît qu'une fuite de mémoire s'est produite, vous devez vérifier tous les pilotes ou STREAMS susceptibles d'avoir demandé de la mémoire à KMA et qui ne l'ont pas retournée.

Vérification de l'allocation de mémoire du noyau (`sar -k`)

Utilisez la commande `sar -k` pour générer des rapports sur les activités du programme d'allocation de mémoire du noyau (KMA).

```
$ 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
```

La liste suivante décrit la sortie de l'option `-k`.

`sml_mem` Volume de mémoire (en octets) disponible pour KMA dans le pool des demandes de faibles quantités de mémoire. Dans ce pool, une petite demande est inférieure à 256 octets.

`alloc` Volume de mémoire (en octets) que KMA a alloué à des demandes de faibles quantités de mémoire en puisant dans son pool.

`fail` Nombre de demandes de faibles quantités de mémoire qui ont échoué.

`lg_mem` Volume de mémoire (en octets) disponible pour KMA dans le pool des demandes de grandes quantités de mémoire. Dans ce pool, une grande demande va de 512 octets à 4 Ko.

`alloc` Volume de mémoire (en octets) que KMA a alloué à des demandes de grandes quantités de mémoire en puisant dans son pool.

`fail` Nombre de demandes de grandes quantités de mémoire qui ont échoué.

`ovsz_alloc` Volume de mémoire alloué à des demandes surdimensionnées, c'est-à-dire supérieures à 4 Ko. Ces demandes sont satisfaites par le programme d'allocation de page. Il n'existe donc aucun pool.

`fail` Nombre de demandes de quantités surdimensionnées de mémoire qui ont échoué.

EXEMPLE 3-7 Vérification de l'allocation de mémoire du noyau (`sar -k`)

L'exemple suivant illustre un exemple abrégé de sortie de la commande `sar -k`.

```
$ sar -k

SunOS balmy 5.10 Generic_144500-10 sun4v    ...
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
```

Vérification de la communication interprocessus (`sar -m`)

Utilisez la commande `sar -m` pour signaler les activités de communication interprocessus.

```
$ sar -m
00:00:00   msg/s  sema/s
01:00:00    0.00    0.00
```

Ces chiffres sont généralement nuls (0,00), sauf si vous exécutez des applications qui utilisent des messages ou des sémaphores.

La sortie de l'option `-m` est la suivante :

`msg/s` Nombre d'opérations de message (envois et réceptions) par seconde

`sema/s` Nombre d'opérations de sémaphore par seconde

L'exemple abrégé suivant illustre la sortie de la commande `sar -m`.

```
$ sar -m

SunOS balmy 5.10 Generic_144500-10 sun4v    ...

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
```

Vérification de l'activité de chargement de page (`sar -p`)

Utilisez la commande `sar -p` pour obtenir un rapport sur l'activité de chargement de page, qui inclut les défauts de protection et de traduction.

```
$ 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
```

La liste suivante décrit les statistiques rapportées à partir de l'option `-p`.

`atch/s` Nombre de défauts de page, par seconde, qui sont résolus en récupérant une page actuellement en mémoire (pages jointes par seconde). Les instances comprennent la réallocation d'une page incorrecte dans la liste des espaces libres et le partage d'une page de texte actuellement utilisée par un autre processus. Par exemple, plusieurs processus qui accèdent au même le texte du programme.

`pgin/s` Nombre de fois, par seconde, où les systèmes de fichiers reçoivent des demandes de chargement de page.

`ppgin/s` Nombre de pages transférées en mémoire par seconde. Une seule demande de chargement de page, par exemple une demande de verrou logiciel (voir `slock/s`) ou d'une grande taille de bloc, peut impliquer le chargement de plusieurs pages.

`pflt/s` Nombre de défauts de page résultant d'erreurs de protection. Les instances d'erreurs de protection indiquent un accès non autorisé à une page et la copie sur écriture. En règle générale, ce nombre est essentiellement constitué de copies sur écriture.

`vflt/s` Nombre de défauts de page liés à la traduction de l'adresse, par seconde. Ces erreurs sont appelées erreurs de validité ; elles se produisent lorsqu'une entrée de table de processus n'existe pas pour une adresse virtuelle donnée.

`slock/s` Nombre d'erreurs, par seconde, causées par des demandes de verrou logiciel nécessitant des E/S physiques. Le transfert de données d'un disque vers la mémoire constitue un exemple d'occurrence de demande de verrou logiciel. Le système bloque la page qui doit recevoir les données de sorte qu'elle ne puisse être ni réclamée ni utilisée par un autre processus.

EXEMPLE 3-8 Vérification de l'activité de chargement de page (`sar -p`)

L'exemple suivant illustre la sortie de la commande `sar -p`.

```
$ sar -p

SunOS balmy 5.10 Generic_144500-10 sun4v    ...

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
```

Vérification de l'activité de la file d'attente (`sar -q`)

Utilisez la commande `sar -q` pour signaler les informations suivantes :

- Longueur moyenne de la file d'attente lorsqu'elle est occupée.
- Pourcentage de temps pendant lequel la file d'attente est occupée.

```
$ sar -q
00:00:00 runq-sz %runocc swpq-sz %swpocc
```

La sortie de l'option `-q` est décrite ci-dessous.

`runq-sz`	Nombre de threads de noyau dans la mémoire qui attendent l'exécution d'une CPU. En règle générale, cette valeur doit être inférieure à 2. Des valeurs toujours plus élevées indiquent que le système est lié à la CPU.
`%runocc`	Pourcentage de temps pendant lequel les files d'attente de répartition sont occupées.
`swpq-sz`	Nombre moyen de processus extraits du swap.
`%swpocc`	Pourcentage de temps pendant lequel les processus sont extraits du swap.

EXEMPLE 3-9 Vérification de l'activité de la file d'attente

L'exemple suivant illustre la sortie de la commande `sar -q`. Si la valeur `%runocc` est élevée (supérieure à 90 %) et si la valeur `runq-sz` est supérieure à 2, la CPU est très chargée et la réponse est altérée. Dans ce cas, une nouvelle capacité de CPU peut être nécessaire pour obtenir une réponse acceptable du système.

```
# sar -q
SunOS balmy 5.10 Generic_144500-10 sun4v    ...

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
```

Vérification de la mémoire non utilisée (`sar -r`)

Utilisez la commande `sar -r` pour indiquer le nombre de pages de mémoire et de blocs de disque de fichier swap qui ne sont actuellement pas utilisés.

```
$ sar -r
00:00:00 freemem freeswap
01:00:00    2135   401922
```

La sortie de l'option `-r` est la suivante :

`freemem`	Nombre moyen de pages de mémoire disponibles pour les processus utilisateur dans les intervalles échantillonnés par la commande. La taille de page dépend de la machine.
`freeswap`	Nombre de blocs de disque de 512 octets qui sont disponibles pour la permutation de page.

EXEMPLE 3-10 Vérification de la mémoire non utilisée (`sar -r`)

L'exemple suivant illustre la sortie de la commande `sar -r`.

```
$ sar -r

SunOS balmy 5.10 Generic_144500-10 sun4v    ...

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
```

Vérification de l'utilisation de la CPU (`sar -u`)

Utilisez la commande `sar -u` pour afficher les statistiques d'utilisation de CPU.

```
$ sar -u
00:00:00    %usr    %sys    %wio   %idle
01:00:00       0       0       0     100
```

La commande `sar` sans aucune option est équivalente à la commande `sar -u`. A un moment donné, le processeur est occupé ou inactif. Lorsqu'il est occupé, le processeur est en mode utilisateur ou système. Lorsqu'il est inactif, le processeur attend la fin de l'E/S ou n'a aucune tâche à effectuer.

La sortie de l'option `-u` est la suivante :

`%usr` — Pourcentage de temps pendant lequel le processeur est en mode utilisateur.

`%sys` — Pourcentage de temps pendant lequel le processeur est en mode système.

