

# ProcessSaver テンプレートガイド

第 3 版

2013年 4月

日本電気株式会社

HP-UX 版  
Linux 版  
Solaris 版

# 目次

## はじめに 本資料の目的 導入事例

- HP-UX 編
  - syslogデーモンプロセスの監視事例
  - 代表的なOS標準デーモンの一括監視事例
  - グループ監視の導入事例
  - 同一名プロセスの監視事例
  - Oracle の監視事例
  - WebSAM MCOperations / SystemManager / NetvisorProV の監視事例
  - WebSAM JobCenter の監視事例
  - WebSAM NetWorker の監視事例
  - WebSAM DeliveryManager の監視事例
- Linux 編
  - syslogデーモンプロセスの監視事例
  - 代表的なOS標準デーモンの一括監視事例
  - Apache HTTP Server の監視事例
  - PostgreSQL の監視事例
  - JBoss の監視事例
  - Apache Tomcat の監視事例
  - WebSAM JobCenter の監視事例
  - ESMPRO/ServerAgent の監視事例
  - StoragePathSavior の監視事例
  - WebSAM MCOperations / ApplicationNavigator の監視事例
  - HULFT の監視事例
- Solaris 編
  - syslogデーモンプロセスの監視事例
  - Apache HTTP Serverの監視事例

## 商標

# はじめに

本製品は以下のオペレーティングシステムに対応しています。

- HP-UX版
  - HP-UX 11.00 (PA-RISC)
  - HP-UX 11i v1 (PA-RISC)
  - HP-UX 11i v2 (PA-RISC / Itanium)
  - HP-UX 11i v3 (Itanium)
- Linux版
  - Red Hat Linux (IA32 および EM64T搭載マシンにも対応)
    - Red Hat Enterprise Linux AS 3.0 / 4.0
    - Red Hat Enterprise Linux ES 3.0 / 4.0
    - Red Hat Enterprise Linux 5
    - Red Hat Enterprise Linux 6
  - Oracle Linux (IA32 および EM64T搭載マシンにも対応)
    - Oracle Enterprise Linux 5
    - Oracle Enterprise Linux 6
  - SUSE Linux (IA32 搭載マシンに対応)
    - SUSE Linux Enterprise 9~11
  - MIRACLE LINUX (IA32 搭載マシンに対応)
    - MIRACLE LINUX V3.0
  - Turbolinux (IA32 搭載マシンに対応)
    - Turbolinux 10 Server
  - Asianux (IA32 搭載マシンに対応)
    - Asianux 2.0
  - CentOS (IA32 搭載マシンに対応)
    - CentOS 4
- Solaris版
  - Solaris 7
  - Solaris 8
  - Solaris 9
  - Solaris 10
  - Solaris 11
  - Open Solaris

本書は、HP-UX版/Linux版/Solaris版共通のドキュメントとなっています。

# 本資料の目的

## 本資料の目的

ProcessSaver によるプロセス監視ソリューションを支援するための導入事例をご紹介します。  
本資料でご紹介する事例は、監視対象となる製品の SG 設定・運用スタイル、バージョンの差異、さらにパッチ等によって、監視手順に変更が発生する場合があります。

実際に導入する場合は、十分な検証と動作確認を実施願います。

なお、本事例はクラスタソフトを含めた連携手順を記載しているものではありません。

また本資料では、監視対象ごとに pfile を作成していますが、システム構成に応じてアプリケーションのグループごとに pfile を統合することを推奨します。

また、本資料をご利用になる前に、ユーザーズガイドを一読されることをお勧めします。

ProcessSaver (HP-UX 版) をご利用の方

→ HP-UX 編をご覧ください。 ( 5ページ～ )

ProcessSaver (Linux 版) をご利用の方

→ Linux 編をご覧ください。 ( 23ページ～ )

ProcessSaver (Solaris 版) をご利用の方

→ Solaris 編をご覧ください。 ( 45ページ～ )

## 導入事例

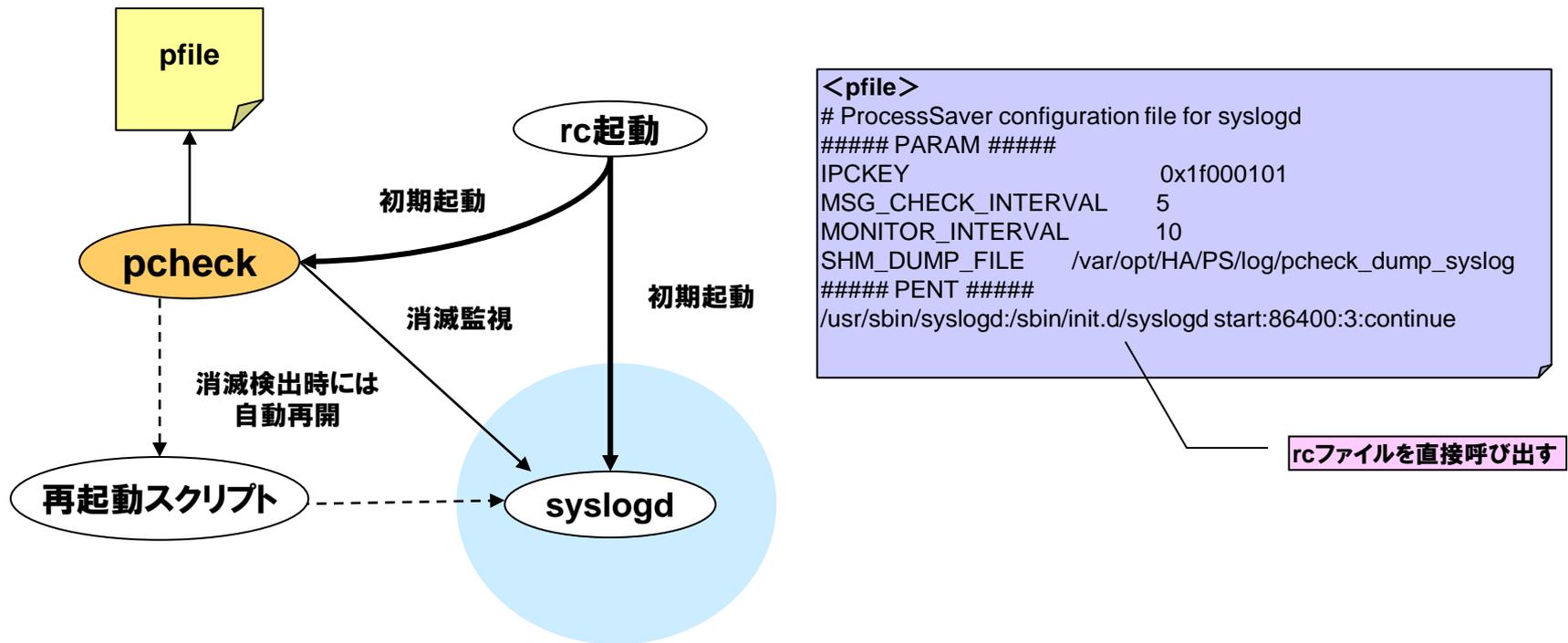
- **運用環境に合わせた導入例を紹介します。**
  - (1) **syslog デーモンプロセスの監視事例**
  - (2) **代表的なOS標準デーモンの一括監視事例**
  - (3)~(4) **グループ監視の導入事例**
  - (5)~(7) **同一名プロセスの監視事例**
  - (8) **Oracle の監視事例**
  - (9) **WebSAM MCOperations / SystemManager / NetvisorPro V の監視事例**
  - (10) **WebSAM JobCenter の監視事例**
  - (11) **WebSAM NetWorker の監視事例**
  - (12) **WebSAM DeliveryManager の監視事例**

# (1) syslog デーモンプロセスの監視事例

## syslogd の消滅監視事例

### 監視手順

- ノード立ち上げ契機で syslog デーモンを起動します。
- pcheck により消滅監視を実行します。
- 消滅検出時には、再起動スクリプトで再開します。



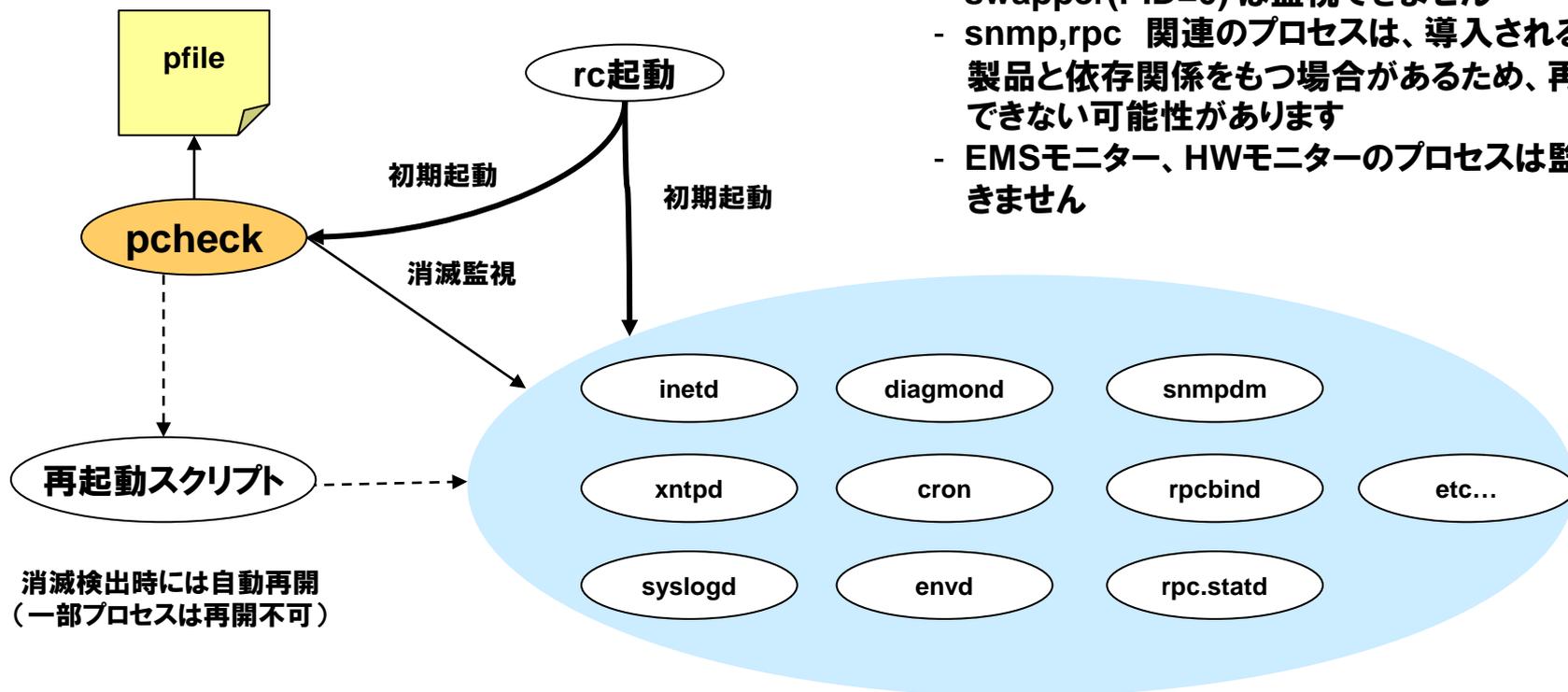
## (2) 代表的なOS標準デーモンの一括監視事例

### 監視手順

- rcファイルから各種デーモンを起動します。
- pcheck による消滅監視を実行します。
- 消滅検出時には、再起動スクリプトで再開します。

### 注意事項

- swapper(PID=0) は監視できません
- snmp,rpc 関連のプロセスは、導入されるPP製品と依存関係をもつ場合があるため、再起動できない可能性があります
- EMSモニター、HWモニターのプロセスは監視できません



# (続き)

```
<pfile>
# ProcessSaver configuration file for HP-UX
##### PARAM #####
IPCKEY          0x1f000102
MSG_CHECK_INTERVAL 5
MONITOR_INTERVAL 10
SHM_DUMP_FILE   /var/opt/HA/PS/log/pcheck_dump_HPUX

##### PENT #####

##### HP-UX 11.11 & 11.23 & 11.31 daemon
init:-:0:0:continue
vhand:-:0:0:continue
statdaemon:-:0:0:continue
unhashdaemon:-:0:0:continue          ##### HP-UX 11.11 & 11.23 & 11.31 daemon (restart)
#supsched:-:0:0:continue             /usr/sbin/syslogd:/sbin/init.d/syslogd start:86400:3:continue
strmem:-:0:0:continue                /usr/sbin/envd:/sbin/init.d/envd start:86400:3:continue
strweld:-:0:0:continue                /usr/sbin/cron:/sbin/init.d/cron start:86400:3:continue
strfreebd:-:0:0:continue              #/usr/sbin/stm/uut/bin/sys/diagmond:/PROJECT/HAPS/hpux/__diag_rst.sh:86400:3:continue
ttisr:-:0:0:continue                  #/usr/sbin/inetd:/sbin/init.d/inetd start:86400:3:continue
lvmkd:-:0:0:continue                  # SNMP Master Agent for 11.11 & 11.23 & 11.31
smpsched:-:0:0:continue                #/usr/sbin/snmpdm:/PROJECT/HAPS/hpux/__snmp_rst.sh:86400:3:continue
sblksched:-:0:0:continue
/opt/dce/sbin/rpcd:-:0:0:continue
/usr/sbin/rpcbind:-:0:0:continue
/usr/sbin/syncer:0:0:0:continue
```

