CLUSTERPRO
MC ProcessSaver 2.0 for Linux

FAQ 集

- 導入に関する質問
- 構築段階における質問
- 運用段階における質問
- 主なメッセージ一覧

はしがき

本書は、CLUSTERPRO MC ProcessSaver 2.0 for Linux （以後 ProcessSaver と記載します）のよく寄せられるご質問とその回答について記載したものです。

（1） 商標および登録商標

- ✓ Linux は、Linus Torvalds 氏の米国およびその他の国における商標または登録商標です。
- ✓ CLUSTERPRO は、日本電気株式会社の登録商標です。
- ✓ ProcessSaver は、日本電気株式会社の登録商標です。
- ✓ その他、本書に登場する会社名および商品名は各社の商標または登録商標です。
- ✓ なお、本書では®、TM マークを明記しておりません。

目次

はじめに

本書はお客様からよく寄せられるご質問とその回答を記載しております。
機能、利用方法、活用方法に加えてトラブルの解決方法などについても記載しておりますので、お問い合わせの前に本書、各マニュアルをご確認ください。

（1）本書の活用（検索）方法

- 質問のカテゴリ（導入・構築・運用）をもとに検索を行う場合は、以下の項目から検索してください。

- 画面コンソールやシステムログに出力されたメッセージから検索する場合は、以下の項目から検索してください。

1　導入に関する質問

1.1　動作環境について

Q1.	ProcessSaver をインストールおよび、動作させるために必要なディスク使用量はどれくらいですか？
A1.	ProcessSaver は /opt と /var 配下を使用します。 使用量は以下のとおりです。 ディレクトリ / 説明 / サイズ /opt 配下 / コマンドなどのバイナリファイルを格納 / 約 1MB /var 配下 / 監視定義ファイル（pfile）や ProcessSaver が出力するメモリダンプファイルを保存 / 約 15MB メモリダンプファイルは、すべてサイクリックになっていますので 15MB を超えることはありません。 上の表は、pcheck 1プロセスの使用量です。 pcheck を複数起動する場合の /var の使用量は、約 15MB × 起動 pcheck 数となります。

Q2.	ProcessSaver のメモリ使用量はどれくらいですか？
A2.	使用するメモリは、pcheck 1プロセスにつき約 15MB です。 pcheck を複数起動する場合のメモリ使用量は、約 15MB × 起動 pcheck 数となります。

Q3.	/var 配下の空き容量がなくなった場合、ProcessSaver の動作上なにか問題がありますか？
A3.	/var 配下の空き容量がない状態では、正しく動作しない可能性があるため動作の保障ができません。 /var 配下の不要ファイルを削除して空き容量を確保したあと、pcheck を再起動してください。 その際、以下のファイルは ProcessSaver の管理ファイルですので、削除しないでください。 /var/tmp/.pcheck.***IPCKEY*** ファイル名の ***IPCKEY*** 部分は、監視定義ファイル（pfile）内の IPCKEY パラメータに設定している値（16 進数）を 10 進数に変換した値で表示されます。 監視定義ファイル（pfile）の IPCKEY が “0x1f000100” の場合、以下のファイルが管理ファイルとなります。 - /var/tmp/.pcheck.520093952 * IPCKEY パラメータとは、ProcessSaver で使用する共有メモリを確保するための key を指定するパラメータです。

Q4.	ProcessSaver で監視を実施しているサーバの Java(JRE など) をバージョンアップしたいのですが、ProcessSaver の動作に影響はありますか？
A4.	ProcessSaver は、Java を使用しておりませんので影響ありません。 ただし、Java(JRE など) のバージョンを変更することにより ProcessSaver で監視対象としているプロセスのプロセス名が変更となる場合には、変更されたプロセス名にあわせて監視定義ファイル（pfile）を修正していただく必要があります。

Q5.	ProcessSaver で Apache Struts の機能を使用しているか教えてください。
A5.	ProcessSaver では、Apache Struts を使用しておりません。

1.2 ライセンスについて

Q1.	コードワード申請時に必要なホスト情報は、どのようなものですか？
A1.	コードワード申請は、以下のどちらかの情報が必要です。 ・ ホスト ID ・ NIC の MAC アドレス それぞれの情報の確認方法は、以下を参照してください。 - ホスト ID の確認方法 hostid コマンドを実行し、出力された値（16 進数）を 10 進数に変換した値をコードワード申請用紙に記載してください。 [実行例] # /usr/bin/hostid 7f0100 実行例の場合、コードワード申請用紙には出力値 “7f0100” を 10 進数に変換した値である“8323328” を記載してください。 - MAC アドレスの確認方法 ifconfig コマンドを実行し、“HWaddr” 項目に表示された値（16 進数）のコロン(:)を削除した状態で 10 進数に変換した値をコードワード申請用紙に記載してください。 NIC が複数存在する場合は、搭載されている NIC のいずれか 1 つの MAC アドレスを取得してください。 [実行例] # /sbin/ifconfig eth0 eth0 Link encap:Ethernet HWaddr 00:0D:29:46:1F:70 inet addr:xx.xx.xx.xx Bcast:xx.xx.xx.xx Mask:xx.xx.xx.xx ~ 以下、表示結果省略 ~ 実行例の場合、コードワード申請用紙には出力値 “00:0D:29:46:1F:70” のコロン(:)を削除し、その数値を 10 進数に変換した値である “56527036272” を記載してください。 コードワードの詳細は、製品添付資料の『コードワードについて』を参照してください。