`%wio` — Pourcentage de temps pendant lequel le processeur est inactif et en attente de fin d'E/S.

`%idle` — Pourcentage de temps pendant lequel le processeur est inactif et n'attend pas la fin d'E/S.

Une valeur `%wio` élevée signifie généralement qu'un ralentissement de disque s'est produit.

EXEMPLE 3-11 Vérification de l'utilisation de la CPU (`sar -u`)

L'exemple suivant illustre la sortie de la commande `sar -u`.

```
$ sar -u

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
```

Vérification du statut des tables système (`sar -v`)

Utilisez la commande `sar -v` pour indiquer le statut de la table des processus, d'inodes, de fichiers et d'enregistrements de mémoire partagée.

```
$ 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
```

La liste suivante décrit la sortie de l'option `-v`.

`proc-sz` — Nombre d'entrées de processus (structures `proc`) en cours d'utilisation ou d'allocation dans le noyau.

`inod-sz` — Nombre total d'inodes en mémoire par rapport au nombre maximal d'inodes alloués dans le noyau. Ce nombre n'est pas un filigrane élevé strict. Le nombre peut être dépassé.

`file-sz` — Taille de la table de fichiers système ouverte. `sz` prend la valeur `0`, parce que l'espace est alloué de manière dynamique pour la table de fichiers.

`ov` — Dépassements qui surviennent entre les points d'échantillonnage de chaque table.

`lock-sz` — Nombre d'entrées de la table des enregistrements de la mémoire partagée qui sont en cours d'utilisation ou d'allocation dans le noyau. `sz` prend la valeur `0`, parce que l'espace est alloué de manière dynamique pour la table des enregistrements de mémoire partagée.

EXEMPLE 3-12 Vérification du statut des tables système (`sar -v`)

L'exemple abrégé suivant illustre la sortie de la commande `sar -v`. Cet exemple montre que toutes les tables sont suffisamment grandes pour éviter les débordements. Ces tables sont allouées de façon dynamique en fonction de la quantité de mémoire physique.

```
$ sar -v

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
```

Vérification de l'activité de swap (`sar -w`)

Utilisez la commande `sar -w` pour générer des rapports sur l'activité de permutation et de commutation.

```
$ sar -w
00:00:00 swpin/s bswin/s swpot/s bswot/s pswch/s
01:00:00    0.00     0.0    0.00     0.0      22
```

La liste suivante décrit les valeurs cibles et les observations relatives à la sortie de la commande `sar -w`.

`swpin/s` Nombre de transferts LWP vers la mémoire par seconde.

`bswin/s` Nombre de blocs transférés pour une introduction dans le swap par seconde. `/* (float)PGTOBLK(xx->cvmi.pgswapin) / sec_diff */`.

`swpot/s` Nombre moyen de processus qui sont extraits du swap par seconde. Si le nombre est supérieur à 1, vous devrez peut-être augmenter la mémoire.

`bswot/s` Nombre de blocs transférés pour une extraction du swap par seconde.

`pswch/s` Nombre de commutateurs de thread de noyau, par seconde.

Remarque - Toutes les introductions de processus dans le swap comprennent l'initialisation du processus.

EXEMPLE 3-13 Vérification de l'activité de permutation (`sar -w`)

L'exemple suivant illustre la sortie de la commande `sar -w`.

```
$ sar -w

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
```

Vérification de l'activité du terminal (`sar -y`)

Utilisez la commande `sar -y` pour surveiller les activités des périphériques du terminal.

```
$ sar -y
00:00:00 rawch/s canch/s outch/s rcvin/s xmtin/s mdmin/s
01:00:00       0       0       0       0       0       0
```

Si vous disposez d'un grand nombre de terminaux d'E/S, vous pouvez utiliser ce rapport pour déterminer si des lignes sont défectueuses. Les activités enregistrées sont définies dans la liste ci-dessous.

`rawch/s`	Caractères d'entrée (file d'attente brute) par seconde
`canch/s`	Caractères d'entrée traités par la règle (file d'attente canonique) par seconde
`outch/s`	Caractères de sortie (file d'attente de sortie) par seconde
`rcvin/s`	Interruptions matérielles du récepteur par seconde
`xmtin/s`	Interruptions matérielles de l'émetteur par seconde
`mdmin/s`	Interruptions du modem par seconde

Le nombre d'interruptions du modem par seconde (`mdmin/s`) doit être proche de zéro. Le nombre d'interruptions de réception et de transmission par seconde (`xmtin/s` et `rcvin/s`) doit être inférieur ou égal au nombre de caractères entrants ou sortants, respectivement. Si ce n'est pas le cas, recherchez les lignes défectueuses.

EXEMPLE 3-14 Vérification de l'activité du terminal (`sar -y`)

L'exemple suivant illustre la sortie de la commande `sar -y`.

```
$ sar -y

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
```

Vérification des performances globales du système (`sar -A`)

Utilisez la commande `sar -A` pour afficher les statistiques issues de toutes les options afin de fournir une vue des performances globales du système.

Cette commande fournit une perspective plus globale. Si les données de plusieurs segments temporels sont affichées, le rapport comporte des moyennes.

Collecte automatique des données sur l'activité du système (`sar`)

La collecte automatique de données sur l'activité du système implique trois commandes : `sadc`, `sa1` et `sa2`.

L'utilitaire de collecte des données `sadc` collecte régulièrement les données sur l'activité du système et les enregistre dans un fichier au format binaire, à raison d'un fichier par 24 heures. Vous pouvez configurer l'exécution régulière de la commande `sadc` (généralement toutes les heures) et chaque fois que le système démarre en mode multiutilisateur. Les fichiers de données sont placés dans le répertoire `/var/adm/sa`. Chaque fichier est nommé `sa`*dd*, où *dd* indique la date du jour. Le format de la commande est le suivant :

```
/usr/lib/sa/sadc [t n] [ofile]
```

La commande échantillonne *n* fois avec un intervalle de *t* secondes, qui doit être supérieur à cinq secondes entre les échantillons. Cette commande écrit ensuite dans le fichier binaire *ofile* ou dans la sortie standard.

Exécution de la commande `sadc` lors de l'initialisation

La commande `sadc` doit être exécutée au moment de l'initialisation du système pour enregistrer les statistiques à partir de la date de réinitialisation des compteurs. Pour s'assurer que la commande `sadc` est exécutée au moment de l'initialisation, la commande `svcadm enable system/sar:default` crée un enregistrement dans le fichier de données quotidien.

L'entrée de la commande est au format suivant :

```
/usr/bin/su sys -c "/usr/lib/sa/sadc /var/adm/sa/sa`date +%d`"
```

Exécution périodique de la commande `sadc` avec le script `sa1`

Pour générer des enregistrements périodiques, vous devez exécuter régulièrement la commande `sadc`. Pour cela, la méthode la plus simple consiste à annuler le commentaire des lignes suivantes dans le fichier `/var/spool/cron/crontabs/sys` :

```
# 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
```

Les entrées par défaut de `sys crontab` effectuent les opérations suivantes :

- Les deux premières entrées `crontab` entraînent l'écriture d'un enregistrement dans le fichier `/var/adm/sa/sa`*jj* toutes les 20 minutes, de 8 h à 17 h, du lundi au vendredi, et toutes les heures le reste du temps.
- La troisième entrée écrit un enregistrement dans le fichier `/var/adm/sa/sar`*dd* toutes les heures, du lundi au vendredi, et inclut toutes les options `sar`.

Vous pouvez modifier ces valeurs par défaut en fonction de vos besoins.

Génération de rapports à l'aide du script shell `sa2`

Un autre script de shell, `sa2`, génère des rapports plutôt que des fichiers de données binaires. La commande `sa2` appelle la commande `sar` et écrit la sortie ASCII dans un fichier de rapport.

Configuration de la collecte automatique des données (`sar`)

La commande `sar` peut servir à recueillir les données sur l'activité du système ou à rapporter les données collectées dans les fichiers d'activité quotidiens créés par la commande `sadc`.