# (3) グループ監視の導入事例1

## pcheck を階層的に使用しグループ監視する事例

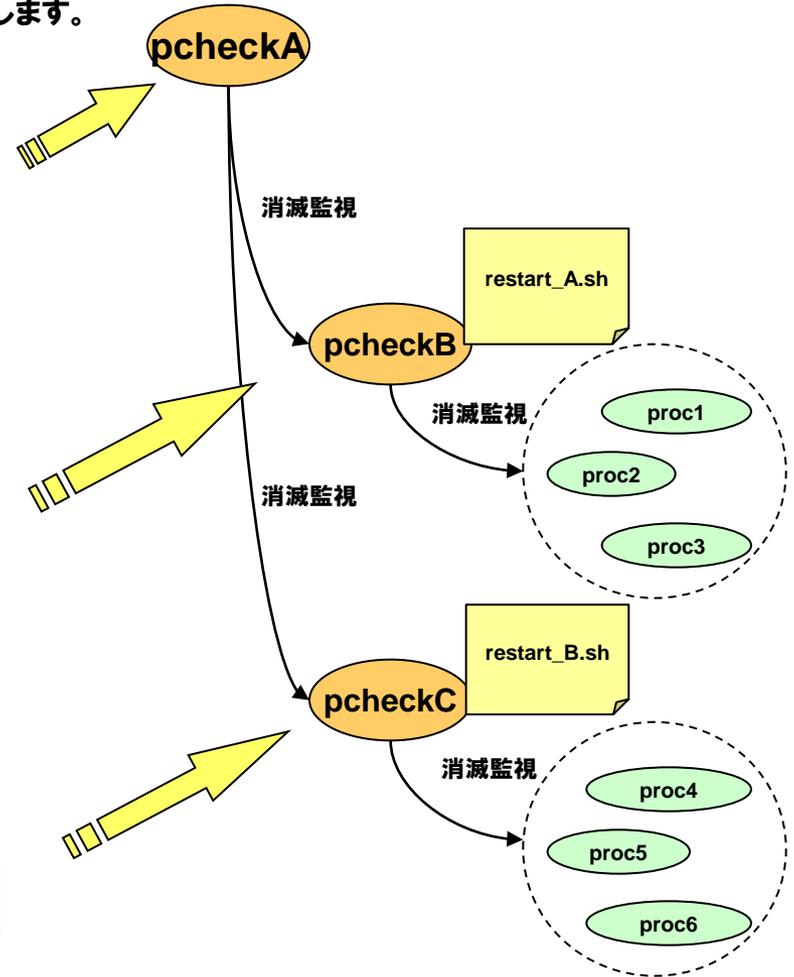
- pcheckA が配下の pcheckB、pcheckC の消滅を監視します。
- pcheckB、pcheckC はそれぞれのプロセス群の消滅を監視します。

```
<pfile pcheckA>
##### PARAM #####
IPCKEY                0x1f000301
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheckA_dump
##### PENT #####
pcheckB:/var/opt/HA/PS/conf/restart_A.sh:86400:3:continue
pcheckC:/var/opt/HA/PS/conf/restart_B.sh:86400:3:continue

<pfile pcheckB>
##### PARAM #####
IPCKEY                0x1f000302
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheckB_dump
##### PENT #####
proc1:-:0:0:shutdown
proc2:-:0:0:shutdown
proc3:-:0:0:shutdown

<pfile pcheckC>
##### PARAM #####
IPCKEY                0x1f000303
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheckC_dump
##### PENT #####
proc4:-:0:0:shutdown
proc5:-:0:0:shutdown
proc6:-:0:0:shutdown
```

子のpfileのリトライオーバーアクションは必ず shutdown を指定する



# (4) グループ監視の導入事例2

## pfile にグループ名を設定し監視する事例

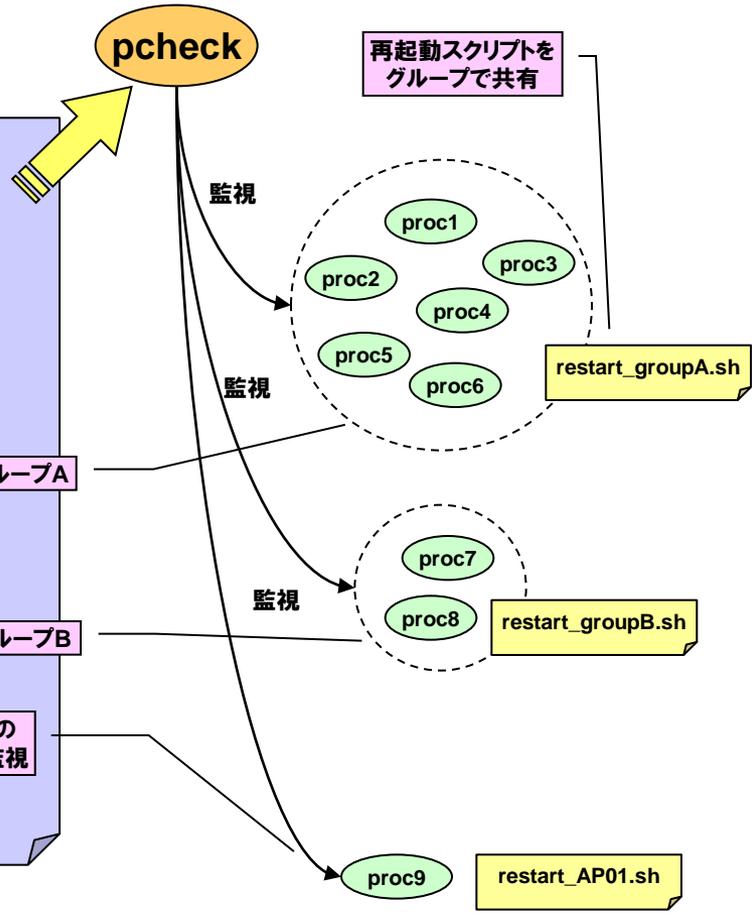
pfileのoptionに、`grouptag=<グループ名>`を指定することで、同じグループ名を持つ監視対象プロセスはプロセスグループとして扱われ、グループ内のプロセスの消滅→再起動に連動して再起動します。  
また、グループ内の複数のプロセスが同時に消滅した場合でも再起動スクリプトが競合することはありません。

```
<pfile>
##### PARAM #####
IPCKEY                0x1f000104
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     10
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_dump

##### PENT #####
# groupA
proc1:/var/opt/HA/PS/conf/restart_groupA.sh:86400:3:continue:grouptag=groupA
proc2:/var/opt/HA/PS/conf/restart_groupA.sh:86400:3:continue:grouptag=groupA
proc3:/var/opt/HA/PS/conf/restart_groupA.sh:86400:3:continue:grouptag=groupA
proc4:/var/opt/HA/PS/conf/restart_groupA.sh:86400:3:continue:grouptag=groupA
proc5:/var/opt/HA/PS/conf/restart_groupA.sh:86400:3:continue:grouptag=groupA
proc6:/var/opt/HA/PS/conf/restart_groupA.sh:86400:3:continue:grouptag=groupA

# groupB
proc7:/var/opt/HA/PS/conf/restart_groupB.sh:86400:3:continue:grouptag=groupB
proc8:/var/opt/HA/PS/conf/restart_groupB.sh:86400:3:continue:grouptag=groupB

# AP 01
proc9:/var/opt/HA/PS/conf/restart_AP01.sh:86400:3:continue
```



グループ名指定

グループA

グループB

従来の  
単独監視

再起動スクリプトを  
グループで共有

restart\_groupA.sh

restart\_groupB.sh

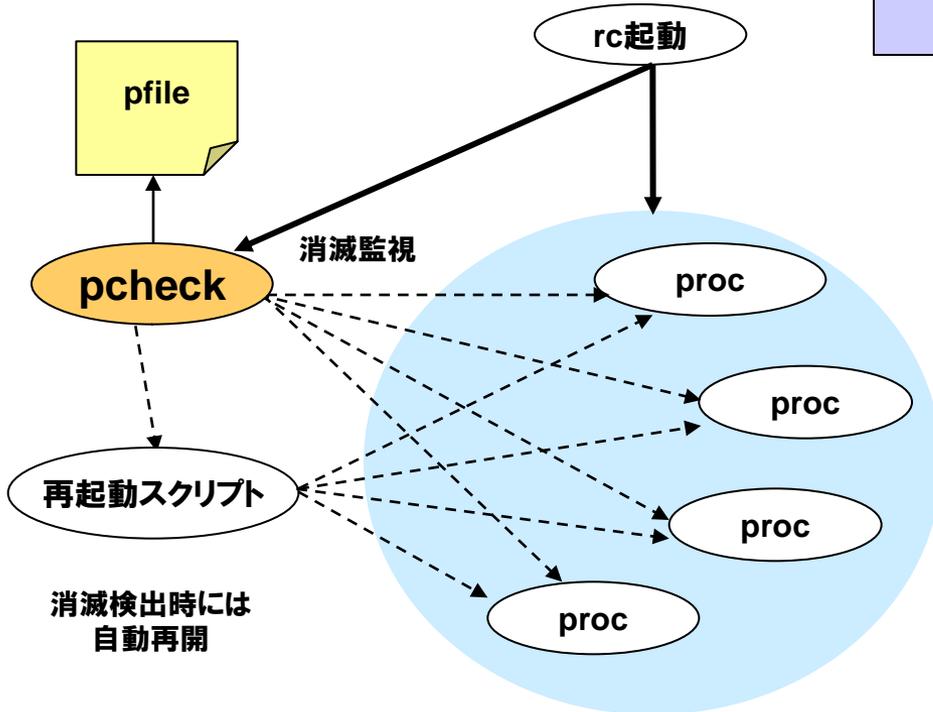
restart\_AP01.sh

# (5) 同一名プロセスの監視事例1

## 同一名プロセスを起動数で監視する事例

### 監視手順

- 同じ名前のプロセスを起動数で監視します。
- 記述された個数未満(この例では4個)になるとプロセスの消滅を検知して、再起動スクリプトを実行します。



```
<pfile>
# ProcessSaver configuration file for sample
##### PARAM #####
IPCKEY                0x1f000006
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     10
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_dump
##### PENT #####
# watch 4 counts proc
proc:/var/opt/HA/PS/conf/restart_proc.sh:86400:3:continue:min_proc_count=4
```

同一プロセスを監視する個数を記載

### 引数の指定方式

- pfileのプロセス監視エントリのoption情報に min\_proc\_count を指定します。

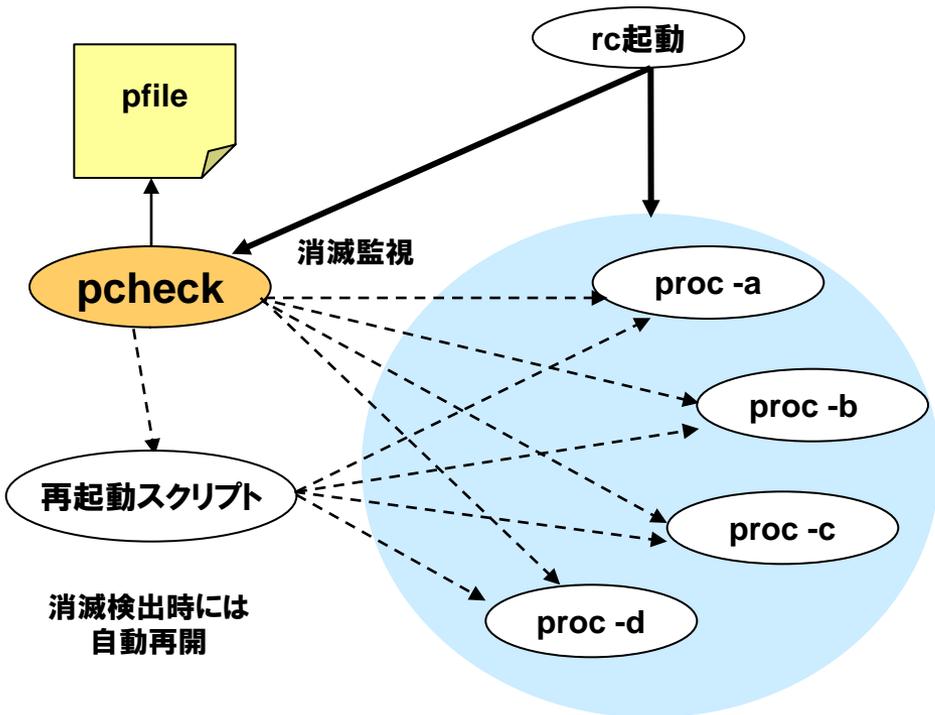
(※)ただし、監視対象プロセス自身が消滅すると、プロセス数が min\_proc\_count 以上の場合でも消滅を検知します。

# (6) 同一名プロセスの監視事例2

## 同一名プロセスを引数で区別する事例

### 監視手順

- 監視対象プロセスを引数によって区別できる場合、引数付きでプロセスを監視します。
- プロセス毎に再起動スクリプトを分けることが可能となります。



```
<pfile>
# ProcessSaver configuration file for sample
##### PARAM #####
IPCKEY                0x1f000107
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheck_dump_sample
##### PENT #####
# watch 4 counts proc
/usr/bin/proc -a:/usr/bin/sample1.sh:86400:3:continue
/usr/bin/proc -b:/usr/bin/sample2.sh:86400:3:continue
/usr/bin/proc -c:/usr/bin/sample3.sh:86400:3:continue
/usr/bin/proc -d:/usr/bin/sample4.sh:86400:3:continue
```

引数つきでプロセス名を指定

### 引数の指定方式

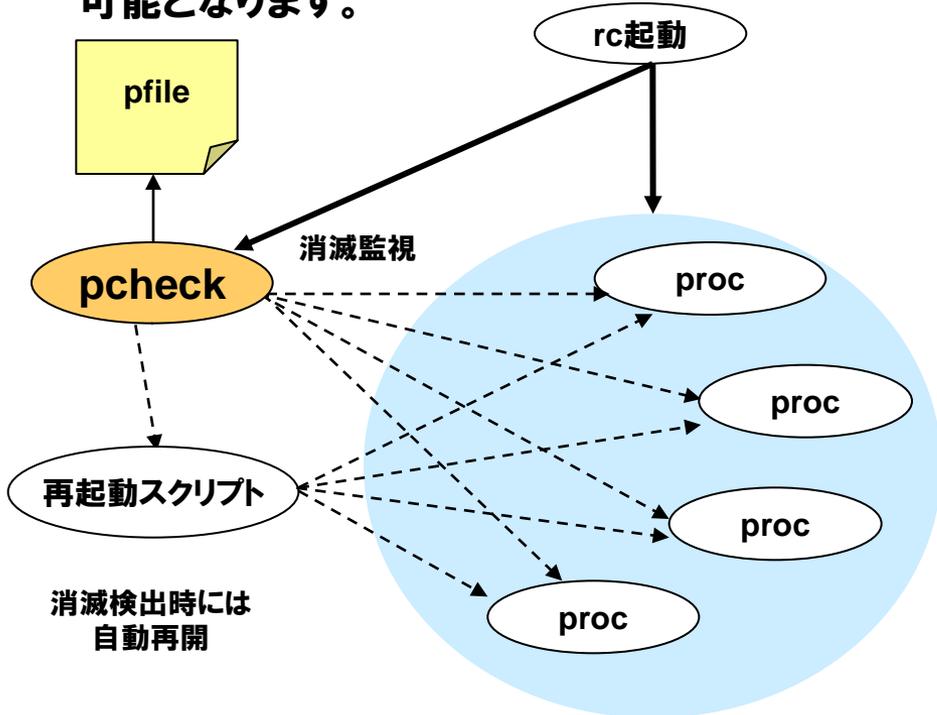
- pfileのプロセス監視エントリのプロセス名と引数を記述します。
- 引数は最大で64文字まで指定可能です。
- 指定された引数を含むプロセス名が検査対象となります。