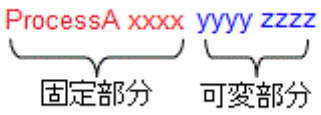
Q2.	ProcessSaver でクラスタ構成の場合、ライセンスは現用系のみ購入すればよいですか？ それとも現用系／待機系のそれぞれ購入する必要がありますか？
A2.	現用系／待機系のそれぞれにライセンスが必要です。 ライセンスは、CPU ライセンスとなっております。 また、ライセンスにはインストール媒体は、付属しておりません。 そのため、インストール媒体をお持ちでない場合は、インストール媒体を別途ご購入いただく必要があります。 ＊ インストール媒体は、現用系／待機系で共用していただくことが可能です。

Q3.	仮想環境で ProcessSaver を使用したいのですが特別なライセンスが必要ですか？
A3.	仮想マシンごとに使用できる仮想マシン用ライセンスを用意しております。 仮想マシンで ProcessSaver を使用する場合は、仮想マシン用ライセンスをご購入ください。

1.3 設定方法について

Q1.	監視したいプロセスのほかに同一名（同じ名称）のプロセスが複数起動している状態です。 すべての同一名（同じ名称）プロセスを監視したい場合、どのような方法で監視すればよいでしょうか？
A1.	監視定義ファイル（pfile）の個別部情報(PENT) に監視を行いたい個数分の設定を記載してください。 PENT に定義することで、それぞれの行に対して以下に同一名（同じ名称）のプロセス（ProcessA -U -n）が、4個起動しそのすべてを監視する監視定義ファイル（pfile）の設定例を記載します。 [監視定義ファイル（pfile）の設定例] ##### PENT ##### ProcessA -U -n:＜再起動スクリプト＞:86400:3:continue ProcessA -U -n:＜再起動スクリプト＞:86400:3:continue ProcessA -U -n:＜再起動スクリプト＞:86400:3:continue ProcessA -U -n:＜再起動スクリプト＞:86400:3:continue

Q2.	同一名（同じ名称）プロセスが、一定数以上起動しているか監視を行いたいのですが、どのような方法で監視すればよいでしょうか？
A2.	個数監視（プロセス数で監視する方式）をご使用ください。 以下に個数監視方法と設定方法を簡単に記載します。 監視定義ファイル （pfile）のオプション部に min_proc_count オプションを指定することで、監視対象プロセスを含めた同一名プロセス数の監視を行うことができます。 以下に同一名（同じ名称）のプロセスが 4 個未満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する監視定義ファイル （pfile）の設定例を記載します。 [監視定義ファイル （pfile）の設定例] ##### PENT ##### ＜プロセス名＞:＜再起動スクリプト＞:86400:3:continue:min_proc_count=4 個数監視を使用した場合の注意点を以下に記載します。 pcheck は、同一名（同じ名称）プロセス群から『1.3 設定方法について』『Q1』に記載しているルールに則って代表して1つのプロセスを監視するとともに、そのプロセスと同一名（同じ名称）の個数を監視します。 そのため、代表として選択された監視対象プロセスが異常終了した場合、たとえ同一名（同じ名称）プロセスがmin_proc_count オプションに指定した数以上起動していても監視対象プロセスの異常と判断します。 個数監視の詳細は、製品添付資料の『ユーザーズガイド』『4.2.同一名プロセス監視の導入手順』を参照してください。

Q3.	起動ごとに引数が変わってしまうプロセスを監視したいのですが、どのような設定をすればよいでしょうか？
A3.	監視対象プロセスのプロセス名の固定（起動ごとに変更されない）部分を監視定義ファイル（pfile） の process_name パラメータに指定してください。 設定例を以下に記載します。 - 監視対象プロセス  [監視定義ファイル（pfile） の設定例] ##### PENT ##### ProcessA xxxx: ＜再起動スクリプト＞:86400:3:continue

Q4.	Oracle の監視を行う場合の一般的な設定について教えてください。
A4.	Oracle のバックグラウンドプロセスおよび、リスナープロセスを監視してください。 Oracle 監視の詳細は、製品添付資料の『テンプレートガイド』『2-1. Oracle Database の監視事例』を参照してください。 ProcessSaver で Oracle を監視する場合の注意点は、Oracle プロセス起動までに時間がかかるため監視開始時および、再起動時は監視待ち合わせを行う必要があります。 また、tnslsnr プロセスは、データベースへの接続時に複数プロセスが起動するため、データベースへ接続を行っていない状態で監視を開始してください。 Oracle の動作などを考慮した監視を行う場合は、CLUSTERPRO MC ApplicationMonitor の併用導入をご検討ください。