La commande `sar` présente les formats suivants :

```
sar [-aAbcdgkmpqruvwy] [-o file] t [n]

sar [-aAbcdgkmpqruvwy] [-s time] [-e time] [-i sec] [-f file]
```

Le premier format échantillonne les compteurs d'activité cumulés dans le système d'exploitation toutes les *t* secondes, *n* fois. La valeur *t* doit être d'au moins cinq secondes. Autrement, la commande elle-même peut influer sur l'échantillon. Vous devez spécifier l'intervalle de prise des échantillons. Sinon, la commande fonctionne selon le second format. La valeur par défaut de *n* est 1.

L'exemple suivant, à l'aide du second format, prend deux échantillons séparés de 10 secondes. Si l'option `-o` a été spécifiée, les échantillons sont enregistrés au format binaire.

```
$ sar -u 10 2
```

La commande `sar` avec le deuxième format, sans spécification de l'intervalle d'échantillonnage ou du nombre d'échantillons, extrait les données d'un fichier enregistré auparavant. Ce fichier correspond au fichier spécifié par l'option `-f` ou, par défaut, au fichier d'activité quotidien standard, `/var/adm/sa/sa`*dd*, du jour le plus récent.

Les options `-s` et `-e` définissent les heures de début et de fin du rapport. Les heures de début et de fin sont sous la forme *hh*[*:mm*[*:ss*]], où *hh*, *mm* et *ss* représentent les heures, minutes et secondes.

L'option `-i` indique, en secondes, l'intervalle de sélection des enregistrements. Si l'option `-i` est absente, tous les intervalles trouvés dans le fichier d'activité quotidien sont signalés.

Les options de la commande `sar` et leurs actions sont les suivantes :

Remarque - Si vous n'utilisez aucune option, cela équivaut à appeler la commande `sar` avec l'option `-u`.

`-a` Vérifie les opérations d'accès aux fichiers.

`-b`	Vérifie l'activité du tampon.
`-c`	Vérifie les appels système.
`-d`	Vérifie l'activité de chaque périphérique de bloc.
`-g`	Vérifie le renvoi de page et la libération de la mémoire.
`-k`	Vérifie l'allocation de mémoire du noyau.
`-m`	Vérifie la communication interprocessus.
`-nv`	Vérifie le statut des tables système.
`-p`	Vérifie l'activité swap et de distribution.
`-q`	Vérifie l'activité de la file d'attente.
`-r`	Vérifie la mémoire non utilisée.
`-u`	Vérifie l'utilisation de la CPU.
`-w`	Vérifie le volume de permutation et de commutation.
`-y`	Vérifie l'activité du terminal.
`-A`	Génère des rapports sur les performances globales du système, ce qui équivaut à saisir toutes les options

▼ Configuration de la collecte automatique des données

1. **Prenez le rôle `root`.**

   Reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

2. **Exécutez la commande `svcadm enable system/sar:default`.**

   Cette version de la commande `sadc` crée un enregistrement spécial qui marque l'heure de la réinitialisation des compteurs (temps d'initialisation).

3. **Modifiez le fichier `crontab /var/spool/cron/crontabs/sys`.**

   Remarque - Ne modifiez pas directement un fichier `crontab`. Utilisez plutôt la commande `crontab -e` pour apporter des modifications à un fichier `crontab`.

```
# crontab -e sys
```

4. **Annulez le commentaire des lignes suivantes :**

```
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
```

Pour plus d'informations, reportez-vous à la page de manuel `crontab`(1).

♦ ♦ ♦ CHAPITRE 4

Planification des tâches système

Ce chapitre décrit la planification de tâches de routine ou ponctuelles du système à l'aide des commandes `crontab` et `at`.

Ce chapitre explique également comment contrôler l'accès à ces commandes en utilisant les fichiers suivants :

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

Ce chapitre comprend les sections suivantes :

Méthodes d'exécution automatique des tâches système

Vous pouvez configurer l'exécution automatique de plusieurs tâches système. Certaines de ces tâches doivent être exécutées à intervalles réguliers. D'autres tâches ne doivent être exécutées qu'une seule fois, peut-être en dehors des heures de travail, par exemple le soir ou le week-end.

Cette section contient des informations générales à propos des commandes `crontab` et `at`, qui vous permettent de planifier l'exécution automatique de tâches de routine. La commande `crontab` planifie les tâches répétitives. La commande `at` planifie les tâches ponctuelles.

Le tableau suivant résume les commandes `crontab` et `at`, ainsi que les fichiers vous permettant de contrôler l'accès à ces commandes.

TABLEAU 4-1 Récapitulatif des commandes : planification des tâches système

Commande	Tâches planifiées	Emplacement des fichiers	Fichiers qui contrôlent l'accès
`crontab`	Plusieurs tâches système à	`/var/spool/cron/crontabs`	`/etc/cron.d/cron.allow` et `/etc/cron.d/cron.deny`

Commande	Tâches planifiées	Emplacement des fichiers	Fichiers qui contrôlent l'accès
	intervalles réguliers		
`at`	Une seule tâche système	`/var/spool/cron/atjobs`	`/etc/cron.d/at.deny`

Planification de tâches répétitives avec `crontab`

Vous pouvez planifier l'exécution des tâches de routine liées à l'administration du système tous les jours, toutes les semaines ou tous les mois à l'aide de la commande `crontab`.

Les tâches d'administration système quotidiennes `crontab` peuvent comprendre entre autres :

- Suppression des fichiers datés de plusieurs jours des répertoires temporaires
- Exécution des commandes récapitulatives de comptabilisation
- Prise d'instantanés du système à l'aide des commandes `df` et `ps`
- Surveillance quotidienne de la sécurité
- Exécution des sauvegardes du système

Les tâches d'administration système hebdomadaires avec `crontab` peuvent comprendre entre autres :

- Reconstruction de la base de données `catman` à utiliser avec la commande `man -k`
- Exécution de la commande `fsck -n` pour répertorier les problèmes de disque

Les tâches d'administration système mensuelles avec `crontab` peuvent comprendre entre autres :

- Liste des fichiers non utilisés au cours d'un mois donné
- Génération des rapports comptables mensuels

En outre, vous pouvez planifier des commandes `crontab` pour exécuter d'autres tâches système de routine, telles que l'envoi de rappels ou la suppression de fichiers de sauvegarde.

Pour consulter des instructions détaillées sur la planification des tâches avec `crontab`, reportez-vous à la section “Création ou modification d'un fichier `crontab`” à la page 85.

Planification d'une tâche ponctuelle avec `at`

La commande `at` vous permet de planifier l'exécution d'une tâche à un moment ultérieur. Le travail peut comporter une seule commande ou un script.

De manière analogue à `crontab`, la commande `at` vous permet de planifier l'exécution automatique de tâches de routine. Toutefois, contrairement aux fichiers `crontab`, les fichiers `at` n'exécutent qu'une seule fois leurs tâches. Ils sont ensuite supprimés de leur répertoire. Par conséquent, la commande `at` se révèle particulièrement utile pour l'exécution de commandes ou de scripts uniques qui orientent la sortie dans des fichiers séparés à des fins d'examen ultérieur.

La soumission d'un travail `at` implique la saisie d'une commande et le suivi de la syntaxe de commande `at` pour spécifier les options de planification de l'exécution du travail. Pour plus d'informations sur la soumission de tâches avec `at`, reportez-vous à “Soumission d'un fichier de travail `at`” à la page 93.

La commande `at` enregistre la commande ou le script exécuté, ainsi qu'une copie de la variable d'environnement actuelle, dans le répertoire `/var/spool/cron/atjobs`. Le nom de fichier d'une tâche `at` correspond à un long nombre qui spécifie son emplacement dans la file d'attente `at`, suivi de l'extension `.a`, par exemple `793962000.a`.

Au démarrage, le démon `cron` cherche les travaux `at` et vérifie si de nouveaux travaux sont soumis. Une fois que le démon `cron` a exécuté un travail `at`, le fichier du travail `at` est supprimé du répertoire `atjobs`. Pour plus d'informations, reportez-vous à la page de manuel `at`(1).