# (7) 同一名プロセスの監視事例3

## 同一名プロセスをユーザ情報で区別する事例

### 監視手順

- 監視対象プロセスの起動ユーザが異なる場合、ユーザ情報( uidまたはuname)をキーにプロセスを区別します。
- プロセス毎に再起動スクリプトを分けることが可能となります。



```
<pfile>
# ProcessSaver configuration file for sample
##### PARAM #####
IPCKEY                0x1f000107
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheck_dump_sample
##### PENT #####
# watch 4 counts proc
/usr/bin/proc:/usr/bin/sample1.sh:86400:3:continue:uid=501
/usr/bin/proc:/usr/bin/sample2.sh:86400:3:continue:uid=502
/usr/bin/proc:/usr/bin/sample3.sh:86400:3:continue:uname=oracle
/usr/bin/proc:/usr/bin/sample4.sh:86400:3:continue:uname=guest
```

uid または uname  
を指定

### 引数の指定方式

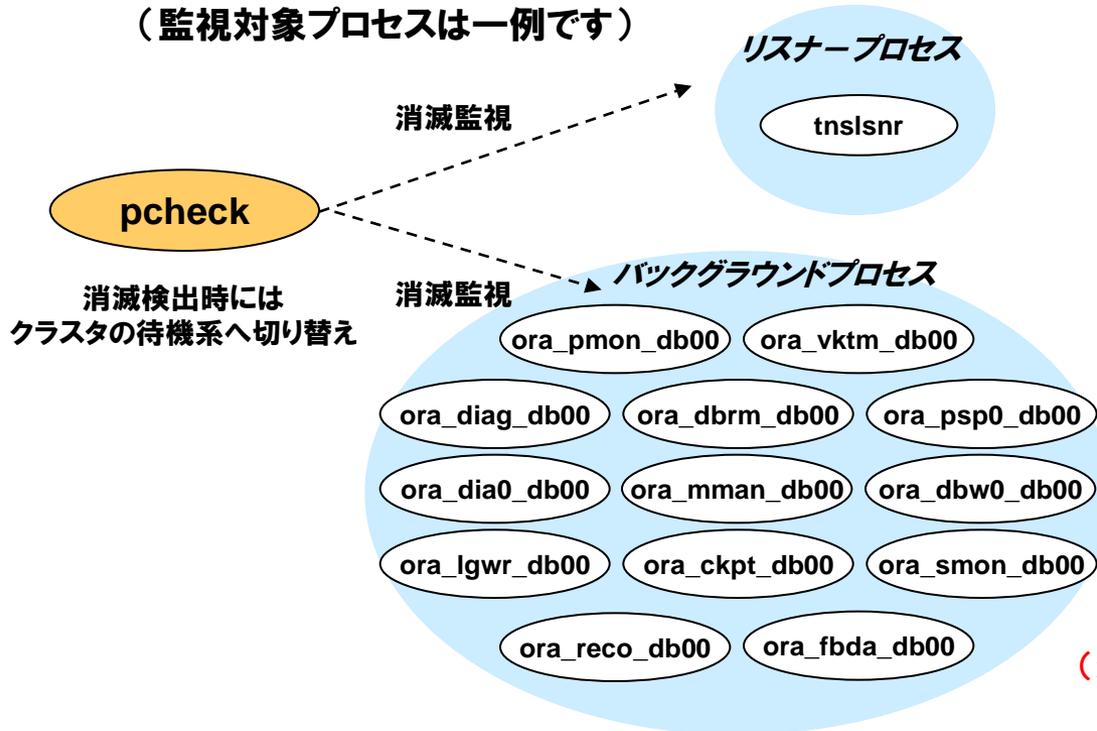
- pfileのプロセス監視エントリのoption情報に uid(またはuname)を指定します。

# (8) Oracle の監視事例

## Oracle 11g R1 の消滅監視事例

### 監視手順

- pcheck により Oracle のプロセスを監視します。
- 監視対象とするプロセス名はデータベース名により異なります。
- 本事例はクラスタ構成において、Oracle プロセスの消滅時に待機系への切り換えを行う事例となります。  
(監視対象プロセスは一例です)



```
<pfile>
# ProcessSaver configuration file for Oracle
##### PARAM #####
IPCKEY                0x10003001
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheck_dump

##### PENT #####
# Background Process
ora_pmon_db00--:0:0:exit
ora_vktm_db00--:0:0:exit
ora_diag_db00--:0:0:exit
ora_dbrm_db00--:0:0:exit
ora_psp0_db00--:0:0:exit
ora_dia0_db00--:0:0:exit
ora_mman_db00--:0:0:exit
ora_dbw0_db00--:0:0:exit
ora_lgwr_db00--:0:0:exit
ora_ckpt_db00--:0:0:exit
ora_smon_db00--:0:0:exit
ora_reco_db00--:0:0:exit
ora_fbda_db00--:0:0:exit

# Listener
/opt/app/oracle/product/11.1.0/db_1/bin/tnslsnr--:0:0:exit
```

(注) pcheck による Oracle プロセスの監視については消滅監視のみとなります。(ストール監視等は不可) そのため Oracle の特性を理解した監視を行う場合は、HA/ApplicationMonitor for Oracle を導入して監視を行うことを推奨しております。

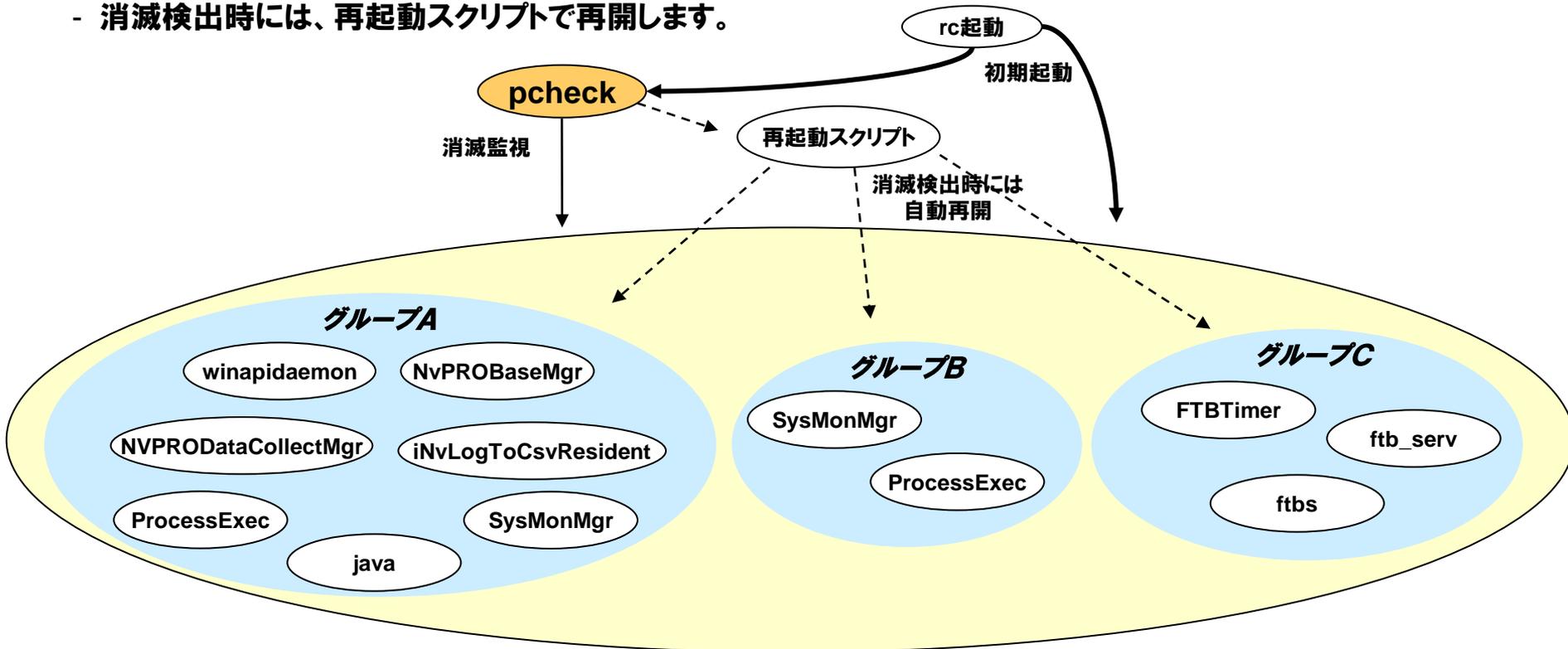
# (9) WebSAM MCOperations/SystemManager/NetvisorPro V の監視事例

## 以下製品導入時の消滅監視事例

- WebSAM MCOperations 3.5.1.1 (マネージャ)
- WebSAM SystemManager 5.4.1.1 (マネージャ)
- WebSAM NetvisorPro V 4.0.0.39

## 監視手順

- pcheck により各プロセスを監視します。  
※各プロセスに依存関係があり消滅時にはまとめて再起動する必要があるため、グループ監視を行います。
- 消滅検出時には、再起動スクリプトで再開します。



# (続き)

```
<pfile>
##### PARAM #####
IPCKEY                0x3f000005
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     10
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_manager_dump
##### PENT #####
/opt/UMF/Operations/Manager/bin/winapidaemon:/var/opt/HA/PS/conf/bin/UMFOperations_restart.sh:86400:3:continue:grouptag=UMFOperations
/opt/UMF/Operations/Manager/bin/NVPRODataCollectMgr:/var/opt/HA/PS/conf/bin/UMFOperations_restart.sh:86400:3:continue:grouptag=UMFOperations
/opt/UMF/Operations/Manager/bin/ProcessExec:/var/opt/HA/PS/conf/bin/UMFOperations_restart.sh:86400:3:continue:grouptag=UMFOperations
/opt/java1.5/jre/bin/IA64N/java -Xmx512m -cp /opt/UMF/Operations/Manager/bin/Nvplog.jar%:/opt/UMF/Operations/Manager/bin/NvPROrm:/var/opt/HA/PS/conf/bin/UMFOperations_restar
t.sh:86400:3:continue:grouptag=UMFOperations
/opt/UMF/Operations/Manager/bin/SysMonMgr:/var/opt/HA/PS/conf/bin/UMFOperations_restart.sh:86400:3:continue:grouptag=UMFOperations
/opt/UMF/Operations/Manager/bin/NvPROBaseMgr:/var/opt/HA/PS/conf/bin/UMFOperations_restart.sh:86400:3:continue:grouptag=UMFOperations
/opt/UMF/Operations/Manager/bin/iNvLogToCsvResident:/var/opt/HA/PS/conf/bin/UMFOperations_restart.sh:86400:3:continue:grouptag=UMFOperations
/opt/UMF/FWBASE/Manager/bin/SysMonMgr:/var/opt/HA/PS/conf/bin/UMFBase_restart.sh:86400:3:continue:grouptag=UMFBaseManager
/opt/UMF/FWBASE/Manager/bin/ProcessExec:/var/opt/HA/PS/conf/bin/UMFBase_restart.sh:86400:3:continue:grouptag=UMFBaseManager
/opt/SS/FTBase/common/bin/FTBTimer:/var/opt/HA/PS/conf/bin/ftb_restart.sh:86400:3:continue:grouptag=ftb_daemon
/opt/SS/FTBase/bin/ftb_serv:/var/opt/HA/PS/conf/bin/ftb_restart.sh:86400:3:continue:grouptag=ftb_daemon
/opt/SS/FTBase/bin/ftbs:/var/opt/HA/PS/conf/bin/ftb_restart.sh:86400:3:continue:grouptag=ftb_daemon
```

## <再起動スクリプト (UMFOperations\_restart.sh) の例>

```
#!/bin/sh

## UMFOperationsManager stop
/sbin/init.d/UMFOperationsManager_1 stop
/usr/bin/sleep 10

## UMFOperationsManager start
/sbin/init.d/UMFOperationsManager_1 start
/usr/bin/sleep 10

exit 0
```

## <再起動スクリプト (UMFBase\_restart.sh) の例>

```
#!/bin/sh

## UMFBaseManager stop
/sbin/init.d/UMFBaseManager_2 stop
/usr/bin/sleep 10

## UMFBaseManager start
/sbin/init.d/UMFBaseManager_2 start
/usr/bin/sleep 10

exit 0
```

## <再起動スクリプト (ftb\_restart.sh) の例>

```
#!/bin/sh

## ftb daemon stop
/opt/SS/FTBase/bin/ftb_daemon stop
/usr/bin/sleep 10

## ftb daemon start
/opt/SS/FTBase/bin/ftb_daemon start
/usr/bin/sleep 10

exit 0
```

# (続き)

## 以下製品導入時の消滅監視事例

- WebSAM MCOperations 3.5.1.1 (エージェント)
- WebSAM SystemManager 5.4.1.1 (エージェント)

## 監視手順

- pcheck により各プロセスを監視します。
  - ※SysMonAgt、ProcessExec プロセスは、依存関係があり、消滅時にはまとめて再起動する必要があるため、グループ監視を行います。
- 消滅時には、再起動スクリプトで再開します。

### <再起動スクリプト (UMFOperationsAgt\_restart.sh) の例>

```
#!/bin/sh

## UMFOperationsAgent stop
/sbin/init.d/UMFOperationsAgent_1 stop
/usr/bin/sleep 10

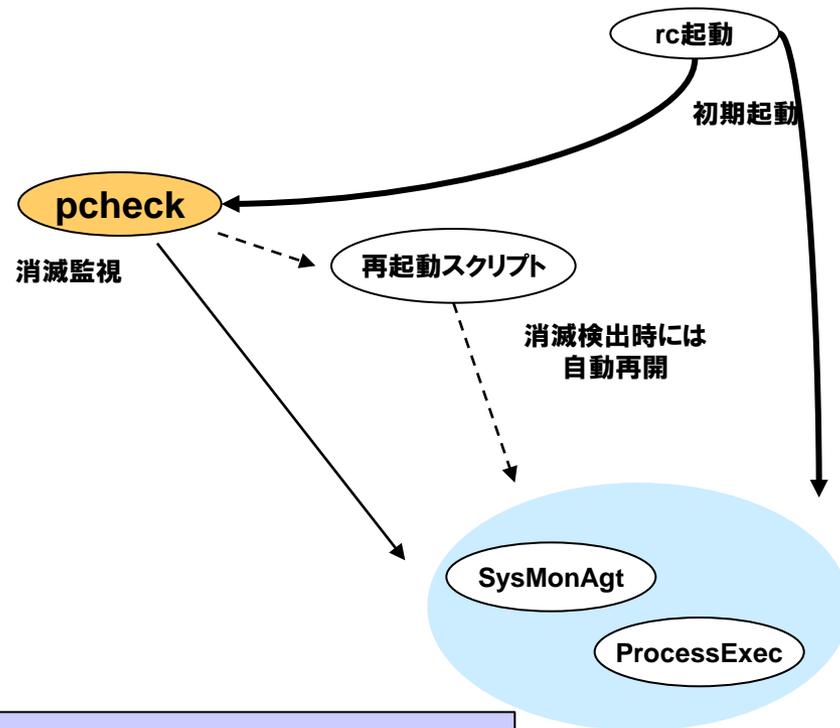
## UMFOperationsAgentstart
/sbin/init.d/UMFOperationsAgent_1 start
/usr/bin/sleep 10

exit 0
```