Q5.	Oracle の監視を ProcessSaver と CLUSTERPRO MC ApplicationMonitor を併用した方法で監視することは可能でしょうか？
A5.	『1.3 設定方法について』『Q4.』の記載のとおり、Oracle の動作などを考慮した監視を行う場合は、CLUSTERPRO MC ApplicationMonitor での監視を推奨しております。 ただし ProcessSaver と CLUSTERPRO MC ApplicationMonitor のそれぞれに特化した監視方法を使用し、併用して監視を行うことも可能です。 ProcessSaver では、Oracle のプロセスを監視することでプロセスの消滅を即座に検知できます。 Oracle 監視の詳細は、製品添付資料の『テンプレートガイド』『2-1. Oracle Database の監視事例』を参照してください。

Q6.	CLUSTERPRO でクラスタ連携を行う場合、監視定義ファイル（pfile） に設定するリトライオーバアクションの設定値 “exit” と “shutdown” の違いについて教えてください。
A6.	基本動作は同じですが pcheck の終了コード（戻り値）が異なります。 “exit” と “shutdown” ともに監視対象プロセスの再起動に失敗し、最大リトライ回数に達した場合に、pcheck を終了します。 - exit 終了コード ： -1 CLUSTERPRO などのクラスタウェア製品と連携し、フェイルオーバを行う場合はこの値を指定します。 - shutdown 終了コード ： 0 pcheck を階層的に使用したグループ監視（pcheck に親子関係を持たせる方法）で監視を行う場合などにおいて、子 pcheck にはこのパラメータを指定します。 クラスタ連携を行う場合、通常は “exit” を使用します。 ただし、”shutdown” を使用しても、クラスタ連携動作に問題はありません。 将来の機能強化において pcheck の終了処理を “exit” 指定時と “shutdown” 指定時で動作仕様を分ける可能性がありますのでクラスタ連携を行う場合は、“exit” を指定することを推奨します。

Q7.	ProcessSaver の設定ファイル、再起動スクリプトのサンプルなどはありますか？
A7.	用意しております。 監視設定のサンプルや設定方法を記載したテンプレートガイド、およびテンプレート（サンプルファイル）をインストール媒体に同梱しています。 代表的な製品や OS 標準の重要プロセスなどについては、テンプレートガイド、およびテンプレート（サンプルファイル）をお使いいただくことで容易に導入していただくことが可能です。 テンプレート（サンプルファイル）の詳細は、製品添付資料の『テンプレートガイド』を参照してください。

Q8.	プロセスに異常が発生し再起動を行う場合、関係性（依存関係）のあるプロセスも再起動を行う必要があります。 この場合、どのような方法で監視を行えばよいでしょうか？
A8.	グループ監視機能をご使用ください。 ProcessSaver は、依存関係を持っている複数のプロセスを１つのグループとして設定することにより、複数のプロセスをまとめて監視することができます。 グループ監視機能を使用するには、監視定義ファイル（pfile）のオプション部に grouptag オプションを指定してください。 ProcessA,B,C に依存関係がある場合の監視定義ファイル（pfile）の設定例を以下に記載します。 [監視定義ファイル（pfile）の設定例] ##### PENT ##### ProcessA:＜再起動スクリプト＞:86400:3:continue:grouptag=XXX ProcessB:＜再起動スクリプト＞: 86400:3:continue:grouptag=XXX ProcessC:＜再起動スクリプト＞: 86400:3:continue:grouptag=XXX * grouptag オプション使用時の注意事項を以下に記載します。 XXX には、任意のグループ名を指定してください。 同一グループ名、再起動スクリプト名、GRACE 値、再起動回数、リトライオーバアクションは同じである必要があります。 grouptag を指定することにより、ProcessA,B,C のどのプロセスの消滅を検知した場合でも同じ再起動スクリプトが実行され、まとめて再起動を行うことができます。 再起動スクリプトは、対象プロセスを起動した後に sleep コマンドなどで一定時間の猶予をとってください。 再起動スクリプト終了後に、対象プロセスの起動に時間がかかると、監視対象プロセスの起動前に監視が開始される可能性があります。 グループ監視の詳細は、製品添付資料の『ユーザーズガイド』『4.1.グループ監視の導入手順』を参照してください。

Q9.	監視定義ファイル（pfile）内に指定するプロセス名やファイル（SHM_DUMP_FILE など）のパスの指定にワイルドカード(*)を使用することは可能でしょうか？
A9.	監視定義ファイル（pfile）のパラメータ値にワイルドカード(*)を使用することはできません。

Q10.	監視定義ファイル（pfile）内に指定するプロセス名やファイル（SHM_DUMP_FILE など）のパスの指定を相対パスで指定することは、可能でしょうか？
A10.	パス指定は、絶対パスで設定してください。