Pour obtenir les instructions détaillées sur la planification des travaux `at`, reportez-vous à la section “Création d'un travail `at`” à la page 93.

Planification des tâches système

Cette section comprend des tâches permettant de planifier les tâches système à l'aide de fichiers `crontab`.

Liste des tâches pour la création et modification de fichiers `crontab`

Tâche	Description	Pour obtenir des instructions
Création ou modification d'un fichier `crontab`	Utilisez la commande `crontab -e` pour créer ou modifier un fichier `crontab`.	“Création ou modification d'un fichier `crontab`” à la page 85
Vérification de l'existence d'un fichier `crontab`	Utilisez la commande `ls -l` pour vérifier le contenu du fichier `/var/spool/cron/crontabs`.	“Vérification de l'existence d'un fichier `crontab`” à la page 86
Affichage d'un fichier `crontab`	Utilisez la commande `ls -l` pour afficher le fichier `crontab`.	“Affichage d'un fichier `crontab`” à la page 86

Tâche	Description	Pour obtenir des instructions
Suppression d'un fichier `crontab`	Le fichier `crontab` est configuré avec des autorisations restrictives. Utilisez la commande `crontab -r` plutôt que la commande `rm` pour supprimer un fichier `crontab`.	“Suppression d'un fichier `crontab`” à la page 88
Refus de l'accès à `crontab`.	Pour refuser aux utilisateurs l'accès aux commandes `crontab`, ajoutez des noms d'utilisateurs au fichier `/etc/cron.d/cron.deny`.	“Refus d'accès à la commande `crontab`” à la page 90
Limitation de l'accès à `crontab` aux utilisateurs spécifiés	Pour permettre aux utilisateurs d'accéder à la commande `crontab`, ajoutez des noms d'utilisateurs au fichier `/etc/cron.d/cron.allow`.	“Restriction de l'accès à la commande `crontab` aux utilisateurs spécifiés” à la page 90

Planification d'une tâche système répétitive (`cron`)

Les sections suivantes décrivent la création, la modification, l'affichage et la suppression des fichiers `crontab`, ainsi que la façon d'en contrôler l'accès.

Contenu d'un fichier `crontab`

Le démon `cron` planifie les tâches système en fonction des commandes contenues dans chaque fichier `crontab`. Un fichier `crontab` se compose de commandes, une par ligne, à exécuter à intervalles réguliers. Le début de chaque ligne indique la date et l'heure auxquelles le démon `cron` doit exécuter la commande.

Par exemple, un fichier `crontab` nommé `root` est fourni pendant l'installation du logiciel Oracle Solaris. Le contenu du fichier inclut les lignes de commande suivantes :

```
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)
```

La sortie de chacune de ces lignes de commande est la suivante :

- La première ligne exécute la commande `logadm` à 3h10 chaque jour.
- La deuxième ligne exécute le script `nfsfind` chaque dimanche à 3h15.
- La troisième ligne exécute un script qui vérifie le passage à l'heure d'été (et apporte des corrections, si nécessaire) chaque jour à 2h10.

  En l'absence de fuseau horaire HTR et de fichier `/etc/rtc_config`, cette entrée n'a aucun effet.

x86 uniquement - Le script `/usr/sbin/rtc` ne peut s'exécuter que sur un système x86.

- La quatrième ligne recherche (et supprime) les entrées en double dans le tableau des services de sécurité génériques, `/etc/gss/gsscred_db`, chaque jour à 3h30.

Pour plus d'informations sur la syntaxe des lignes dans un fichier `crontab`, reportez-vous à la section “Syntaxe des entrées du fichier `crontab`” à la page 84.

Les fichiers `crontab` sont enregistrés dans le répertoire `/var/spool/cron/crontabs`. Plusieurs fichiers `crontab` en plus de `root` sont fournis pendant l'installation du logiciel Oracle Solaris.

`adm`	Comptabilisation
`root`	Fonctions générales du système et nettoyage du système de fichiers
`sys`	Collecte des données sur les performances
`uucp`	Nettoyage `uucp` général

En plus des fichiers `crontab` par défaut, vous pouvez créer des fichiers `crontab` personnalisés pour planifier vos propres tâches système. Les fichiers `crontab` personnalisés sont nommés d'après les comptes utilisateur dans lesquels ils sont créés, par exemple `jean`, `marie`, `dubois` ou `dupont`.

Pour accéder aux fichiers `crontab` appartenant à `root` ou à d'autres utilisateurs, des privilèges de superutilisateur sont requis.

Gestion de la planification par le démon `cron`

Le démon `cron` gère la planification automatique des commandes `crontab`. Le rôle du démon `cron` consiste à vérifier dans le répertoire `/var/spool/cron/crontab` la présence de fichiers `crontab`.

Le démon `cron` effectue les tâches suivantes au démarrage :

- Il vérifie l'existence de nouveaux fichiers `crontab`.
- Il lit les heures d'exécution qui sont répertoriées à l'intérieur de ces fichiers.
- Il soumet l'exécution des commandes au bon moment.
- Il reçoit les notifications des commandes `crontab` relatives aux fichiers `crontab` mis à jour.

De la même manière, le démon `cron` contrôle la planification des fichiers `at`. Ces fichiers sont stockés dans le répertoire `/var/spool/cron/atjobs`. Le démon `cron` reçoit également les notifications des commandes `crontab` relatives aux travaux `at` soumis.

Syntaxe des entrées du fichier `crontab`

Un fichier `crontab` se compose de commandes, une par ligne, qui s'exécutent automatiquement à l'heure spécifiée par les cinq premiers champs de chaque ligne de commande, séparés par des espaces.

TABLEAU 4-2 Valeurs acceptables pour les champs d'heure `crontab`

Champ d'heure	valeurs
Minute	0-59
Heure	0-23
Jour du mois	1-31
Mois	1-12
Jour de la semaine	0-6 (0 = dimanche)

Suivez les instructions ci-dessous pour utiliser des caractères spéciaux dans les champs d'heure `crontab` :

- Utilisez un espace pour séparer chaque champ.
- Utilisez une virgule pour séparer plusieurs valeurs.
- Utilisez un trait d'union pour définir une plage de valeurs.
- Utilisez l'astérisque comme caractère générique pour inclure toutes les valeurs possibles.
- Utilisez un signe de commentaire (#) au début d'une ligne pour indiquer un commentaire ou une ligne vierge.

Par exemple, l'entrée de commande `crontab` suivante affiche un message de rappel dans la fenêtre de la console utilisateur à 16 heures, le premier et le quinze de chaque mois.

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

Chaque commande contenue dans un fichier `crontab` doit comporter une ligne, même si cette dernière est très longue. Le fichier `crontab` ne reconnaît pas les retours chariot supplémentaires. Pour plus d'informations sur les entrées et options de la commande `crontab`, reportez-vous à la page de manuel `crontab`(1).

Création et modification de fichiers `crontab`

La façon la plus simple de créer un fichier `crontab` consiste à utiliser la commande `crontab -e`. Cette commande ouvre l'éditeur de texte défini pour votre environnement système dans la variable d'environnement `EDITOR`. Si cette variable n'a pas été définie, la commande `crontab` utilise l'éditeur par défaut, `ed`.

L'exemple suivant montre comment déterminer si un éditeur est défini et configure `vi` comme éditeur par défaut.

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

Lorsque vous créez un fichier crontab, il est automatiquement placé dans le répertoire /var/spool/cron/crontabs et reçoit votre nom d'utilisateur. Vous pouvez créer ou modifier un fichier crontab pour un autre utilisateur ou pour l'utilisateur root si vous disposez des privilèges d'utilisateur root.

▼ Création ou modification d'un fichier crontab

Avant de commencer

Si vous créez ou modifiez un fichier crontab appartenant à un autre utilisateur, vous devez prendre le rôle root. Reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Vous n'avez pas besoin de prendre le rôle root pour modifier votre propre fichier crontab.