### <pfile>

```
##### PARAM #####
IPCKEY                               0x3f000005
MSG_CHECK_INTERVAL                    5
MONITOR_INTERVAL                      10
SHM_DUMP_FILE                         /var/opt/HA/PS/log/pcheck_agent_dump
##### PENT #####
/opt/UMF/Operations/Agent/bin/SysMonAgt:/var/opt/HA/PS/conf/bin/UMFOperationsAgt_restart.sh:86400:3:continue:grouptag=UMFOperationsAgt
/opt/UMF/Operations/Agent/bin/ProcessExec:/var/opt/HA/PS/conf/bin/UMFOperationsAgt_restart.sh:86400:3:continue:grouptag=UMFOperationsAgt
```

- (注) WebSAM MCOperations と SystemManager は、共通のエージェントプロセスを使用しています。
- (注) ご利用の環境によって監視対象プロセス名、再起動手順が異なる場合があります。

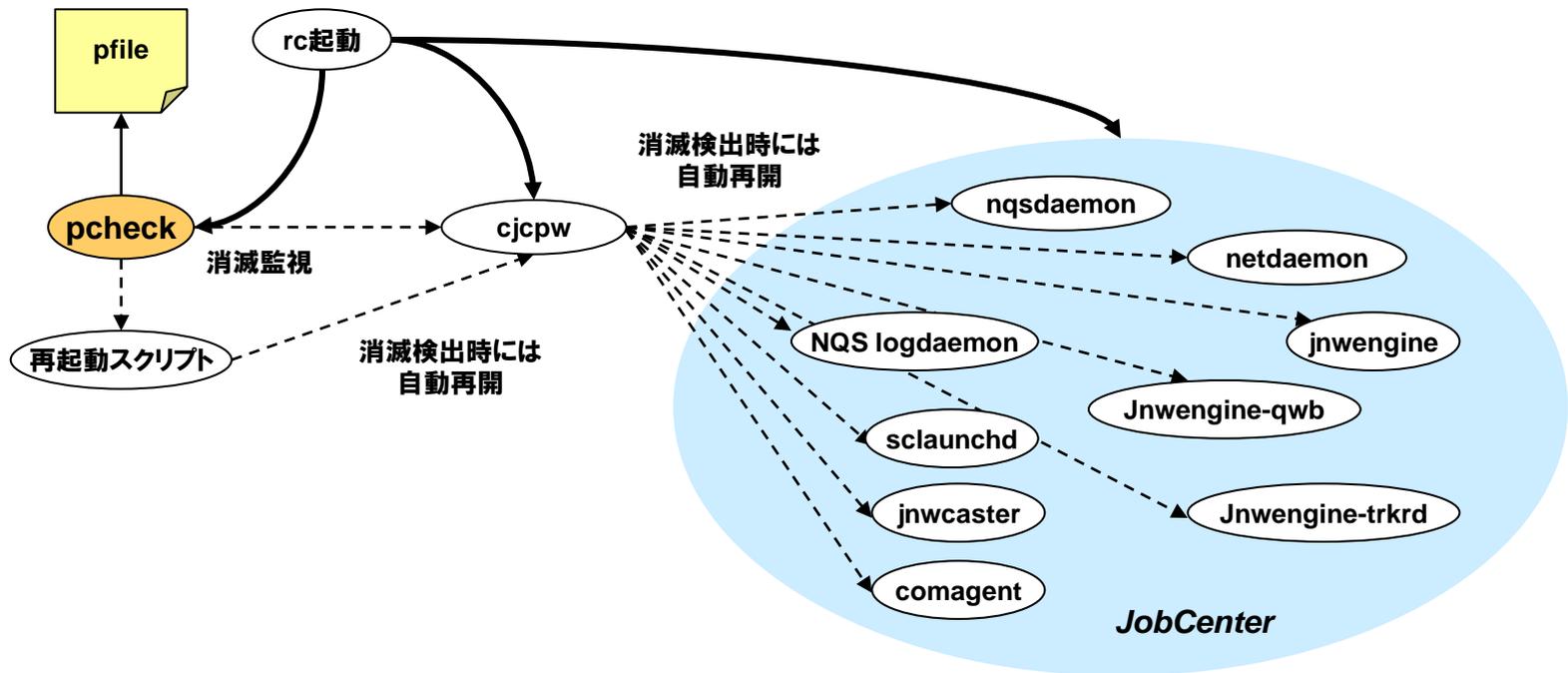


# (10) WebSAM JobCenter の監視事例

## WebSAM JobCenter 12.9 の消滅監視事例

### 監視手順

- pcheck が JobCenter の管理デーモン(cjcpw)を監視
- cjcpw が各 JobCenter のプロセスを監視
  - ※ JobCenter の他プロセスは cjcpw デーモンによって消滅監視が行われているため、pcheck で監視を行う必要はありません。



# (続き)

```
<pfile>
##### PARAM #####
IPCKEY                0x3f000008
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     10
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_JobCenter_dump
##### PENT #####
/usr/lib/nqs/cluster/cjcpw:/var/opt/HA/PS/conf/bin/restart_JC.sh:86400:3:exit
```

## <再起動スクリプト (/var/opt/HA/PS/conf/bin/restart\_JC.sh) の例>

```
#!/bin/sh

## JobCenter stop
/usr/lib/nqs/cluster/cjcpw -stop <サイト名> } JobCenterの停止処理
/usr/bin/sleep 10

## JobCenter start
/usr/lib/nqs/cluster/cjcpw <サイト名> /home/<サイト名> & } JobCenterの起動処理
/usr/bin/sleep 10

exit 0
```

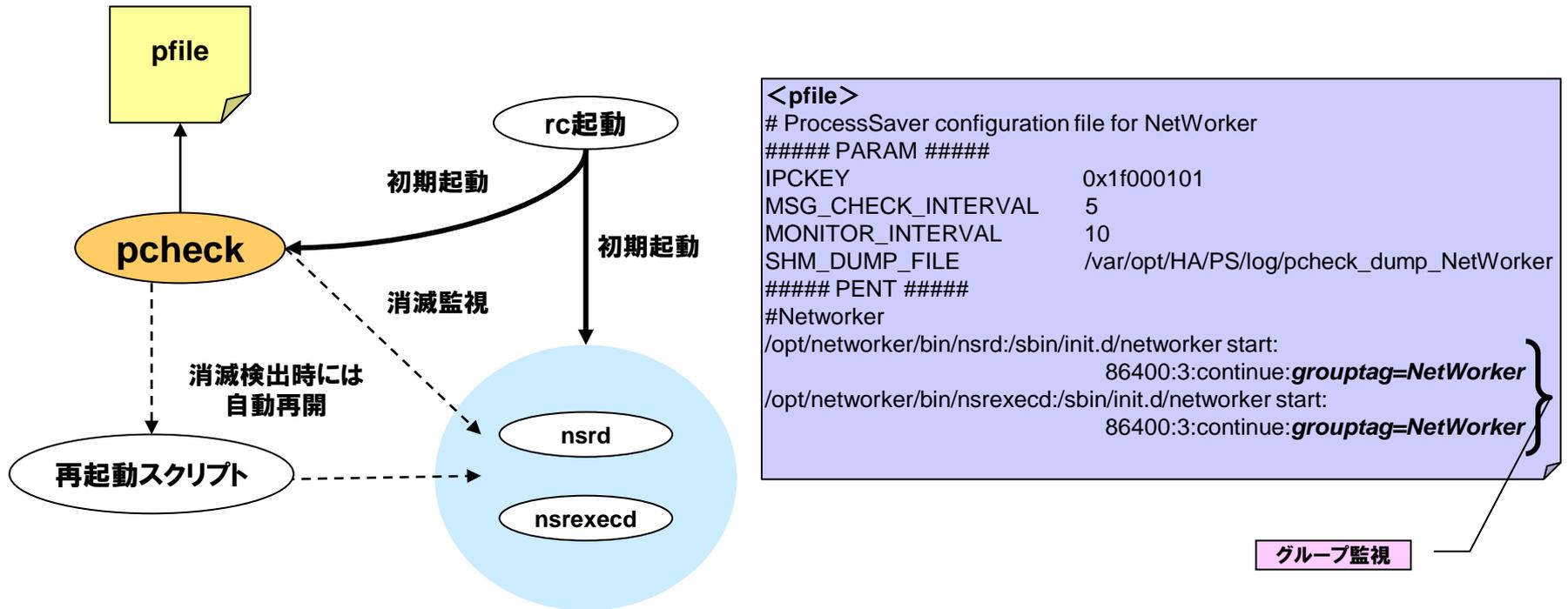
(注) ご利用の環境によって、監視対象プロセス名、再起動手順が変更となる場合があります。  
本設定はクラスタ環境の場合の監視事例となります。

# (11) WebSAM NetWorker の監視事例

## WebSAM NetWorker 7.2.1 の消滅監視事例

### 監視手順

- pcheck により NetWorker のプロセスをグループ監視します。  
NetWorker の nsrd、nsrexecd プロセスは依存関係があるためグループ監視を行います。
- 消滅検出時には、再起動スクリプトでまとめて再開します。



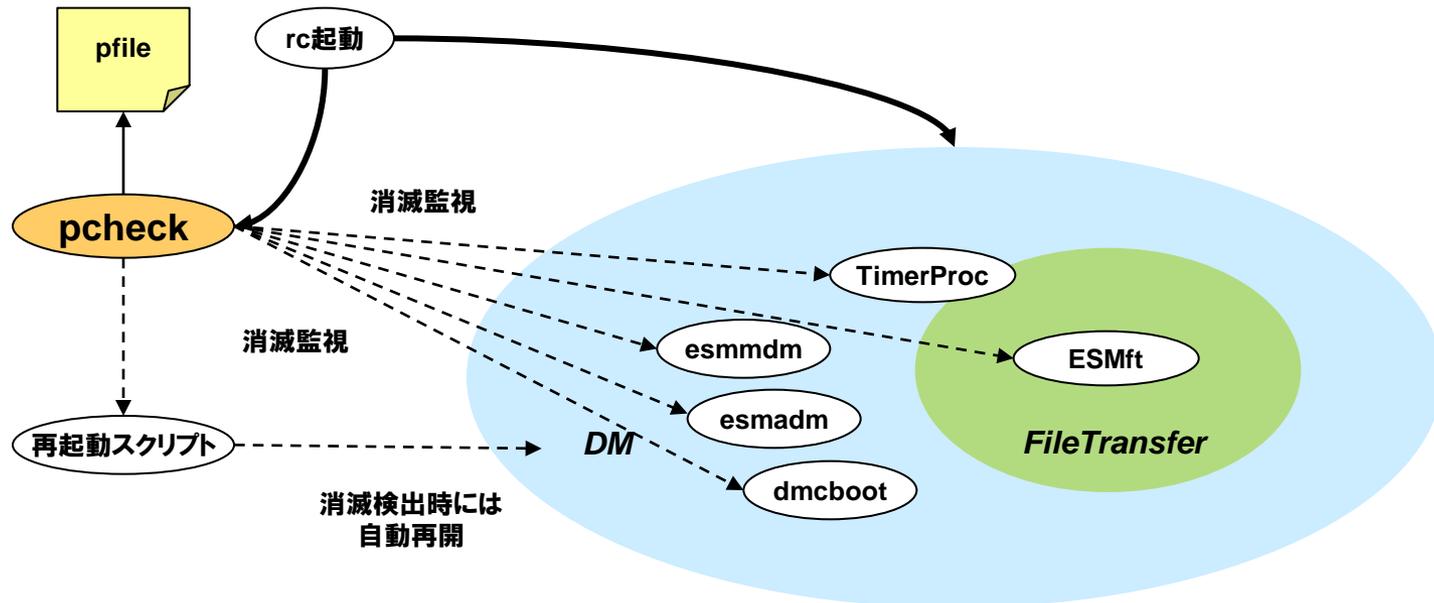
(注) ご利用の環境によって監視対象プロセス名、再起動手順が異なる場合があります。

# (12) WebSAM DeliveryManager の監視事例

## WebSAM DeliveryManager Standard Edition 8.2 の消滅監視手順

### 監視手順

- pcheck により DeliveryManager の各プロセスをグループ監視します。  
※DeliveryManager の各プロセスは依存関係があるためグループ監視を行います。
- 消滅検出時には、再起動スクリプトでまとめて再開します。



(注) ご利用の環境によって監視対象プロセス名、再起動手順が異なる場合があります。

# (続き)

## <pcheck の pfile例>

```
##### PARAM #####
IPCKEY          0x3f000017
MSG_CHECK_INTERVAL 5
MONITOR_INTERVAL 10
SHM_DUMP_FILE   /var/opt/HA/PS/log/pcheck_dump_DM

##### PENT #####
## WebSAM DeliveryManager
/opt/SS/DeliveryManager/common/bin/TimerProc:/var/opt/HA/PS/conf/bin/restart_DM.sh:86400:3:continue:grouptag=DeliveryManager
/opt/SS/FileTransfer/bin/ESMft:/var/opt/HA/PS/conf/bin/restart_DM.sh:86400:3:continue:grouptag=DeliveryManager
/opt/SS/DeliveryManager/bin/esmmdm:/var/opt/HA/PS/conf/bin/restart_DM.sh:86400:3:continue:grouptag=DeliveryManager
/opt/SS/DeliveryManager/bin/esmadm:/var/opt/HA/PS/conf/bin/restart_DM.sh:86400:3:continue:grouptag=DeliveryManager
/opt/SS/DeliveryManager/bin/dmcbboot:/var/opt/HA/PS/conf/bin/restart_DM.sh:86400:3:continue:grouptag=DeliveryManager
```

## <再起動スクリプト (/var/opt/HA/PS/conf/bin/restart\_DM.sh)の例>

```
#!/usr/bin/sh
PROC_LIST=" ¥
    /opt/SS/DeliveryManager/common/bin/TimerProc ¥
    /opt/SS/FileTransfer/bin/ESMft ¥
    /opt/SS/DeliveryManager/bin/esmmdm ¥
    /opt/SS/DeliveryManager/bin/esmadm ¥
    /opt/SS/DeliveryManager/bin/dmcbboot"

# DeliveryManager process kill
for i in $PROC_LIST
do
    pid=`ps -ef | awk -v tgt=$i '$8 == tgt {print $2}'`
    if [ -n "$pid" ]
    then
        /usr/bin/kill -15 $pid
    fi
done
/usr/bin/sleep 5

# DeliveryManager process stop
/sbin/init.d/esmpro_daemon stop
/usr/bin/sleep 10

# DeliveryManager process start
/sbin/init.d/esmpro_daemon start
/usr/bin/sleep 120

exit 0
```