Q11.	監視対象プロセス自身を、監視定義ファイル（pfile）の再起動スクリプト名に設定できますか？ 監視定義ファイル（pfile）の再起動スクリプト名に監視対象プロセスが起動するよう、直接指定することは可能ですか？
A11.	監視対象のプロセス自身を再起動スクリプトとして設定することはできません。 ProcessSaver の仕様として再起動スクリプトは、監視対象プロセスの起動後、必ず終了する必要があります。 監視対象プロセスを起動後、自身は終了するようなスクリプトを作成して、再起動スクリプトに設定してください。

Q12.	CLUSTERPRO MC StorageSaver のプロセスを監視したいのですが、どのような設定をすればよいでしょうか？
A12.	srgwatch プロセスを監視対象プロセスに設定してください。 監視対象のプロセス名として “/opt/HA/SrG/local/bin/srgwatch”、再起動スクリプトに “/etc/init.d/srgctl restart” を指定することで、監視、再起動を行うことができます。 [監視定義ファイル（pfile） の設定例] ##### PENT ##### /opt/HA/SrG/local/bin/srgwatch:/etc/init.d/srgctl restart:86400:3:continue srgwatch プロセスは、StorageSaver のデーモンプロセス（srgd）を監視しているため、srgwatch プロセス以外のプロセスを監視する必要はありません。

Q13.	プロセス名が同一であり引数の一部が同じであるプロセスをそれぞれ監視したいのですが、どのような設定をすればよいでしょうか？
A13.	pname_full_match オプションを使用することによりそれぞれのプロセスを監視できます。 pname_full_match オプションは、完全一致で対象プロセス名のマッチング処理を行います。 そのため、プロセス名は ps コマンドを “-ef” オプションで実行し、出力されたプロセス名をそのまま指定してください。 ただし、監視を行いたいプロセスが引数を含めすべて同名で複数起動する場合は、本設定でも監視を行うことはできませんので、必ず対象のプロセスが 1 つしか起動しないことをあらかじめ確認してください。 たとえば以下のようなプロセス名のプロセスが存在し、 “proc aaaa” の監視を行いたい場合。 ・ proc aaaa ・ proc aaaa_bbbb 監視定義ファイル（pfile）のオプション部に pname_full_match を指定し、引数を含めたプロセス名を process_name パラメータに指定することで “proc aaaa” と “proc aaaa_bbbb” を区別して監視を行うことができます。 [監視定義ファイル（pfile） の設定例] ##### PENT ##### proc aaaa:＜再起動スクリプト＞: 86400:3:continue:pname_full_match=enable pname_full_match オプションを指定しない場合のプロセス名（process_name パラメータ）の指定方法の注意点は以下です。 - 引数を省略した場合は、プロセス名が完全に一致する場合のみ監視対象とします。 - 引数を含めて指定した場合は、プロセス名は完全一致、引数部分については、前方一致で監視対象を選択します。

Q14.	監視定義ファイル（pfile） の個別情報部(PENT)の option に指定可能な文字列長を教えてください。
A14.	option 全体で指定できる文字列長は、最大で 255 文字です。 option についての詳細は、製品添付資料の『ユーザーズガイド』『3.3.pfile ファイルについて』を参照してください。

Q15.	監視定義ファイル（pfile）の include_strings オプションに、文字列 “=(イコール)” を指定することはできますか？
A15.	監視定義ファイル（pfile）の include_strings オプションに “=” を指定することはできません。 include_strings に指定可能な文字は、英数字とハイフン(“-”)、アンダースコア(“_”)、スラッシュ(“/”)、ドット(“.”) のみです。 また、半角スペースは指定することはできません。

Q16.	同一名プロセスが複数起動する状態でプロセス名（引数）に “=”（イコール） 含むプロセスの監視を行うため、include_strings オプションを使用したのですが、 include_strings オプションには、 “=”（イコール）を含む文字列を指定できません。 このような場合どのような設定を行えばいいのでしょうか？
A16.	監視対象のプロセス名（引数）に “=” が含まれる場合、監視定義ファイル （pfile） のプロセス名（process_name オプション）に引数を含めて監視対象のプロセス名を指定する。 もしくは、“=” を除き、プロセスの検索条件を複数で指定することにより “=” を含んだプロセスを監視することができます。 たとえば以下のようなプロセス名のプロセスを監視対象とする場合は、“&” 区切りで複数の文字列を記載することで監視することができます。 - proc_aaa BBB=CCC [監視定義ファイル （pfile） の設定例] ##### PENT ##### proc_aaa:＜再起動スクリプト＞:86400:3:continue:***include_strings= BBB&CCC***