1. Créez un fichier crontab ou modifiez un fichier existant.

```
# crontab -e [username]
```

où *username* indique le nom du compte utilisateur pour lequel vous souhaitez créer ou modifier un fichier crontab. Vous pouvez créer votre propre fichier crontab sans privilèges de superutilisateur, mais vous devez disposer des privilèges de superutilisateur pour créer ou modifier un fichier crontab pour root ou un autre utilisateur.

Attention - Si par mégarde vous écrivez la commande crontab sans option, appuyez sur le caractère d'interruption de l'éditeur pour quitter sans enregistrer les modifications. En revanche, si enregistrez les modifications et quittez le fichier, le fichier crontab est remplacé par un fichier vide.

2. Ajoutez des lignes de commande au fichier crontab.

Suivez la syntaxe décrite dans “Syntaxe des entrées du fichier crontab” à la page 84. Le fichier crontab est placé dans le répertoire /var/spool/cron/crontabs.

3. Vérifiez les modifications apportées au fichier crontab.

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

Exemple 4-1 Création d'un fichier crontab

L'exemple suivant décrit la création d'un fichier crontab pour un autre utilisateur.

```
# crontab -e mary
```

L'entrée de commande suivante ajoutée à un nouveau fichier crontab supprime automatiquement tous les fichiers journaux du répertoire personnel de Mary à 1h00 chaque dimanche matin. L'entrée de commande ne redirigeant pas la sortie, les caractères de redirection sont ajoutés à la ligne de commande après *.log. L'exécution correcte de la commande est ainsi garantie.

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

Affichage et vérification de fichiers crontab

Vous pouvez utiliser la commande crontab -l, pour afficher et vérifier le contenu d'un fichier crontab.

Vérification de l'existence d'un fichier crontab

Pour vérifier l'existence d'un fichier crontab pour un utilisateur, utilisez la commande ls -l dans le répertoire /var/spool/cron/crontabs. Par exemple, l'exemple de sortie suivant indique que des fichiers crontab existent pour divers utilisateurs du système.

```
$ ls -l /var/spool/cron/crontabs
drwxr-xr-x   2 root     sys           12 Nov 26 16:55 ./
drwxr-xr-x   4 root     sys            4 Apr 28  2012 ../
-rw-------   1 root     sys          190 Jun 28  2011 adm
-rw-------   1 root     staff          0 Nov 13  2012 mary
-rw-------   1 root     un           437 Oct  8  2012 johndoe
-r--------   1 root     root         453 Apr 28  2012 lp
-rw-------   1 root     sparccad      63 Jul 17 10:39 mary2
-rw-------   1 root     sparccad     387 Oct 14 15:15 johndoe2
-rw-------   1 root     other       2467 Nov 26 16:55 root
-rw-------   1 root     sys          308 Jun 28  2011 sys
-rw-------   1 root     siete        163 Nov 20 10:40 mary3
-r--------   1 root     sys          404 Jan 24  2013 uucp
```

Affichage d'un fichier crontab

La commande crontab -l affiche le contenu d'un fichier crontab, comme la commande cat affiche le contenu de fichiers d'autres types. Vous n'avez pas besoin d'accéder au répertoire /var/spool/cron/crontabs (contenant les fichiers crontab) pour utiliser cette commande.

Par défaut, la commande crontab -l affiche votre propre fichier crontab. Pour afficher les fichiers crontab qui appartiennent à d'autres utilisateurs, vous devez être connecté en tant que superutilisateur.

```
You can use the crontab command as follows:

# crontab -l [username]
```

où *username* indique le nom du compte utilisateur pour lequel vous souhaitez afficher un fichier `crontab`. L'affichage du fichier `crontab` d'un autre utilisateur exige des privilèges de superutilisateur.

Attention - Si, par mégarde, vous saisissez la commande `crontab` sans option, appuyez sur le caractère d'interruption de l'éditeur pour quitter sans enregistrer les modifications. En revanche, si enregistrez les modifications et quittez le fichier, le fichier `crontab` est remplacé par un fichier vide.

EXEMPLE 4-2 Affichage d'un fichier `crontab`

L'exemple suivant montre comment afficher le contenu du fichier `crontab` par défaut à l'aide de la commande `crontab -l`.

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

EXEMPLE 4-3 Affichage du fichier `root crontab` par défaut

Cet exemple illustre l'affichage du fichier `crontab root` par défaut.

```
$ su
Password:

# 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___
```

EXEMPLE 4-4 Affichage du fichier `crontab` d'un autre utilisateur

Cet exemple illustre l'affichage du fichier `crontab` d'un autre utilisateur.

```
$ su
Password:
# crontab -l jones
13 13 * * * cp /home/jones/work_files /usr/backup/. > /dev/null 2>&1
```

Suppression des fichiers crontab

Par défaut, les fichiers crontab sont protégés de sorte que vous ne puissiez pas supprimer par inadvertance un fichier crontab en utilisant la commande rm. Utilisez plutôt la commande crontab -r pour supprimer des fichiers crontab.

Par défaut, la commande crontab -r affiche votre propre fichier crontab.

Vous n'avez pas besoin d'accéder au répertoire /var/spool/cron/crontabs (contenant les fichiers crontab) pour utiliser cette commande.

▼ Suppression d'un fichier crontab

Avant de commencer

Prenez le rôle root pour supprimer un fichier crontab appartenant à root ou à un autre utilisateur. Les rôles contiennent des autorisations et des commandes privilégiées. Reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

Vous n'avez pas besoin de prendre le rôle root pour supprimer votre propre fichier crontab.

1. **Supprimez le fichier crontab.**

```
# crontab -r [username]
```

où *username* indique le nom du compte utilisateur pour lequel vous souhaitez supprimer un fichier crontab. Pour supprimer des fichiers crontab pour un autre utilisateur, prenez le rôle root.

Attention - Si, par mégarde, vous saisissez la commande crontab sans option, appuyez sur le caractère d'interruption de l'éditeur pour quitter sans enregistrer les modifications. En revanche, si enregistrez les modifications et quittez le fichier, le fichier crontab est remplacé par un fichier vide.

2. **Vérifiez que le fichier crontab a bien été supprimé.**

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

Exemple 4-5 Suppression d'un fichier crontab

L'exemple suivant montre comment l'utilisateur smith supprime son propre fichier crontab à l'aide de la commande crontab -r.

```
$ ls /var/spool/cron/crontabs
```

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

Contrôle de l'accès à la commande crontab

Vous pouvez contrôler l'accès à la commande crontab au moyen de deux fichiers du répertoire /etc/cron.d : cron.deny et cron.allow. Ces fichiers permettent uniquement aux utilisateurs spécifiés d'exécuter les tâches de commande crontab telles que la création, la modification, l'affichage ou la suppression de leurs propres fichiers crontab.

Les fichiers cron.deny et cron.allow contiennent une liste de noms d'utilisateur, chaque ligne comportant un seul nom.

Ces fichiers de contrôle d'accès fonctionnent comme suit :

- Si le fichier cron.allow existe, seuls les utilisateurs qui y figurent peuvent créer, modifier, afficher ou supprimer les fichiers crontab.
- Si cron.allow n'existe pas, tous les utilisateurs peuvent soumettre des fichiers crontab, à l'exception de ceux qui figurent dans cron.deny.
- Si ni cron.allow ni cron.deny n'existent, vous devez prendre le rôle root pour exécuter la commande crontab.
- Vous devez prendre le rôle root pour pouvoir modifier ou créer les fichiers cron.deny et cron.allow.

Le fichier cron.deny, créé pendant l'installation du logiciel Oracle Solaris, contient les noms d'utilisateur suivants :

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

Aucun des noms d'utilisateur du fichier cron.deny par défaut ne peut accéder à la commande crontab. Vous pouvez modifier ce fichier afin d'ajouter d'autres noms d'utilisateur qui n'auront pas accès à la commande crontab.