## 導入事例

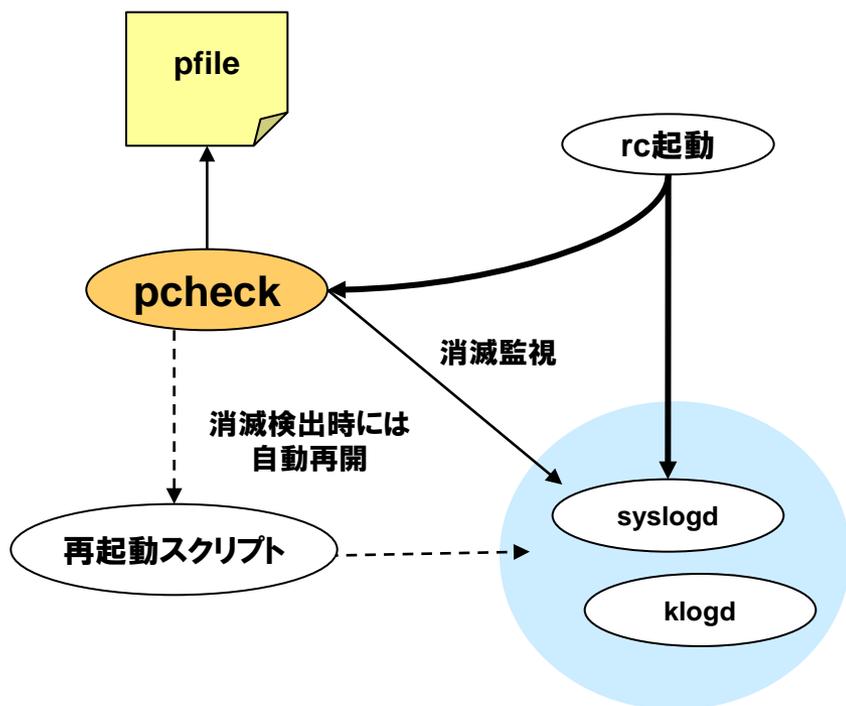
- ・運用環境に合わせた導入例を紹介します。
  - (1) syslog デーモンの監視事例
  - (2) 代表的な OS 標準デーモンの一括監視事例
  - (3) Apache HTTP Server の監視事例
  - (4) PostgreSQL の監視事例
  - (5) Jboss の監視事例
  - (6) Apache Tomcat の監視事例
  - (7) WebSAM JobCenter の監視事例
  - (8) ESMPRO/ServerAgent の監視事例
  - (9) StoragePathSavior の監視事例
  - (10) WebSAM MCOperations / ApplicationNavigator の監視事例
  - (11) HULFT の監視事例

# (1) syslog デーモンプロセスの監視事例

## syslogd の消滅監視事例

### 監視手順

- ノード立ち上げ契機で syslog デーモンを起動します。
- pcheck により消滅監視を実行します。
- 消滅検出時には、再起動スクリプトで再開します。



### RedHat Linux 5.x 以前のバージョンの例

```
<pfile>
# ProcessSaver configuration file for syslogd
##### PARAM #####
IPCKEY                0x1f000101
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheck_dump_syslog
##### PENT #####
syslogd -m 0:/etc/init.d/syslog restart:86400:3:continue:grouptag=syslog
klogd -x:/etc/init.d/syslog restart:86400:3:continue:grouptag=syslog
```

### RedHat Linux 6.x の例

```
<pfile>
# ProcessSaver configuration file for syslogd
##### PARAM #####
IPCKEY                0x1f000101
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheck_dump_syslog
##### PENT #####
/sbin/rsyslogd:/etc/init.d/rsyslog start:86400:3:continue
```

グループ指定により、依存関係のある syslog と klogd をまとめて監視

rsyslogd のみを単独で監視

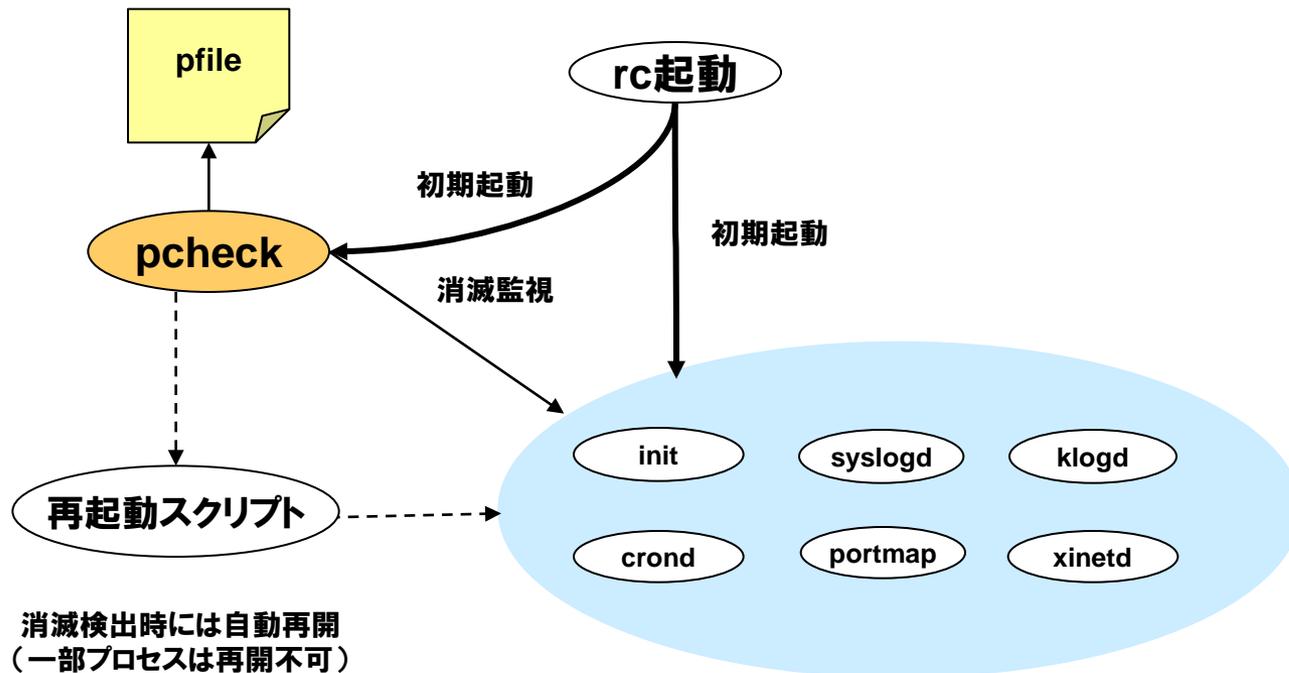
(注) 監視対象プロセス、再起動スクリプト(rcファイル)は、OSのディストリビューションやバージョン、インストール設定により異なる場合があります。

## (2) 代表的なOS標準デーモンの一括監視事例

### RedHat Linux バージョン 4.x または 5.x の消滅監視事例

#### 監視手順

- rcファイルから各種デーモン(init、syslogd、klogd、crond、portmap、xinetd)を起動します。
- pcheck による消滅監視を実行します。
- 消滅検出時には、再起動スクリプトで再開します。

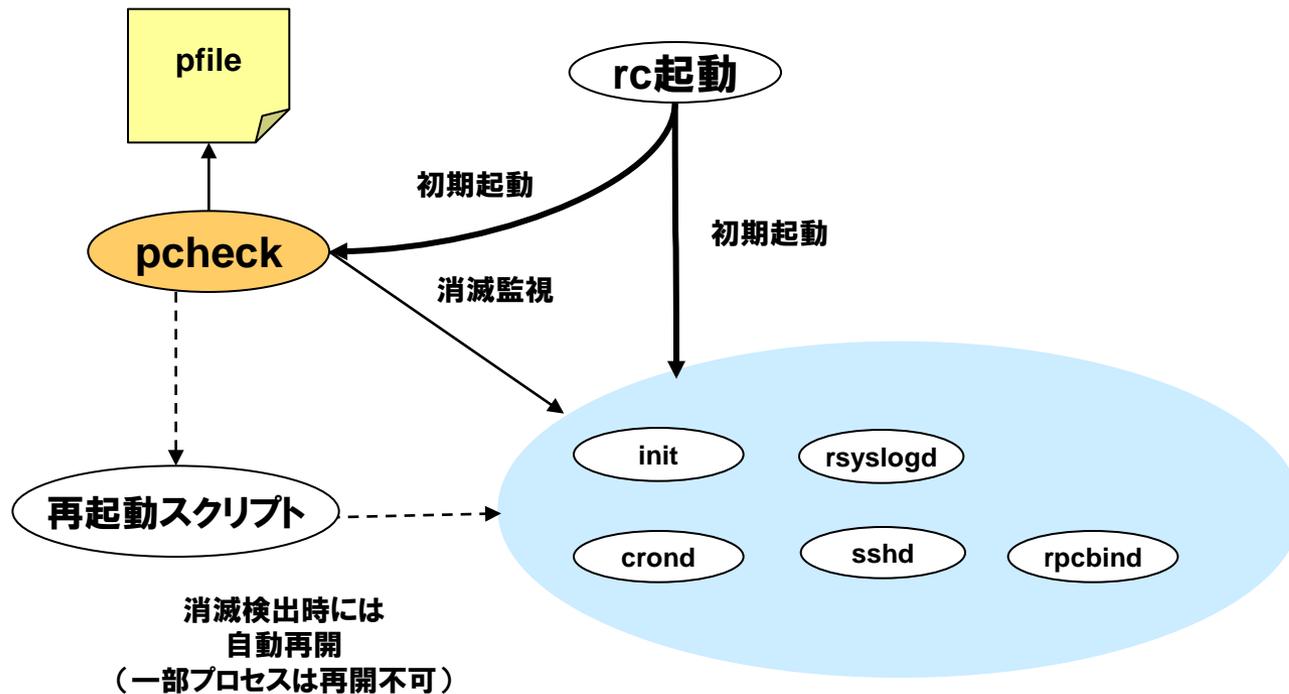


# (続き)

## RedHat Linux バージョン 6.x の消滅監視事例

### 監視手順

- rcファイルから各種デーモン(init、rsyslogd、crond、sshd、rpcbind)を起動
- pcheckによる消滅監視
- 消滅検出時には、再起動スクリプトで再開



# (続き)

## RedHat Linux 4.x または 5.x の例

```
<pfile>
##### PARAM #####
IPCKEY                0x1f000101
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheck_OS_dump

##### PENT #####
## pname:restart shell:grace:retry_count_max:retry_over_action
init:-:0:0:continue
syslogd -m 0:/etc/init.d/syslog restart:86400:3:continue:grouptag=syslog
klogd -x:/etc/init.d/syslog restart:86400:3:continue:grouptag=syslog
crond:/etc/init.d/crond start:86400:3:continue
portmap:/etc/init.d/portmap start:86400:3:continue
xinetd -stayalive -pidfile /var/run/xinetd.pid:/etc/init.d/xinetd start:86400:3:continue
```

グループ監視により、依存関係のある  
syslogd と klogd をまとめて監視

## RedHat Linux 6.x の例

```
<pfile>
##### PARAM #####
IPCKEY                0x1f000101
MSG_CHECK_INTERVAL    5
MONITOR_INTERVAL      10
SHM_DUMP_FILE         /var/opt/HA/PS/log/pcheck_OS_dump

##### PENT #####
## pname:restart shell:grace:retry_count_max:retry_over_action
/sbin/init:-:0:0:continue
/sbin/rsyslogd:/etc/init.d/rsyslog start:86400:3:continue
crond:/etc/init.d/crond start:86400:3:continue
/usr/sbin/sshd:/etc/init.d/sshd start:86400:3:continue
rpcbind:/etc/init.d/rpcbind start:86400:3:continue
```

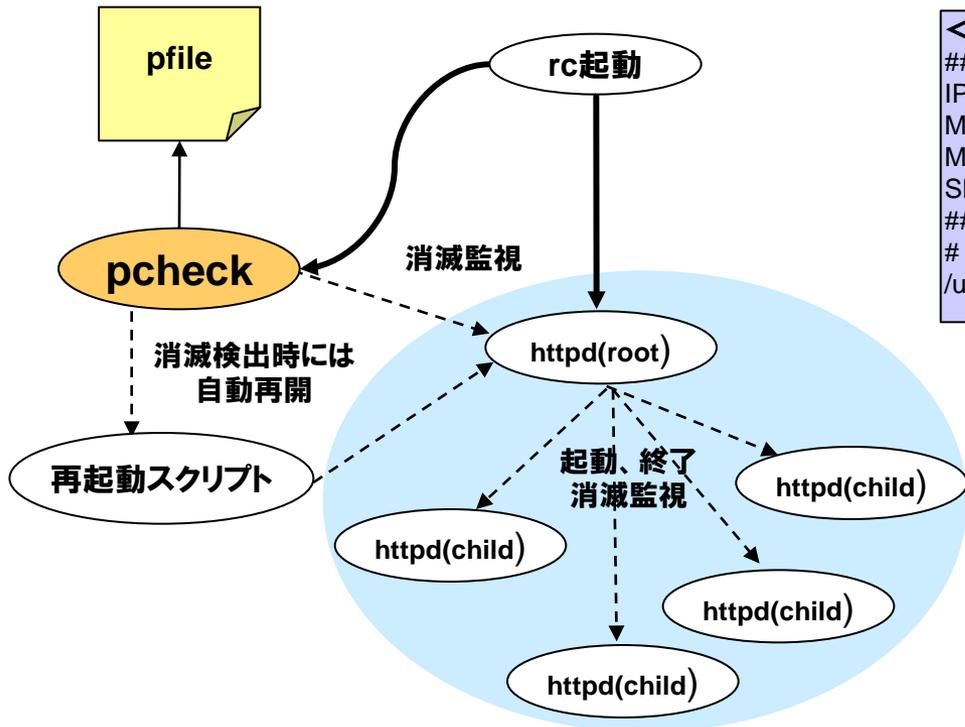
rsyslogd のみを単独で監視

# (3) Apache HTTP Server の監視事例

## Apache HTTP Server 2.2.3 の消滅監視事例

### 監視手順

- ノード立ち上げ契機で httpd(root) を起動します。
- httpd(root) により配下 httpd(child) の起動や終了、自動再開を実行します。
- pcheck により httpd(root) の消滅を監視します。



```
<pfile>
##### PARAM #####
IPCKEY                0x1f000102
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     30
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_dump_APACHE
##### PENT #####
# APACHE httpd
/usr/sbin/httpd:/var/opt/HA/PS/conf/bin/apache_restart.sh:86400:3:continue
```