1.4 再起動スクリプトについて

Q1.	監視定義ファイル（pfile）の再起動スクリプト名の値に su コマンドを指定することは可能ですか？
A1.	監視定義ファイル（pfile）の再起動スクリプトに su コマンドを直接指定することは可能です。 ただし、su コマンドを直接指定する場合は、以下の点に注意してください。 - 監視定義ファイル（pfile）の再起動スクリプトには、基本的にシェルスクリプトのファイル名を指定する仕様となっておりますので su コマンドなどのコマンドを直接指定する場合には、必ず動作検証を行った上で問題ないことを確認してから設定してください。 * 一般的には su コマンドを実行する場合、su コマンドを実行するスクリプトを作成し、それを再起動スクリプトに指定します。 - su コマンドは、“/bin/su”（絶対パス）で指定してください。 - su コマンドを使用する場合、必ず -c オプションを使用してください。 [コマンド実行例] /bin/su - ＜ユーザ名＞ -c “＜実行コマンド＞” - su コマンドを使用してコマンドを実行する場合、以下のように実行するコマンドに引数を指定することはできません。 [コマンド実行例] # /bin/su - ＜ユーザ名＞ -c “＜実行コマンド＞ ＜実行コマンドの引数＞” * このような場合には、su コマンドを使用してコマンドを実行するスクリプトを作成し、それを再起動スクリプトとして記述する方式をご利用ください。 【 次ページに続く 】

	【 前ページからの続き 】 [再起動スクリプトの記述例]（ファイル名： /var/opt/HA/PS/conf/bin/su_start.sh） #! /bin/sh /bin/su - ＜ユーザ名＞ -c “＜実行コマンド＞ ＜実行コマンドの引数＞” exit 0 [監視定義ファイル（pfile）の設定例] ##### PENT ##### proc1:/var/opt/HA/PS/conf/bin/su_start.sh:86400:3:continue

Q2.	対象プロセス自身が自動で再起動するプロセスの場合、再起動スクリプトにはなにを指定すればよいですか？
A2.	監視対象プロセスが、起動するための時間を確保するために sleep コマンドなどで待ち合わせのみを行う再起動スクリプトを指定してください。 再起動スクリプトを指定しない場合、監視対象プロセスの異常終了を検出後、すぐに監視を再開します。 [再起動スクリプトの記述例] #! /bin/sh /bin/sleep 60 exit 0 * 下線部分の sleep コマンドの引数にしている値は、十分な待ち合わせ時間を指定してください。

1.5 監視機能について

Q1.	ProcessSaver のプロセス自身 (pcheck) を監視することは可能ですか？
A1.	pcheck の監視を行う場合は、別の pcheck で監視してください。 pcheck は、自身の終了を検知する機能はありません。 pcheck を別の pcheck で監視する方法の詳細については、製品添付資料の『ユーザーズガイド』『2.3.2.pcheck を階層的に使用したグループ監視』を参照してください。

Q2.	ProcessSaver では、プロセスの監視をどのような方法で実施していますか？
A2.	“/proc” を使用しプロセス情報を取得します。 pcheck 起動時や監視対象プロセスの再起動後は、/proc/<PID>/ ディレクトリに存在するシステム上の全 PID 情報を取得し、取得した全プロセス情報から監視対象のプロセス情報（PID など）を保持します。 監視対象プロセスの PID の保持後は、保持している PID に合致する /proc/<PID>/ へアクセスしてプロセスの状態監視を行います。

Q3.	pcheck 起動時、監視はいつから開始されるのですか？
A3.	pcheck を起動後、監視定義ファイル（pfile）に設定された MONITOR_INTERVAL (秒)経過後に監視を開始します。 ただし、pcheck が起動し MONITOR_INTERVAL (秒)経過した時点で監視対象プロセスが起動していない場合は、一定時間（デフォルトは 60 秒）自動的に監視開始を待ち合わせます。 また、待ち合わせ時間内に監視対象プロセスの起動を確認できた時点で監視を開始します。 環境変数 (HAPS_PENDING_TIME) を設定することで、待ち合わせ時間を変更することが可能となります。 詳細は、製品添付資料の『ユーザーズガイド』『4.10.pcheck 起動時の自動待ち合わせ時間を変更する手順』を参照してください。

Q4.	pcheck プロセスを -w オプションを指定して起動した場合、pcheck プロセス自身はすぐに起動しますか？
A4.	pcheck プロセスは、すぐに起動します。 -w オプションを指定して起動した場合 pcheck は、すぐに起動し pcheck 内部で待ち合わせを行います。 なお、-w オプションで指定された秒数の待ち合わせを行っている間は、padmin コマンドでの制御や状態表示を行うことはできません。

Q5.	監視定義ファイル（pfile）の GRACE 値 (grace_time) の計測を開始するタイミングは、いつですか？
A5.	GRACE 値の計測は最初のプロセスの消滅検出時から開始されます。 計測を開始した時点からGRACE 値（秒）経過までに、再起動リトライ回数の上限に達しなければ GRACE 値のカウントとリトライ回数は、ゼロ(0)クリアされます。 詳細は、製品添付資料の『ユーザーズガイド』『3.3.pfile について』を参照してください。