Etant donné qu'aucun fichier cron.allow n'est fourni par défaut, tous les utilisateurs sauf ceux qui figurent dans le fichier cron.deny ont accès à la commande crontab. Si vous créez un fichier cron.allow, seuls ces utilisateurs peuvent accéder à la commande crontab.

▼ Refus d'accès à la commande crontab

1. **Prenez le rôle root.**

   Reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

2. **Modifiez le fichier /etc/cron.d/cron.deny et ajoutez des noms d'utilisateur, un par ligne, qui n'auront pas accès aux commandes crontab.**

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

3. **Vérifiez que le fichier /etc/cron.d/cron.deny contient les nouvelles entrées.**

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

▼ Restriction de l'accès à la commande crontab aux utilisateurs spécifiés

1. **Prenez le rôle root.**

   Reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

2. **Créez le fichier /etc/cron.d/cron.allow.**

3. **Ajoutez le rôle root au fichier cron.allow.**

   Si vous n'ajoutez pas root au fichier, l'accès superutilisateur aux commandes crontab sera refusé.

4. **Ajoutez les noms d'utilisateur, un par ligne, qui peuvent uitliser la commande `crontab`.**

   root
   username1
   username2
   username3
   .
   .
   .

Exemple 4-6 Restriction de l'accès à la commande `crontab` aux utilisateurs spécifiés

L'exemple suivant illustre un fichier `cron.deny` qui empêche les noms d'utilisateur `jones`, `temp` et `visitor` d'accéder à la commande `crontab`.

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

L'exemple suivant montre un fichier `cron.allow`. Les utilisateurs `root`, `dupont` et `dubois` sont les seuls à pouvoir accéder à la commande `crontab`.

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

Vérification de l'accès restreint à la commande `crontab`

Pour vérifier si un utilisateur spécifique peut accéder à la commande `crontab`, utilisez la commande `crontab -l` lorsque vous êtes connecté au compte d'utilisateur.

```
$ crontab -l
```

Si l'utilisateur a accès à la commande `crontab` et a déjà créé un fichier `crontab`, celui-ci est affiché. Dans le cas contraire, si l'utilisateur peut accéder à la commande `crontab`, mais si aucun fichier `crontab` n'existe, un message similaire à celui ci-dessous s'affiche :

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

Soit l'utilisateur est répertorié dans le fichier cron.allow (si ce fichier existe), soit il n'est pas répertorié dans le fichier cron.deny.

Si l'utilisateur ne peut pas accéder à la commande crontab, le message suivant s'affiche, qu'un fichier crontab antérieur existe ou non :

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

Ce message signifie que l'utilisateur ne figure pas dans le fichier cron.allow (si ce fichier existe) ou qu'il figure dans le fichier cron.deny.

Planification de tâches à l'aide de la commande at

Cette section inclut des tâches permettant de planifier les tâches système de routine à l'aide de la commande at.

Utilisation de la commande at

Suivez la procédure décrite ci-après pour créer et gérer des tâches de routine sur votre système.

Planification d'une seule tâche système (at)

Les sections suivantes décrivent l'utilisation de la commande at pour réaliser les tâches suivantes :

- Planification des travaux (commande et scripts) à exécuter ultérieurement
- Permet d'afficher et d'enlever les ordres de fabrication
- Contrôle de l'accès à la commande at

Par défaut, les utilisateurs peuvent créer, afficher et supprimer leurs propres fichiers de travaux at. Pour accéder aux fichiers at appartenant à root ou à d'autres utilisateurs, vous devez disposer des privilèges de superutilisateur.

Soumission d'un fichier de travail at

Lorsque vous soumettez un travail at, un numéro d'identification suivi de l'extension .a lui est attribué. Cette désignation représente le nom de fichier du travail, ainsi que son numéro dans la file d'attente.

La soumission d'un fichier de travail at implique les étapes suivantes :

1. Appel de l'utilitaire at et choix de l'heure d'exécution de la commande.
2. Saisie d'une commande ou d'un script à exécuter ultérieurement

---

Remarque - Si la sortie de cette commande ou de ce script est importante, veillez à la diriger vers un fichier à des fins d'examen ultérieur.

Par exemple, le travail at suivant supprime les fichiers noyau du compte d'utilisateur dubois vers minuit, le dernier jour de juillet.

```
$ 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
```

Création d'un travail at

La tâche suivante explique comment créer un travail at.

▼ Création d'un travail at

1. Lancez l'utilitaire at en spécifiant l'heure d'exécution du travail.

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

-m	Pour qu'il vous envoie un e-mail indique une fois le travail terminé.
time	Indique l'heure à laquelle vous souhaitez planifier le travail. Ajoutez am ou pm si vous ne spécifiez pas l'heure en fonction de l'horloge 24 heures. Les mots-clés acceptables sont midnight, noon et now. Les minutes sont facultatives.
date	Indique au moins les trois premières lettres du mois, le jour de la semaine, ou les mots-clés today ou tomorrow.

2. **A l'invite `at`, saisissez les commandes ou scripts à exécuter, à raison d'un par ligne.**
   Vous pouvez saisir plusieurs commandes en appuyant sur Entrée à la fin de chaque ligne.

3. **Appuyez sur Ctrl-D pour quitter l'utilitaire `at` et enregistrer le travail `at`.**
   Un numéro de file d'attente, qui désigne également le nom de fichier du travail, est attribué au travail `at`. Ce numéro est affiché lorsque vous quittez l'utilitaire `at`.

Exemple 4-7 Création d'un travail `at`

L'exemple suivant montre le travail `at` créé par l'utilisateur `jones` pour supprimer ses fichiers de sauvegarde à 19h30. Elle a utilisé l'option `-m` de manière à recevoir un e-mail une fois le travail terminé.

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

Elle a reçu un e-mail de confirmation de l'exécution de son travail `at`.

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

L'exemple suivant montre comment `jones` a planifié un grand travail `at` pour 4h00 samedi matin. La sortie du travail a été dirigée vers un fichier nommé `big.file`.

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

Affichage de la file d'attente `at`

Pour vérifier vos travaux en attente dans la file d'attente `at`, utilisez la commande `atq`.

```
$ atq
```

Cette commande affiche les informations d'état sur les travaux `at` créés.

Vérification d'un travail `at`

Pour vérifier la création d'un travail `at`, utilisez la commande `atq`. Dans l'exemple suivant, la commande `atq` confirme que les travaux `at` qui appartiennent à `jones` ont été soumis à la file d'attente.

```
$ 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
```

Affichage des travaux `at`

Pour afficher des informations sur les heures d'exécution des travaux `at`, utilisez la commande `at -l`.

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

où `-l` *job-id* est le numéro d'identification optionnel d'un travail particulier dont vous souhaitez afficher l'état. Le système sans être associé à un ID, la commande affiche le statut de tous les travaux soumis par un utilisateur.

EXEMPLE 4-8 Affichage des travaux `at`

L'exemple suivant illustre la sortie de la commande `at -l`, qui fournit des informations sur l'état de tous les travaux soumis par un utilisateur.

```
$ 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
```

L'exemple suivant illustre la sortie affichée lorsqu'un seul travail est spécifié avec la commande `at -l`.

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

▼ Suppression des travaux `at`

Avant de commencer Prenez le rôle `root` pour supprimer un travail `at` appartenant à `root` ou à un autre utilisateur. Reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

Vous n'avez pas besoin de prendre le rôle `root` pour supprimer votre propre travail `at`.

1. **Supprimez le travail `at` de la file d'attente avant qu'il soit exécuté.**

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

   où l'option `-r` *job-id* spécifie le numéro d'identification du travail à supprimer.

2. **Vérifiez que le travail `at` a bien été supprimé à l'aide de la commande `at -l` (ou `atq`).**

La commande at -l affiche les travaux restants dans la file d'attente at. Le travail dont vous avez indiqué le numéro d'identification ne doit pas apparaître.