(注) Webサーバとの連携機能をご利用になる場合には、ProcessSaver WebEditionが必要になります。

# (続き)

<再起動スクリプト(/var/opt/HA/PS/conf/bin/apache\_restart.sh)の例>

```
#!/bin/sh

export LANG=C

HTTPD="/usr/sbin/httpd"

PS_CMD="/bin/ps"
GREP_CMD="/bin/grep"
AWK_CMD="/bin/awk"

pid=`${PS_CMD} -ef | ${GREP_CMD} "${HTTPD}" | ${GREP_CMD} -v "${GREP_CMD}" | ${AWK_CMD} '{printf("%s ",$2)}END{printf("¥n")}'`
if [ -n "$pid" ]
then
    /bin/kill -9 $pid
fi

# apache process restart
/etc/init.d/httpd start

#sleep

/bin/sleep 10

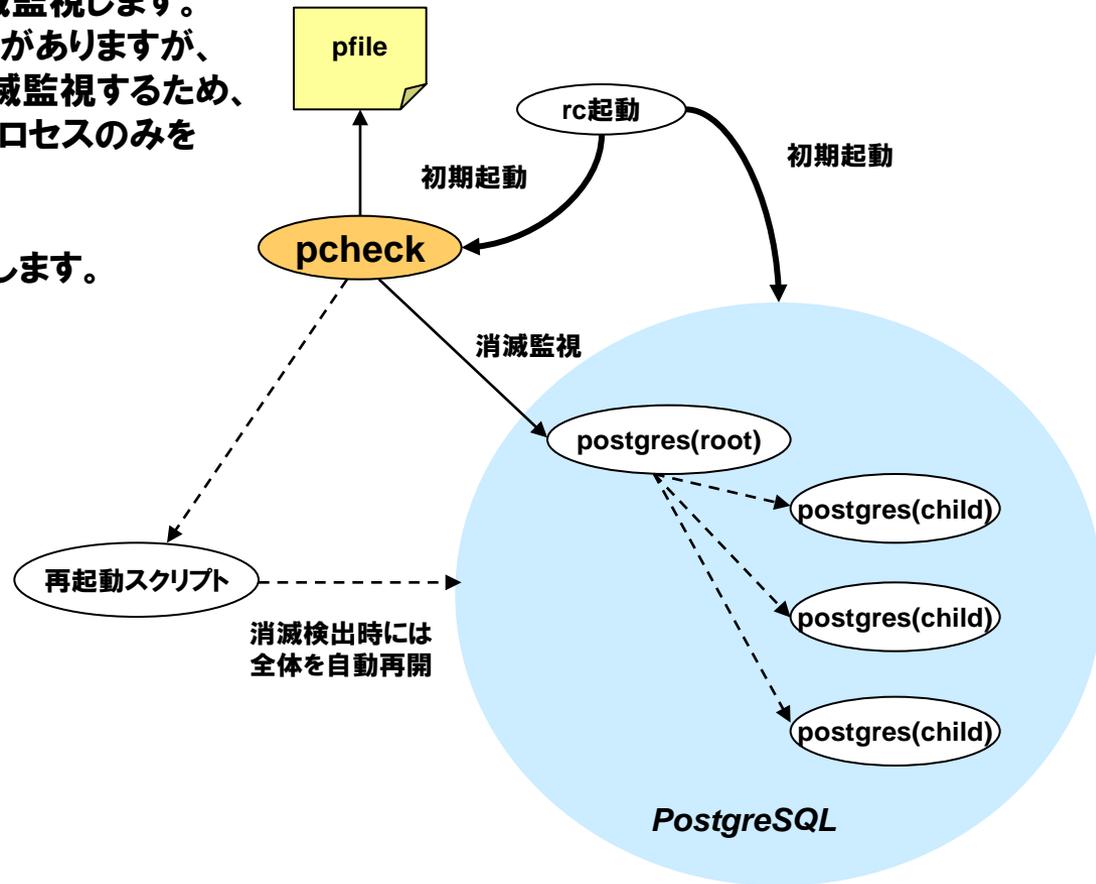
exit 0
```

# (4) PostgreSQLの監視事例

## PostgreSQL 9.1 の消滅監視事例

### 監視手順

- pcheckにより postgres のプロセスを消滅監視します。  
※postgres プロセスは複数起動する場合がありますが、  
個々のプロセスは postgres(root) が消滅監視するため、  
pcheck では、代表の postgres(root) プロセスのみを  
監視します。
- 消滅検出時には、再起動スクリプトで再開します。



(注) ご利用の環境によって、  
監視対象プロセス名や再起動手順が異なる場合があります。

# (続き)

## <pcheck のpfile例>

```
##### PARAM #####
IPCKEY          0x5f000999
MSG_CHECK_INTERVAL 5
MONITOR_INTERVAL 10
SHM_DUMP_FILE   /var/opt/HA/PS/log/pcheck_dump_PostgreSQL

##### PENT #####
## pname:restart shell:grace:retry_count_max:retry_over_action
/usr/local/pgsql/bin/postgres:/bin/su - postgres -c "/var/opt/HA/PS/conf/bin/Postgre_restart.sh":86400:5:continue
```

## <再起動スクリプト (/var/opt/HA/PS/conf/bin/Postgre\_restart.sh) の例>

```
#!/bin/sh

## PostgreSQL DB directory
DB_DIR="/usr/local/pgsql/data/"

### PostgreSQL stop
/usr/local/pgsql/bin/pg_ctl stop -D ${DB_DIR} -m smart
/bin/sleep 10

### PostgreSQL start
/usr/local/pgsql/bin/pg_ctl -D ${DB_DIR} -l /dev/null start
/bin/sleep 10

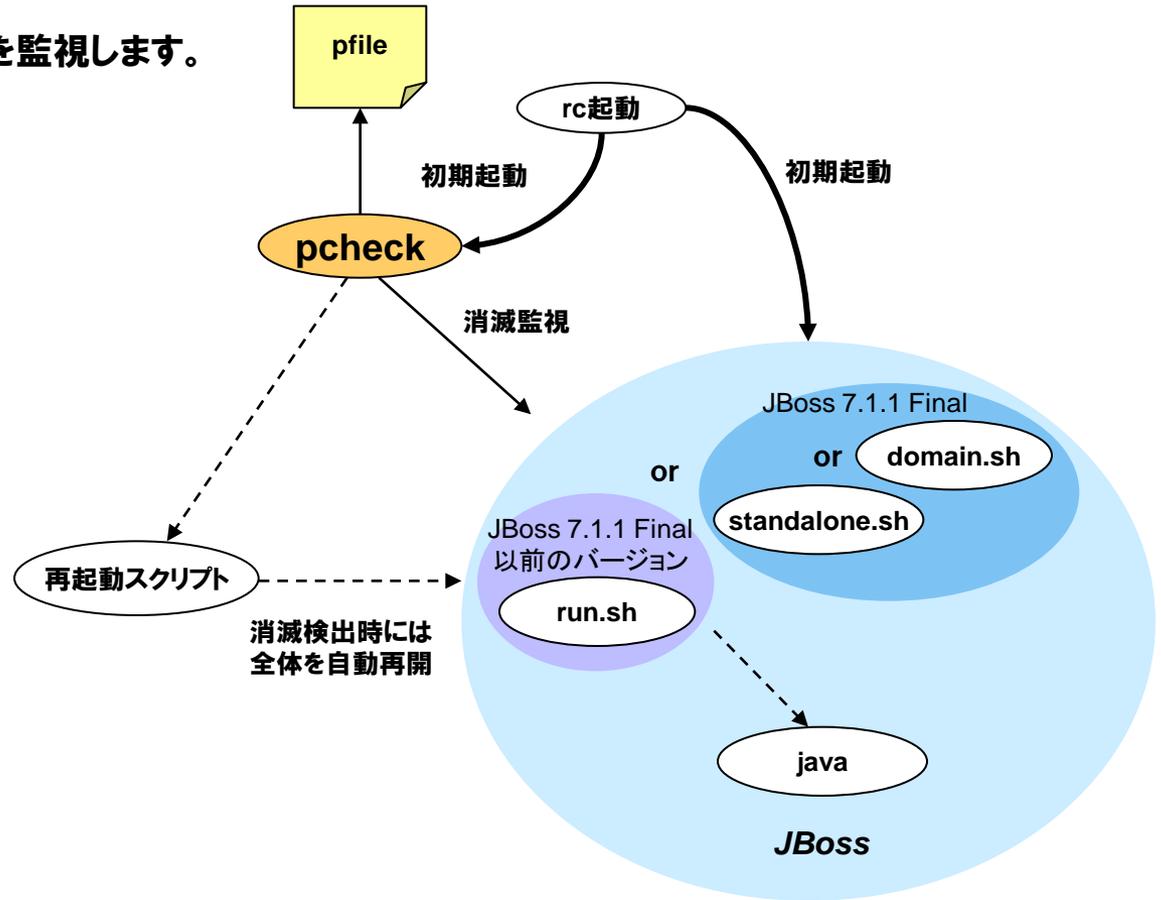
exit 0
```

# (5) JBossの監視事例

## JBoss 7.1.1 および以前のバージョンでの消滅監視事例

### 監視手順

- pcheckによりJBossの各プロセスを監視します。



(注) ご利用の環境によって、  
監視対象プロセス名、再起動手順が異なる場合があります。

# (続き)

## JBoss7.1.1 Final 以前のバージョンを使用する場合の例

### <pcheck のpfile例>

```
##### PARAM #####
IPCKEY                0x1f000110
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     30
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_JBoss_dump

##### PENT #####
### Server start script ###
/bin/sh /home/jboss/jboss4/bin/run.sh -c Server1:/var/opt/HA/PS/conf/bin/JBoss/JBoss.sh:86400:3:continue:grouptag=JBoss_Server1
### Server process ###
java -server -Xms128m -Xmx128m -Dprogram.name=run.sh -Djava.endorsed.dirs=/home/jboss/jboss4/lib/endorsed -classpath
    /home/jboss/jboss4/bin/run.jar¥:/lib/tools.jar org.jboss.Main -c Server1:/var/opt/HA/PS/conf/bin/JBoss/JBoss.sh:
    86400:3:continue:grouptag=JBoss_Server1
```

### <再起動スクリプト (/var/opt/HA/PS/conf/bin/JBoss/JBoss.sh) の例>

```
#!/bin/sh

### JBoss stop
/home/jboss/jboss4/bin/shutdown.sh -S Server1 &
/bin/sleep 30

### JBoss restart
/home/jboss/jboss4/bin/run.sh -c Server1 &
/bin/sleep 30

exit 0
```

# (続き)

## JBoss7.1.1 Final のスタンドアローンモードで使用する場合の例

### <pcheck のpfile例>

```
##### PARAM #####
IPCKEY                0x1f000110
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     30
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_JBoss_dump

##### PENT #####
### Server start script ###
/bin/sh /usr/local/jboss/bin/standalone.sh -b 0.0.0.0:/var/opt/HA/PS/conf/bin/JBoss/JBoss.sh:86400:3:continue:grouptag=JBoss_Group
### Server process ###
/usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0.x86_64/jre/bin/java -D[Standalone] -server -XX:+UseCompressedOops -XX:+TieredCompilation
-Xms64m -Xmx512m -XX:MaxPermSize=256m -Djava.net.preferIPv4Stack=true -Dorg.jboss.resolver.warning=true
-Dsun.rmi.dgc.client.gcInterval=3600000 -Dsun.rmi.dgc.server.gcInterval=3600000 -Djboss.modules.system.pkgs=org.jboss.byteman
-Djava.awt.headless=true -Djboss.server.default.config=standalone.xml -Dorg.jboss.boot.log.file=/usr/local/jboss/standalone/log/boot.log
-Dlogging.configuration=file:/usr/local/jboss/standalone/configuration/logging.properties -jar /usr/local/jboss/jboss-modules.jar
-mp /usr/local/jboss/modules -jaxpmodule javax.xml.jaxp-provider org.jboss.as.standalone -Djboss.home.dir=/usr/local/jboss -b 0.0.0.0:
/var/opt/HA/PS/conf/bin/JBoss/JBoss.sh:86400:3:continue:grouptag=JBoss_Group
```

### <再起動スクリプト (/var/opt/HA/PS/conf/bin/JBoss/JBoss.sh) の例>

```
#!/bin/sh

### JBoss stop
/usr/local/jboss/bin/jboss-cli.sh --connect --command=:shutdown
/bin/sleep 30

### JBoss restart
/usr/local/jboss/bin/standalone.sh -b 0.0.0.0 &
/bin/sleep 30

exit 0
```

# (続き)

## JBoss7.1.1 Final のドメインモードで使用する場合の例

### <pcheck のpfile例>

```
##### PARAM #####
IPCKEY                0x1f000110
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     30
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_JBoss_dump

##### PENT #####
### Server start script ###
/bin/sh /usr/local/jboss/bin/domain.sh -b 0.0.0.0:/var/opt/HA/PS/conf/bin/JBoss/JBossDomain.sh:86400:3:continue:grouptag=JBoss_Group
### Server process [Process Controller] ###
/usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0.x86_64/jre/bin/java -D[Process Controller]:
/var/opt/HA/PS/conf/bin/JBoss/JBossDomain.sh:86400:3:continue:grouptag=JBoss_Group
### Server process [Host Controller] ###
/usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0.x86_64/jre/bin/java -D[Host Controller]:
/var/opt/HA/PS/conf/bin/JBoss/JBossDomain.sh:86400:3:continue:grouptag=JBoss_Group
### Server process [server-one] ###
/usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0.x86_64/jre/bin/java -D[Server¥:server-one]:
/var/opt/HA/PS/conf/bin/JBoss/JBossDomain.sh:86400:3:continue:grouptag=JBoss_Group
### Server process [server-two] ###
/usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0.x86_64/jre/bin/java -D[Server¥:server-two]:
/var/opt/HA/PS/conf/bin/JBoss/JBossDomain.sh:86400:3:continue:grouptag=JBoss_Group
```

### <再起動スクリプト (/var/opt/HA/PS/conf/bin/JBoss/JBoss.sh) の例>

```
#!/bin/sh

### JBoss stop
/usr/local/jboss/bin/jboss-cli.sh --connect controller=localhost:9999 /host=root:shutdown
/bin/sleep 30

### JBoss restart
/usr/local/jboss/bin/domain.sh -b 0.0.0.0 &
/bin/sleep 30

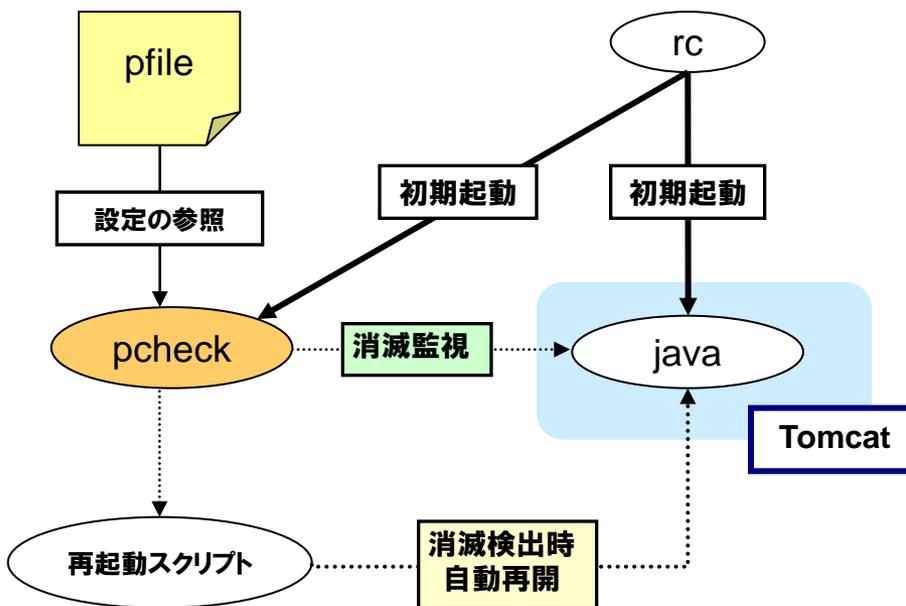
exit 0
```