Q6.	同一名（同じ名称）のプロセスが複数個存在する環境の場合、ProcessSaver はどのようなルールで監視対象プロセスを選択しますか？
A6.	監視対象となるプロセスが同一名（同じ名称）で複数起動していた場合は、以下のルールに則って 1 つのプロセスのみを監視します。 1. プロセス間に親子関係がある場合は、大元の親プロセスを監視します。 2. プロセス間に親子関係がなければ、プロセスの起動時刻のもっとも古いものを監視します。 3. プロセス間に親子関係がなく、プロセスの起動時刻も同じであれば、もっとも PID の小さいものを監視します。

1.6 ログファイルについて

Q1.	syslog へのメッセージ出力を停止することは、可能ですか？
A1.	pcheck をサイレントモードで起動することにより、pcheck が出力するすべての syslog へのメッセージ出力を停止できます。 環境変数 (HAPS_SILENT_MODE) に “1” を設定し、pcheck を起動することによりサイレントモードで動作します。 詳細は、製品添付資料の『ユーザーズガイド』『4.8.サイレントモードでの運用手順』を参照してください。

Q2.	syslog へのメッセージ出力のファシリティ・レベルを変更することは可能ですか？
A2.	ProcessSaver は、ファシリティ・レベルを変更する機能はありません。 ProcessSaver が出力する syslog のファシリティとレベルは以下となります。 - ファシリティ : LOG_USER - レベル : LOG_ERR または LOG_WARNING

Q3.	監視対象プロセスが存在しない場合に syslog へ出力されるメッセージについて教えてください。
A3.	監視対象プロセスが存在しない場合に出力されるメッセージは、以下の2種類となります。 - Set pid fail (xxx) 監視開始時に監視対象プロセスが存在しない場合に出力されます。 - Process [xxx, pid=yyy] Down 監視中に監視対象プロセス（PID）の消滅を検出した場合に出力されます。

Q4.	運用管理製品 (WebSAM SystemManager など) で ProcessSaver が出力するログメッセージの監視を行う場合に監視対象とすべきメッセージを教えてください。
A4.	syslog へ出力される監視対象プロセスの復旧不可能を示すエラーメッセージを監視対象メッセージとして指定してください。 以下のメッセージは、対象プロセスの消滅を検出し、一定回数リトライしても再起動できない状態です。 - 監視対象プロセスの復旧不可能を示すメッセージ ・ RETRY_OVER:CONTINUE ・ RETRY_OVER:SHUTDOWN ・ RETRY_OVER:EXIT なお、主要な運用管理製品についてはメッセージ監視設定サンプル（テンプレート）を用意しておりますのでそちらをご利用ください。

Q5.	/var/tmp 配下に “.pcheck.xxxx”（例：.pcheck.520094465）というファイルが存在しているのですが、このファイルを削除しても構わないのでしょうか？
A5.	/var/tmp 配下に存在する “.pcheck.xxxx” ファイルは、ProcessSaver の管理ファイルとなりますので、削除しないでください。 管理ファイルは、起動している pcheck ごとに作成されます。 このファイルを削除した場合は、対応する pcheck が 運用管理コマンドの一覧表示（padmin –l）を実行しても表示されませんのでご注意ください。 ファイル名の後方の数字は、 pfile に設定した IPCKEY（16 進数）を 10 進数に変換した値です。

Q6.	pcheck 自身の異常終了を syslog などで確認することはできますか？
A6.	異常を示すメッセージが syslog に出力されますので確認することができます。 ただし、pcheck が異常終了に至った原因により syslog などで確認することはできない場合があります。 内部バグなどの内的要因で予期せず終了した場合や、外部からのシグナル（SIGKILL）を受信した場合などの外的要因で pcheck が終了した場合は、一般のプロセスと同様に syslog メッセージは出力されません。 pcheck が、ファイル I/O の異常やメモリ不足などで終了した場合は、異常を示すメッセージを syslog に出力します。 メッセージの詳細については、「syslog メッセージ一覧」を参照してください。

2　構築段階における質問

2.1　　起動、終了について

Q1.	リモートシェルで pcheck を起動しようとしましたが、コマンドに応答がありません。 なにが原因でしょうか？
A1.	リモート側の標準入力／標準出力／標準エラー出力がクローズされていない可能性があります。 リモートシェルを使用して実行するコマンドをバックグラウンド起動する場合には、リモート側の標準入力／標準出力／標準エラー出力を明示的にクローズする処理を行う必要があります。 リモートシェルで pcheck を起動する場合、以下のようにコマンドラインを指定し実行してください。 [rsh での実行例] # rsh –l ＜ユーザ名＞ ＜ホスト名＞ “command 1>&- 2>&- &”