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

Exemple 4-9 Suppression des travaux at

Dans l'exemple suivant, un utilisateur souhaite supprimer un travail at dont l'exécution est planifiée à 4h00 le 7 juillet. Tout d'abord, l'utilisateur affiche la file d'attente at pour localiser le numéro d'identification du travail. Ensuite, il supprime ce travail de la file d'attente at. Enfin, il vérifie que le travail a bien été supprimé de la file d'attente.

```
$ 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
```

Contrôle de l'accès à la commande at

Vous pouvez configurer un fichier pour contrôler l'accès à la commande at, spécifiant les utilisateurs autorisés à créer et supprimer travaux at ou à afficher les informations sur leur file d'attente. Le fichier qui contrôle l'accès à la commande at , /etc/cron.d/at.deny, contient une liste de noms d'utilisateur, un par ligne. Les utilisateurs qui figurent dans ce fichier ne peuvent pas accéder aux commandes at.

Le fichier at.deny, créé pendant l'installation du logiciel Oracle Solaris, contient les noms d'utilisateur suivants :

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

Avec les privilèges de superutilisateur, vous pouvez modifier le fichier at.deny et y ajouter les noms des utilisateurs dont vous souhaitez restreindre l'accès à la commande at.

Refus d'accès à la commande at

En tant qu'utilisateur root, modifiez le fichier /etc/cron.d/at.deny et ajoutez les noms d'utilisateur, un par ligne, auxsquels vous souhaitez refuser l'accès à la commande at.

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

EXEMPLE 4-10 Refus d'accès à at

L'exemple suivant montre un fichier at.deny modifié de sorte que les utilisateurs smith et jones n'ont pas accès à la commande at.

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

Vérification du refus d'accès à la commande at

Pour vérifier si un nom d'utilisateur a bien été ajouté au fichier /etc/cron.d/at.deny, utilisez la commande at -l en étant connecté avec le compte utilisateur en question. Par exemple, si l'utilisateur connecté smith n'a pas accès à la commande at, le message suivant s'affiche :

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

De même, si l'utilisateur tente de soumettre un travail at, le message suivant s'affiche :

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

Ce message confirme que l'utilisateur est répertorié dans le fichier at.deny.

Si l'accès à la commande at est autorisé, la commande at -l ne renvoie rien.

♦ ♦ ♦ CHAPITRE 5

Gestion de la console système, des périphériques terminaux et des services d'alimentation

Ce chapitre décrit la gestion de la console système et des périphériques terminaux connectés localement à l'aide du programme `ttymon` et des services d'alimentation du système.

Ce chapitre comprend les sections suivantes :

Gestion de la console système et des périphériques terminaux connectés localement

La console système est un terminal doté d'attributs spéciaux et utilisé à certaines fins. Par exemple, les messages du noyau qui sont destinés à un administrateur sont envoyés à la console et non à d'autres terminaux.

Un terminal est un moyen d'interaction avec Oracle Solaris. L'affichage des graphiques bitmap de votre système est différent de celui d'un terminal alphanumérique. Un terminal alphanumérique se connecte à un port série et affiche uniquement du texte. Vous n'avez pas à effectuer de procédure spéciale pour gérer l'affichage des graphiques.

Un terminal peut également être associé à l'écran physique et à la disposition du clavier d'un ordinateur. Ce qui distingue le terminal graphique, c'est qu'il doit être associé à la carte graphique et à l'écran d'un ordinateur. Par conséquent, au lieu de transmettre des caractères à partir d'un port série, les caractères sont tracés sur la mémoire de la carte graphique qui se trouve dans l'ordinateur.

Services SMF gérant la console système et les périphériques terminaux connectés localement

La console système et les périphériques terminaux connectés localement sont représentés sous forme d'instances du service SMF, `svc:/system/console`. Ce service définit en grande partie le comportement, chaque instance présentant des valeurs de remplacement spécifiques des paramètres hérités du service. Le programme `ttymon` permet d'offrir des services de connexion pour ces terminaux. Chaque terminal utilise une instance distincte du programme `ttymon`. Les arguments de ligne de commande transmis par le service au programme `ttymon` régissent son comportement.

Les instances de service suivantes sont fournies avec le système :

- `svc:/system/console-login:default`

  L'instance par défaut spécifie toujours que le programme `ttymon` offre une connexion à la console matérielle du système.
- `svc:/system/console-login:{vt2, vt3, vt4, vt5, vt6}`

  D'autres instances de service sont fournies pour les consoles virtuelles du système. Si aucune console virtuelle n'est disponible, ces services sont automatiquement désactivés.

  Pour plus d'informations, reportez-vous à la page de manuel `vtdaemon`(1M).
- `svc:/system/console-login:{terma, termb}`

  Les services `svc:/system/console-login:terma` et `svc:/system/console-login:termb` sont fournis pour plus de commodité. Ces services vous aident à configurer des services de connexion pour des ports `/dev/term/a` et `/dev/term/b` supplémentaires. Ces services sont désactivés par défaut.

Vous pouvez définir d'autres instances de service dans le cadre du service `svc:system/console-login`. Par exemple, si vous avez un périphérique `/dev/term/f` à prendre en charge, vous pouvez instancier `svc:/system/console-login:termf` et le configurer de la manière appropriée.

▼ Configuration des services de connexion sur les terminaux auxiliaires

Des services prédéfinis sont fournis pour les terminaux connectés aux ports série `/dev/term/a` ou `/dev/term/b` d'un système.

1. **Prenez le rôle `root`.**

   Reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

2. **Activez l'instance du service.**

   Par exemple, pour activer les services de connexion pour `/dev/term/a` :

   ```
   # svcadm enable svc:/system/console-login:terma
   ```

3. **Vérifiez que le service est en ligne.**

   ```
   # svcs svc:/system/console-login:terma
   ```

   La sortie doit indiquer que le service est en ligne. Si le service est en mode de maintenance, consultez le fichier journal du service pour plus de détails.

▼ Définition de la vitesse de transmission en bauds sur la console

La prise en charge des vitesses de console sur les systèmes x86 dépend de la plate-forme.

Les vitesses de console suivantes sont prises en charge sur les systèmes SPARC :

- `9600 bps`
- `19200 bps`
- `38400 bps`

1. **Connectez-vous en tant qu'administrateur.**

   Reportez-vous à la section “ A l’aide de vos droits administratifs attribués ” du manuel “ Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ”.

2. **Utilisez la commande `eeprom` pour définir une vitesse de transmission en bauds adaptée à votre type de système.**

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

   Par exemple, pour faire passer la vitesse de transmission en bauds sur la console d'un système x86 à `38400`, tapez :

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

3. **Modifiez la ligne de console dans le fichier `/etc/ttydefs` comme suit :**

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

4. **Apportez les modifications supplémentaires suivantes au type de système.**

   Notez que ces modifications dépendent de la plate-forme.

   - **Sur les systèmes SPARC :** modifiez la vitesse de transmission en bauds dans la version du fichier `options.conf` qui se trouve dans le répertoire `/etc/driver/drv`. Par exemple :

Pour faire passer la vitesse de transmission en bauds à `9600` :

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

Pour faire passer la vitesse de transmission en bauds à `19200` :

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

Pour faire passer la vitesse de transmission en bauds à `38400` :

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

- **Sur les systèmes x86 :** modifiez la vitesse de la console si la redirection série du BIOS est activée.

Gestion des services d'alimentation du système

Dans le système d'exploitation Oracle Solaris 11, la configuration de la gestion de l'alimentation a été déplacée dans un référentiel de configuration SMF. La nouvelle commande `poweradm` permet de gérer directement les propriétés de gestion de l'alimentation du système, plutôt que de les gérer par le biais d'une commande liée à l'alimentation, d'un démon et d'un fichier de configuration. Ces modifications s'inscrivent dans un ensemble plus vaste de modifications destinées à moderniser le cadre de gestion d'alimentation dans le système d'exploitation Solaris 11.