# (6) Apache Tomcatの監視事例

## Apache Tomcat 7.0 の消滅監視事例

### 監視手順

- pcheckにより Apache Tomcat の java プロセスを消滅監視します。
- 消滅検出時には、再起動スクリプトで再開します。



```
<Pfile Tomcat>
# ProcessSaver configuration file for Tomcat
##### PARAM #####
IPCKEY                                0x1f000003
MSG_CHECK_INTERVAL                    5
MONITOR_INTERVAL                      30
SHM_DUMP_FILE                         /var/opt/HA/PS/log/pcheck_Tomcat_dump

##### PENT #####
##### Tomcat #####
/opt/java1.5/bin/java -Djava.util.logging.manager=org.apache.juli.ClassLoad
erLogManager -Djava.util.logging.config.file=/usr/local/tomcat/conf/logging.
properties -Djava.endorsed.dirs=/usr/local/tomcat/common/endorsed -clas
spath */usr/local/tomcat/bin/bootstrap.jar*/usr/local/tomcat/bin/commons-
logging-api.jar -Dcatalina.base=/usr/local/tomcat -Dcatalina.home=/usr/loc
al/tomcat -Djava.io.tmpdir=/usr/local/tomcat/temp org.apache.catalina.start
up.Bootstrap start:/var/opt/HA/PS/conf/bin/tomcat_restart.sh:86400:3:continue
```

監視対象のtomcatのjavaプロセスを、  
ps -ef で出力されるプロセス名ですべて  
記載した場合の例。  
ただし：(コロン)の付くプロセス名はその  
ままでは指定できないため、\*(バックス  
ラッシュコロン)と指定する必要がある。

(注) ご利用の環境によって監視対象プロセス名、再起動手順が異なる場合があります。

# (続き)

<再起動スクリプト ( /var/opt/HA/PS/con/bin/tomcat\_restart.sh )の例>

```
#Start shell for Tomcat

export LANG=C

# JAVA_HOME

JAVA_HOME="/opt/java1.5/bin"
export JAVA_HOME

PS_CMD="/bin/ps"
GREP_CMD="/bin/grep"
AWK_CMD="/bin/awk"

HTTPD="/opt/java1.5/bin/java -Djava.util.logging.manager=org.apache"

# Tomcat Server process start
/etc/init.d/tomcat stop
/bin/sleep 10

# httpd process kill
pid=`${PS_CMD} -ef | ${GREP_CMD} "${HTTPD}" | ${GREP_CMD} -v "${GREP_CMD}" | ${AWK_CMD} '{printf("%s ",$2)}END{printf("¥n")}'`
if [ -n "$pid" ]
then
    /bin/kill -9 $pid
fi

# Tomcat Server process start
/etc/init.d/tomcat start

#sleep
/bin/sleep 10

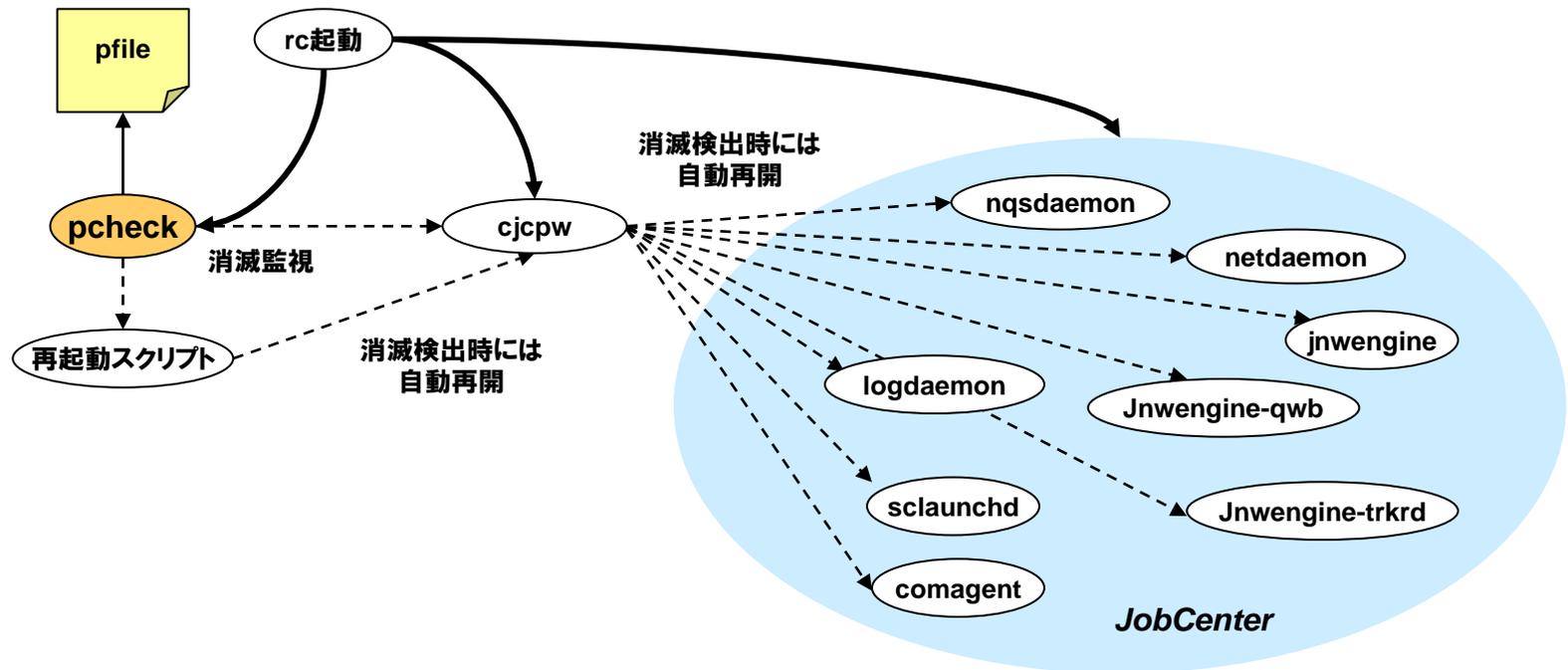
exit 0
```

# (7) WebSAM JobCenter の監視事例

## WebSAM JobCenter 12.8.1 の消滅監視事例

### 監視手順

- pcheck が JobCenter の管理デーモン(cjcpw)を監視
  - cjcpw が各 JobCenter の各プロセスを監視
- ※JobCenter の他プロセスは cjcpw デーモンによって消滅監視が行われているため、pcheck で監視を行う必要はありません。



# (続き)

```
<pfile>
##### PARAM #####
IPCKEY                0x3f000008
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     10
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_JobCenter_dump
##### PENT #####
/usr/lib/nqs/cluster/cjcpw:/var/opt/HA/PS/conf/bin/restart_JC.sh:86400:3:shutdown
```

## <再起動スクリプト (/var/opt/HA/PS/conf/bin/restart\_JC.sh) の例>

```
#!/bin/sh

## JobCenter stop } JobCenterの停止処理
/etc/init.d/nqs stop
/bin/sleep 10

## JobCenter start } JobCenterの起動処理
/etc/init.d/nqs start
/bin/sleep 10

exit 0
```

(注) ご利用の環境によって、監視対象プロセス名、再起動手順が変更となる場合があります。  
本設定はクラスタ環境の場合の監視事例となります。  
ローカル環境の場合は、cjcpw -local プロセスを監視します。

# (8) ESMPRO/ServerAgent の監視事例

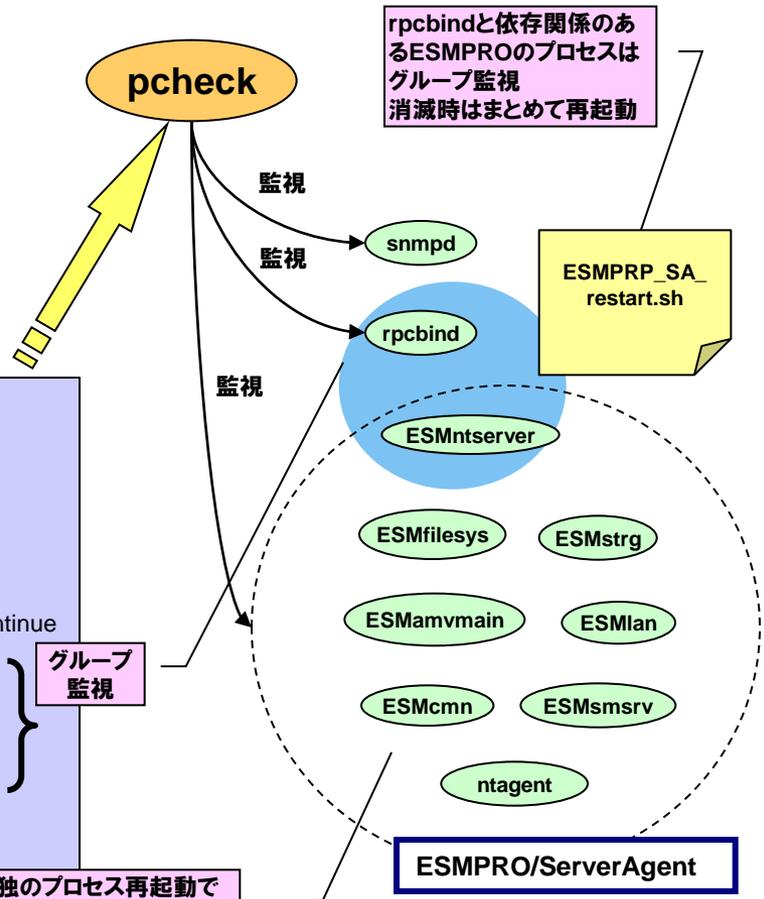
## ESMPRO/ServerAgent 4.4 の消滅監視事例

### 監視手順

- pcheckにより ESMPRO/ServerAgent のプロセスと依存関係のある rpcbind、snmpdを消滅監視します。
  - ※rpcbind、ESMntserver プロセスは、依存関係があり消滅時にはまとめて再起動する必要があるため、グループ監視を行います。
  - ※ESMntserver 以外の ESMPRO/ServerAgentの各プロセスは消滅時には単独の再起動で復旧が可能のため単独で監視します。

```
<pfile>
##### PARAM #####
IPCKEY                0x1f000112
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     10
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_dump_ESMPRO_SA

##### PENT #####
/usr/sbin/snmpd -Lsd -Lf /dev/null -p /var/run/snmpd.pid -a¥:service snmpd start:86400:3:continue
portmap:/var/opt/HA/PS/conf/bin/ESMPRO_SA_restart.sh:86400:3:
                                continue:grouptag=ESMPRO_SA }
# ESMPRO/ServerAgent
/opt/nec/esmpro_sa/bin/ESMntserver:/var/opt/HA/PS/conf/bin/ESMPRO_SA_restart.sh:
                                86400:3:continue:grouptag=ESMPRO_SA }
/opt/nec/esmpro_sa/bin/ESMntserver:/var/opt/HA/PS/conf/bin/ESMPRO_SA_restart.sh:
/opt/nec/esmpro_sa/bin/ESMfilesys:service ESMfilesys start:86400:3:continue
ESMstrg:service ESMstrg start:86400:3:continue
/opt/nec/esmpro_sa/bin/ESMlan:service ESMlan start:86400:3:continue
/opt/nec/esmpro_sa/bin/ESMamvmain:service ESMamvmain start:86400:3:continue
/opt/nec/esmpro_sa/bin/ESMcmn:service ESMcmn start:86400:3:continue
/opt/nec/esmpro_sa/bin/ESMsmsrv:service ESMsmsrv start:86400:3:continue
/opt/nec/esmpro_sa/bin/ntagent:service ntagent start:86400:3:continue
```



(注) ご利用の環境によって監視対象プロセス名、再起動手順が異なる場合があります。

# (続き)

<再起動スクリプト (/var/opt/HA/PS/conf/bin/ESMPRO\_SA\_restart.sh) の例>

```
#!/bin/sh
```

```
##portmap start  
service portmap start  
/bin/sleep 5
```

}

portmap の起動処理

```
##ESMPRO/ServerAgent restart  
/opt/nec/esmpro_sa/bin/ESMRestart  
/bin/sleep 10
```

}

ESMPRO/ServerAgentの再起動処理

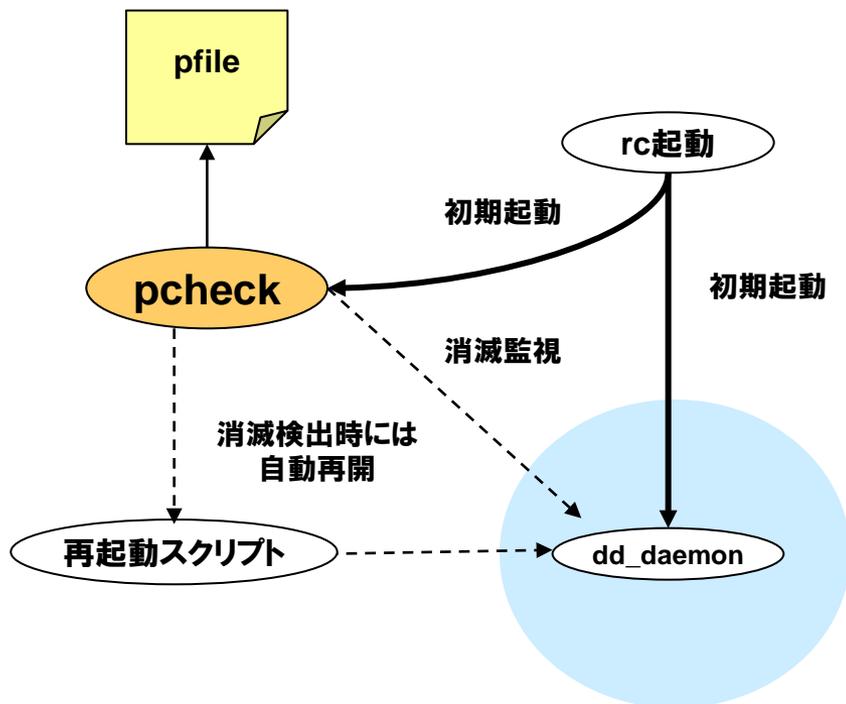
```
exit 0
```

# (9) StoragePathSavior の監視事例

## StoragePathSavior Enterprise for Linux 4.2.2 の消滅監視事例

### 監視手順

- pcheck により dd\_daemon (パス巡回デーモン) を監視します。
- 消滅検出時には、再起動スクリプトで再開します。



```
<pfile>
##### PARAM #####
IPCKEY                                0x1f00011a
MSG_CHECK_INTERVAL                     5
MONITOR_INTERVAL                       10
SHM_DUMP_FILE                          /var/opt/HA/PS/log/pcheck_dump_SPS

##### PENT #####
# StoragePathSavior
dd_daemon:/etc/rc.d/init.d/dd_daemon restart:86400:3:continue
```