Q2.	監視対象プロセスの起動コマンドを実行後に pcheck を起動しましたが、 syslog に以下のメッセージが出力されて監視が正常に行えません。 なにが原因でしょうか？ pcheck[XXX]: Set pid fail (xxx)
A2.	このメッセージは、pcheck 監視開始時に監視対象プロセスが存在しない場合に出力されます。 Oracle や Java アプリケーションなどの起動に時間がかかるプロセスの場合、起動が完了する前に pcheck の監視が開始されている可能性があります。 アプリケーションが確実に起動したことを確認してから pcheck の監視を開始するようにしてください。 監視対象プロセスおよび pcheck を自動起動している場合には、以下の方法が有効です。 - pcheck の自動待ち合わせ昨日を使用し、監視対象プロセスが起動するまで一定時間自動的に待ち合わせを行う。 詳細は、製品添付資料の『ユーザーズガイド』『4.10. pcheck 起動時の自動待ち合わせ時間を変更する手順』を参照してください。 - pcheck に -w オプションを付与して起動し、pcheck 起動後に一定時間待ち合わせてから監視を開始する。 [30 秒後に監視を開始する場合の例] # /opt/HA/PS/bin/pcheck -f ＜pfile 名＞ -w 30

Q3.	pcheck が起動できない、もしくは起動するがすぐに終了してしまいます。 なにが原因でしょうか？
A3.	syslog に pcheck 関連のメッセージが出力されていないか確認してください。 以下に代表的なメッセージとそのメッセージに対する原因および対処方法を記載します。 1. pfile init fail. pfile = ‘pfile 名’ [原因] pfile の初期化に失敗しました。 [対処] pfile の定義に不備がある可能性がありますので直前のログメッセージを確認し、対処してください。 2. fopen(3) fail. pfile = ‘pfile 名’ [原因] pcheck 起動時に指定された pfile を開くことができません。 [対処] ・指定したパスに pfile が存在するか確認してください。 ・pfile が存在する場合、パーミッションや所有権が正しいか確認してください。 3. pcheck is already execute. [原因] 指定された pfile 名で pcheck がすでに起動しているか、他の pfile とIPCKEY (共有メモリキー) が重複している可能性があります。 [対処] すでに起動中の pcheck を停止し、起動してください。 もしくは、同じ IPCKEY を使用している場合は IPCKEY を変更し pcheck を起動してください。 【 次ページ に続く 】

【 前ページからの続き 】

4. shmget(2) fail. errno = ‘エラーNo’

[原因]

共有メモリの確保に失敗しました。

[対処]

空きメモリが不足している可能性があります。

ProcessSaver の動作に必要なメモリを確保してから pcheck を起動してください。

5. illegal option.

[原因]

無効なオプションが指定されました。

[対処]

マニュアルを確認し正しいコマンドを再投入してください。

その他のメッセージについては、製品添付資料の『syslog メッセージ一覧』を参照してください。

Q4.	pcheck を起動しましたが syslog に以下のメッセージが出力されて監視が正常に行えません。なにが原因でしょうか？ pcheck[XXX]: shmget(2) use other process.
A4.	監視定義ファイル（pfile）の IPCKEY で指定した共有メモリキーが他のプロセスですでに使用されている可能性があります。 IPCKEY を OS 内で一意となる値に変更したあと、pcheck の再起動を行ってください。 [監視定義ファイル（pfile）の設定例] ##### PARAM ##### IPCKEY 0x1f000001 * IPCKEY は、0x00000001～0x7fffffff の値が指定できます。 ただし、IPCKEY を小さい値にした場合、他のプロセスが使用する共有メモリキーと重複する可能性が高くなります。 IPCKEY の値は 0x1f000000 以上の比較的大きな値を指定することを推奨します。 * 現在使用中の共有メモリキーの一覧は、以下のコマンドで確認することができます。 # ipcs

Q5.	ProcessSaver のストール監視を行うよう設定をしているのですが、pcheck の起動後すぐにストールを検知してしまいます。 なにが原因でしょうか？

A5.	ストール監視は、指定されたファイルのタイムスタンプと現在の時刻とを比較してプロセスがストールしているか判断するため、pcheck の初期起動時にストール監視の対象ファイルのタイムスタンプが古い場合、ストールと判断する時間を経過していなくてもストールと判断してしまう可能性があります。
	
	pcheck 起動時には、監視対象となるファイルのタイムスタンプを必ず更新してください。
	
	たとえば以下のように監視定義ファイル（pfile）の INIT 処理で touch コマンドを使って pcheck 起動時に明示的にファイルのタイムスタンプを更新する処理を指定することで対処することができます。
	
	[監視定義ファイル（pfile）の設定例]
	_bi_stall {
	PS_INIT AP: /bin/touch <ストール監視対象ファイル名>: DIRECT:-:-:-:0
	PS_EXEC DLL:/opt/HA/PS/lib/lib_bi_stall.so:DIRECT:-:60:0:2:<ストール監視対象ファイル名>:180
	PS_ACTION DLL:/opt/HA/PS/lib/lib_bi_stall.so:DIRECT:-:-:1:0
	}
	＊ 下線の行を pfile に追加してください。
	