Les fonctions de gestion d'alimentation suivantes ne sont plus disponibles :

- `/etc/power.conf`
- `pmconfig` et `powerd`
- Gestion de l'alimentation des périphériques

Les propriétés suivantes décrivent les composants de gestion de l'alimentation :

- `administrative-authority` : définit la source du contrôle administratif pour la gestion de l'alimentation d'Oracle Solaris. Cette propriété peut être définie sur `none`, `platform` (valeur par défaut) ou `smf`.

  Lorsqu'elle est définie sur `platform`, les valeurs `time-to-full-capacity` et `time-to-minimum-responsiveness` proviennent des commandes de gestion de l'alimentation de la plate-forme.

  Lorsqu'elle est définie sur `smf`, les valeurs `time-to-full-capacity` et `time-to-minimum-responsiveness` proviennent de l'utilitaire SMF.

Si vous tentez de définir les valeurs `time-to-full-capacity` ou `time-to-minimum-responsiveness` à partir d'une commande de plate-forme ou d'une propriété de service SMF lorsque vous vous trouvez dans l'autre dispositif, les valeurs sont ignorées.

Lorsque `administrative-authority` est défini sur `none`, la gestion de l'alimentation dans l'instance d'Oracle Solaris est désactivée.

- `time-to-full-capacity` : définit la durée maximale autorisée (en microsecondes) pour permettre au système d'atteindre sa pleine capacité à partir d'un état de capacité inférieure ou moins réactif, alors que le système est dans un état actif. La durée maximale inclut la période pendant laquelle il utilisait certaines ou toutes les fonctions PM comprises dans cette limite.

  Par défaut, cette valeur de paramètre provient de la plate-forme, `i86pc` par exemple, car le paramètre `administrative-authority` est configuré sur plate-forme par défaut.

  Sinon, lorsque `administrative-authority` est défini sur `smf`, cette valeur provient de la définition fournie par le service d'alimentation SMF. Au moment de l'installation, cette valeur n'est pas définie. Si vous choisissez de modifier cette propriété, optez pour une valeur adaptée en fonction de la charge de travail du système ou des applications.
- `time-to-minimum-responsiveness` : définit la durée autorisée en millisecondes pour permettre au système de retourner à son état actif. Ce paramètre permet d'atteindre la capacité minimale requise pour satisfaire la contrainte `time-to-full-capacity`. Par défaut, cette valeur de paramètre provient de la plate-forme, `i86pc` par exemple, car le paramètre `administrative-authority` est configuré par défaut sur plate-forme.

  Sinon, lorsque `administrative-authority` est défini sur `smf`, cette valeur provient de la définition fournie par le service d'alimentation SMF. Au moment de l'installation, cette valeur n'est pas définie. Si vous choisissez de modifier cette propriété, optez pour une valeur adaptée en fonction de la charge de travail du système ou des applications.

  Des valeurs modérées, quelques secondes par exemple, permettent de placer les composants matériels ou les sous-systèmes de la plate-forme dans des états inactifs avec des temps de réponse plus longs. Des valeurs plus importantes, de 30 secondes à quelques minutes par exemple, permettent de suspendre le système entier à l'aide de techniques telles que la mise en veille en mémoire vive.
- `suspend-enable` : par défaut, aucun système exécutant Oracle Solaris n'est autorisé à tenter une opération de suspension. Lorsque cette propriété est définie sur true, une opération de suspension peut être tentée. La valeur du paramètre `administrative-authority` n'a aucun effet sur cette propriété.
- `platform-disabled` : lorsque `platform-disabled` est défini sur `true`, la gestion de l'alimentation de la plate-forme est désactivée. Lorsque le paramètre est défini sur `false`, ce qui correspond au paramétrage par défaut, la gestion de l'alimentation est contrôlée par le biais des valeurs des propriétés ci-dessus.

Pour afficher un bref résumé de statut de la gestion de l'alimentation, exécutez la commande suivante :

```
$ /usr/sbin/poweradm show
```

```
Power management is enabled with the hardware platform as the authority:
time-to-full-capacity set to 250 microseconds
time-to-minimum-responsiveness set to 0 milliseconds
```

Pour afficher des propriétés de gestion de l'alimentation, exécutez la commande suivante :

```
$ /usr/sbin/poweradm list
active_config/time-to-full-capacity          current=250, platform=250
active_config/time-to-minimum-responsiveness current=0, platform=0
active_control/administrative-authority      current=platform, smf=platform
suspend/suspend-enable                       current=false
platform-disabled                            current=false
```

Dans cette sortie, l'élément `active_control/administrative-authority` indique l'origine de la configuration à l'aide de deux paramètres :

- `platform` : la configuration de la gestion de l'alimentation provient de la plate-forme. Il s'agit de la valeur par défaut.
- `smf` : permet aux autres propriétés de gestion de l'alimentation d'être définies à l'aide de la commande `poweradm`.

La propriété `platform-disabled` dans la sortie indique que la gestion de l'alimentation de la plate-forme est activée :

```
platform-disabled                            current=false
```

Pour plus d'informations, reportez-vous à la page de manuel `poweradm`(1M).

EXEMPLE 5-1 Activation et désactivation de la gestion de l'alimentation

Si vous avez précédemment permis à S3-support dans le fichier `/etc/power.conf` de suspendre et de reprendre votre système, la syntaxe `poweradm` similaire est la suivante :

```
# poweradm set suspend-enable=true
```

La propriété `suspend-enable` est définie sur `false` par défaut.

Utilisez la syntaxe suivante pour désactiver la gestion de l'alimentation :

```
# poweradm set administrative-authority=none
```

La désactivation du service de gestion de l'alimentation SMF suivant ne désactive pas la gestion de l'alimentation :

```
online         Sep_02   svc:/system/power:default
```

Utilisez la syntaxe suivante pour désactiver la suspension et reprendre :

```
# poweradm set suspend-enable=false
```

EXEMPLE 5-2 Configuration et affichage des paramètres de gestion de l'alimentation

L'exemple suivant indique comment configurer `time-to-full-capacity` sur 300 microsecondes et `time-to-minimum-responsiveness` sur 500 millisecondes. En dernier lieu, l'instance d'Oracle Solaris est informée des nouvelles valeurs.

```
# poweradm set time-to-full-capacity=300
# poweradm set time-to-minimum-responsiveness=500
# poweradm set administrative-authority=smf
```

La commande suivante permet d'afficher la valeur `time-to-full-capacity` actuelle.

```
# poweradm get time-to-full-capacity
300
```

La commande suivante récupère la valeur `time-to-full-capacity` définie par la plate-forme.

```
# poweradm get -a platform time-to-full-capacity
```

Notez que cette valeur est uniquement identique à la valeur en cours si `administrative-authority` est défini sur plate-forme. Pour plus d'informations, reportez-vous à la description de la propriété `administrative-authority` donnée plus haut.

▼ Récupération à partir du service d'alimentation en mode de maintenance

Si `administrative-authority` est défini sur `smf` avant que `time-to-full-capacity` et `time-to-minimum-responsiveness` n'aient été définis, le service passe en mode de maintenance. Reportez-vous à la tâche ci-dessous pour effectuer une récupération suite à ce scénario.

1. **Connectez-vous en tant qu'administrateur.**

   Reportez-vous à la section " A l'aide de vos droits administratifs attribués " du manuel " Sécurisation des utilisateurs et des processus dans Oracle Solaris 11.2 ".

2. **Définissez `administrative-authority` sur `none`.**

   ```
   # poweradm set administrative-authority=none
   ```

3. **Définissez `time-to-full-capacity` et `time-to-minimum-responsiveness` sur les valeurs souhaitées.**

   ```
   # poweradm set time-to-full-capacity=value
   # poweradm set time-to-minimum-responsiveness=value
   ```

4. **Effacez le service.**

```
# svcadm clear power
```

5. **Définissez `administrative-authority` sur `smf`.**

```
# poweradm set administrative-authority=smf
```

Index

D

E

F

N

P

R

S

T

U

V