(注) ご利用の環境によって監視対象プロセス名、再起動手順が異なる場合があります。

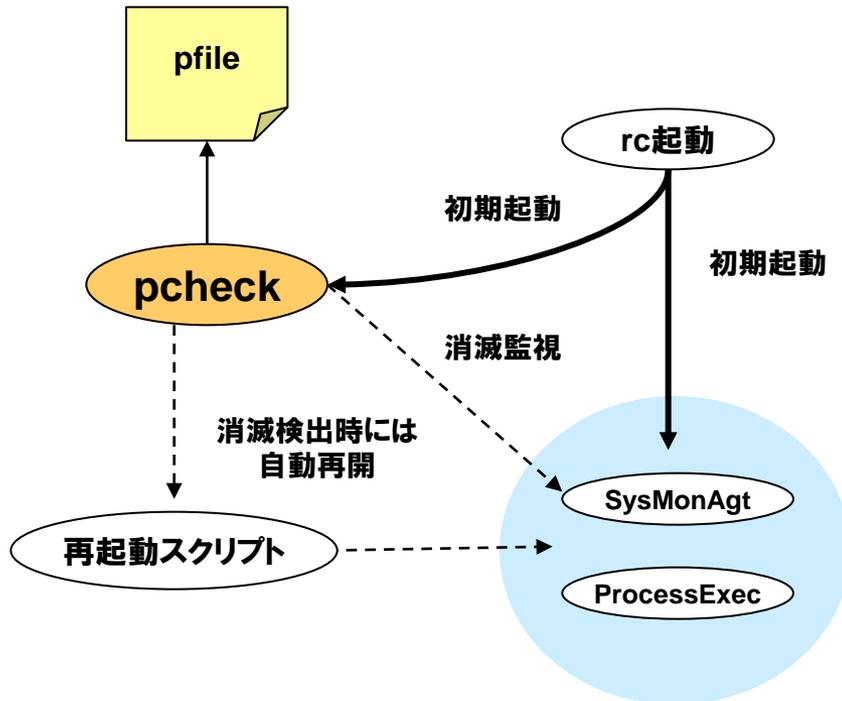
# (10) WebSAM MCOperations / ApplicationNavigator の監視事例

## 以下製品導入時の消滅監視事例

- WebSAM MCOperations 3.4.1 (エージェント)
- ApplicationNavigator 3.0.0 (エージェント)

### 監視手順

- pcheck により MCOperations の各プロセスを監視します。
  - ※SysMonAgt、ProcessExec プロセスは、依存関係があり消滅時にはまとめて再起動する必要があるため、グループ監視を行います。
- 消滅検出時には、再起動スクリプトで再開します。



```
<pfile>
##### PARAM #####
IPCKEY                                0x1f000116
MSG_CHECK_INTERVAL                    5
MONITOR_INTERVAL                      10
SHM_DUMP_FILE                         /var/opt/HA/PS/log/pcheck_MCOAgent_dump

##### PENT #####
# MCOperations(agent)
/opt/UMF/Operations/Agent/bin/SysMonAgt:/var/opt/HA/PS/conf/bin/
MCOAgent_restart.sh:86400:3:continue:group>tag=sysmon
/opt/UMF/Operations/Agent/bin/ProcessExec ja_JP.utf8:/var/opt/HA/PS/conf/bin/
MCOAgent_restart.sh:86400:3:continue:group>tag=sysmon
```

```
<再起動スクリプト (/var/opt/HA/PS/conf/bin/MCOAgent_restart.sh) の例>
#!/bin/sh

## MCOperations Agnet stop
/etc/init.d/UMFOperationsAgent_1 stop
/bin/sleep 10

## MCOperations Agnet start
/etc/init.d/UMFOperationsAgent_1 start
/bin/sleep 10

exit 0
```

(注) WebSAM MCOperations と ApplicationNavigator は、共通のエージェントプロセスを使用しています。

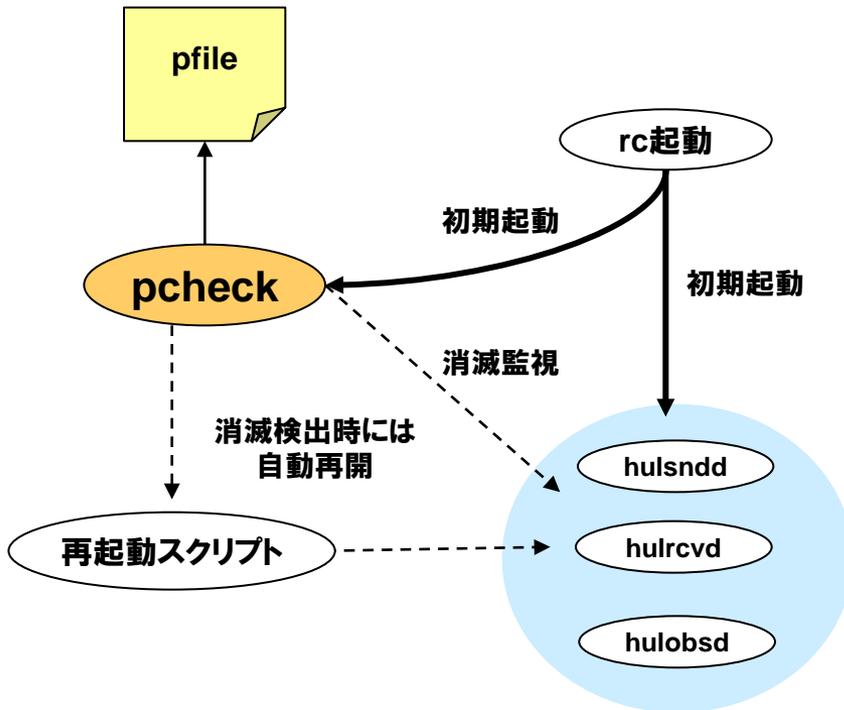
(注) ご利用の環境によって監視対象プロセス名、再起動手順が異なる場合があります。

# (11) HULFT の監視事例

## HULFT7 for Linux-ENT の消滅監視事例

### 監視手順

- pcheck により HULFT の各プロセスを監視します。
- 消滅検出時には、再起動スクリプトで再開します。



### HULFT7 (type L) の例

```
<pfile>
##### PARAM #####
IPCKEY                0x1f000112
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     10
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_hulft_dump

##### PENT #####
/opt/hulft/bin/hulsndd:/var/opt/HA/PS/conf/bin/hulftd start 1:86400:3:continue
/opt/hulft/bin/hulrcvd:/var/opt/HA/PS/conf/bin/hulftd start 2:86400:3:continue
/opt/hulft/bin/hulobsd:/var/opt/HA/PS/conf/bin/hulftd start 3:86400:3:continue
```

### HULFT7 (type L-CL) の例

```
<pfile>
##### PARAM #####
IPCKEY                0x1f000113
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     10
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_hulftcl_dump

##### PENT #####
/usr/local/HULFT/bin/hulsndd:/usr/local/HULFT/bin/hulclustersnd -start:86400:3:continue
/usr/local/HULFT/bin/hulrcvd:/usr/local/HULFT/bin/hulclusterrcv -start:86400:3:continue
/usr/local/HULFT/bin/hulobsd:/usr/local/HULFT/bin/hulclusterobs -start:86400:3:continue
```

(注) ご利用の環境によって監視対象プロセス名、再起動手順が異なる場合があります。

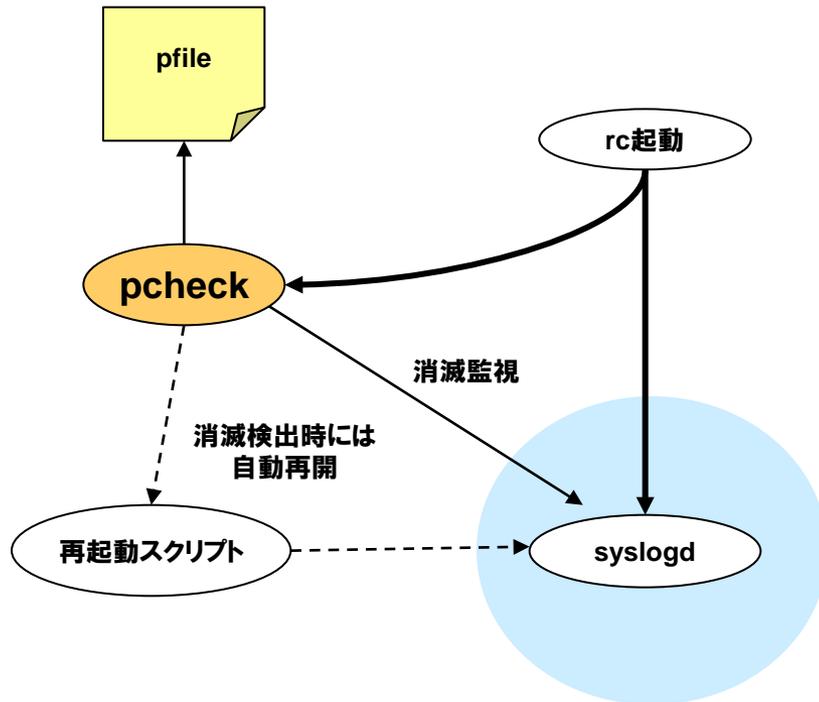
## 導入事例

- 運用環境に合わせた導入例を紹介します。
  - (1) syslog デーモンプロセスの監視事例
  - (2) Apache HTTP Server の監視事例

# (1) syslog デーモンプロセスの監視事例

## syslogdの消滅監視手順

- ノード立ち上げ契機でsyslogデーモン起動
- pcheckにより消滅監視を実行
- 消滅検出時には、再起動スクリプトで再開



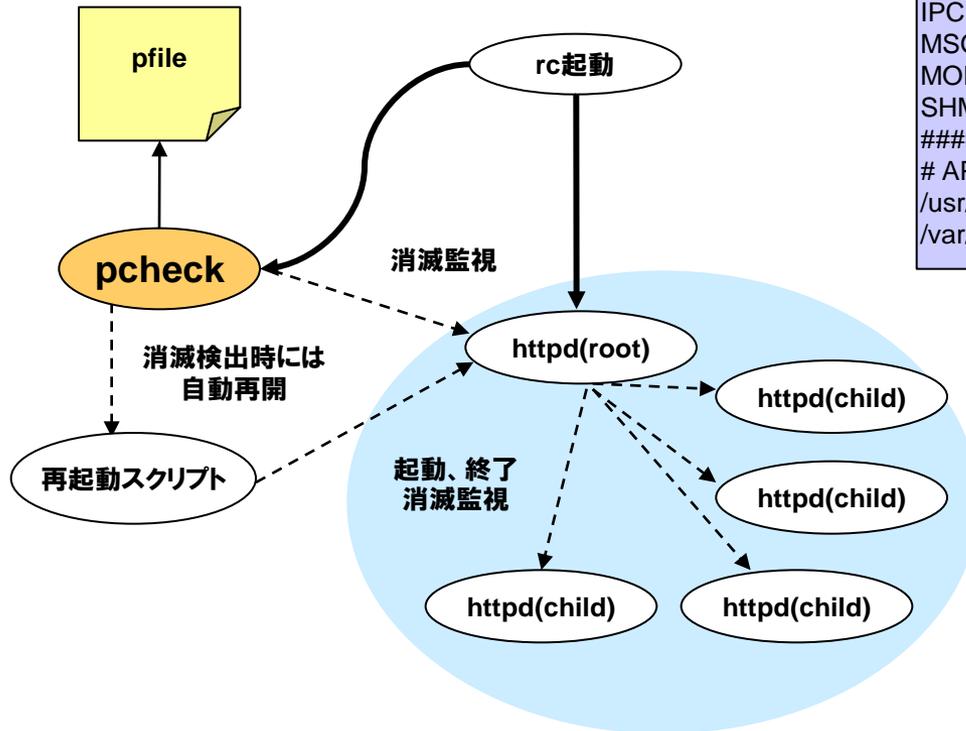
```
<file>
# ProcessSaver configuration file for syslogd
##### PARAM #####
IPCKEY                0x1f000101
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     10
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_dump_syslog
##### PENT #####
/usr/sbin/syslogd:/etc/init.d/syslog start:86400:3:continue
```

rcファイルを直接呼び出す

## (2) Apache HTTP Server の監視事例

### Apache HTTP Server の消滅監視手順

- ノード立ち上げ契機で httpd(root) を起動
- httpd(root) により配下 httpd(child) の起動や終了、自動再開を行います
- pcheck により httpd(root) の消滅を監視することで、全体の可用性を実現します



```
<pfile>
# ProcessSaver configuration file for APACHE
##### PARAM #####
IPCKEY                0x1f000102
MSG_CHECK_INTERVAL   5
MONITOR_INTERVAL     30
SHM_DUMP_FILE        /var/opt/HA/PS/log/pcheck_dump_APACHE
##### PENT #####
# APACHE httpd
/usr/local/apache2/bin/httpd:
/var/opt/HA/PS/conf/bin/APACHE/web.sh:86400:3:continue
```

**(注) Webサーバとの連携機能をご利用になる場合には、ProcessSaver WebEdition が必要になります。**

# 商標

- HP、HP-UXは、米国における米国 Hewlett-Packard Companyの登録商標です。
- Red Hatは、米国およびその他の国におけるRed Hat,Inc.の商標または登録商標です。
- Linux は、Linus Torvalds氏の米国およびその他の国における、登録商標または商標です。
- SUSEは、米国およびその他の国における Novell,Inc. の商標 です。
- MIRACLE LINUXは、ミラクル・リナックス株式会社の登録商標です
- Asianuxは、ミラクル・リナックス株式会社の日本における登録商標です。
- Turbolinuxは、ターボリナックス株式会社の商標または登録商標です。
- JavaおよびすべてのJava関連の商標およびロゴは、米国およびその他の国における米国 Oracle, Inc.の商標または登録商標です。
- ProcessSaverは日本電気株式会社の登録商標です。
- 記載の製品名および会社名はすべて各社の商標または登録商標です。

Empowered by Innovation

**NEC**