	詳しくは、製品添付資料の『ユーザーズガイド』『4.6.ストール監視の導入手順』を参照してください。

Q6.	pcheck で Oracle のリスナープロセス (tnslsnr) を監視しているのですが、対象のプロセスが存在するにもかかわらず消滅を検知しました。なにが原因でしょうか？
A6.	Oracle リスナープロセス (tnslsnr) の監視を行う場合、Oracle の接続要求を受け付けた瞬間に pcheck が監視を開始するとプロセス消滅を検知してしまう場合があります。 これは、監視対象の Oracle リスナープロセス (tnslsnr) は、接続要求を受け付けた瞬間に同名のプロセスを最大 2 個生成する仕様となっており、その生成されたプロセスはその後すぐに消滅します。 同名プロセスが複数存在した場合 pcheck は、ルールに則って 1 つのプロセスのみを監視対象プロセスとして選択します。 そのため、すぐに消滅する Oracle リスナープロセス (tnslsnr) を監視対象プロセスとして選択し、監視開始しプロセスの異常終了を検知すると本事象のような動作となります。 対処としては Oracle への接続要求がない状態で Oracle リスナープロセス (tnslsnr) の監視を開始してください。

2.2 再起動スクリプトについて

Q1.	プロセスの消滅を検知後、再起動スクリプトの実行に失敗しているようですが、なにが原因でしょうか？
A1.	syslog に出力されたメッセージを確認し対処してください。 再起動スクリプト失敗時の代表的な syslog メッセージは以下があります。 1. 再起動スクリプトが存在しない場合 pcheck[xxx]: -> execvp(2) fail (2) pcheck[xxx]: -> restart shell fail (exit status) (233) 監視定義ファイル（pfile）の再起動スクリプト名に指定したファイルが、存在するか確認してください。 2. 再起動スクリプトに実行権がない場合 pcheck[xxx]: -> execvp(2) fail (13) pcheck[xxx]: -> restart shell fail (exit status) (233) 再起動スクリプトに pcheck 起動ユーザの実行権が付与されているかどうか確認してください。 3. 再起動スクリプトの構文エラーの場合 pcheck[xxx]: -> restart shell fail (exit status) (127) 再起動スクリプトの構文が正しくない（syntax error）可能性があります。 再起動スクリプトが単体で実行できるか確認してください。

Q2.	再起動スクリプト実行時の再起動スクリプトの戻り値が “0” と “0”以外の場合 pcheck の処理の違いはありますか？
A2.	“0” 以外の場合のみ、監視対象プロセスの再起動に失敗したことを示すメッセージを syslog に出力します。 それ以外の処理の違いはありません。 再起動スクリプトの終了後は、その結果の如何にかかわらず監視を再開します。 なお、再起動スクリプトが正しく実行できたことを確認するために再起動スクリプト成功時は、“0” を返却し、再起動スクリプト失敗時は、“0” 以外を返却するよう再起動スクリプトを作成することを推奨しています。

Q3.	再起動スクリプトが実行されても、監視対象プロセスが起動できないのですが、なにが原因なのかわかりません。 再起動スクリプトの内容確認を依頼することは可能ですか？
A3.	ProcessSaver は、監視対象に指定したプロセスが異常終了した場合に再起動スクリプトの実行及び syslog への通知を行う製品となっていますので、再起動スクリプトにて監視対象プロセスが起動できていない原因までは当方で調査することができません。 そのため、監視対象プロセス側の観点で再起動スクリプトを確認してください。 例えば、正しく再起動が行えるか確認するために作成した再起動スクリプトを手動で実行することで正しく監視対象プロセスが起動することを確認することが有効です。 再起動スクリプトの作成につきましては、製品添付資料の『ユーザーズガイド』の『3.4.再起動スクリプトについて』を参照してください。

2.3 構成変更について

Q1.	ProcessSaver をインストール後、 IP アドレス、ホスト名が変更となった場合、なにか処置が必要でしょうか？
A1.	すでに登録していただいているコードワードに影響がある場合があります。 コードワードの取得時にご提供いただいたマシン情報により、対処が異なります。 - ホスト ID の場合 ホスト名や IP アドレスが変更された場合は、ホスト ID が変更される場合がありますので変更後のホスト ID を確認し、コードワード申請時と異なる場合は、コードワードの再取得、再登録が必要です。 - MAC アドレスの場合 特に対処は必要ありません。 どちらの場合も ProcessSaver 本体の再起動、再インストールなどの処置は不要です。 ただし IP アドレス、ホスト名の変更により、ProcessSaver で監視対象としているプロセスのプロセス名が変更となる場合には、変更されたプロセス名にあわせて監視定義ファイル（pfile）の設定を修正していただく必要があります。 コードワードの変更手順は製品添付の『コードワードについて』の手順を参照してください。

Q2.	評価用のコードワードから正式コードワードに変更する場合、ProcessSaver でなにか処置が必要でしょうか？
A2.	正式コードワードの再登録以外には、特に必要ありません。 ProcessSaver 本体の再起動、再インストールなどの処置は不要です。 コードワードの変更手順は製品添付の『コードワードについて』の手順を参照してください。

Q3.	仮想化環境において ProcessSaver がすでに設定済みの仮想マシンをベースにし、別の仮想マシンを作成（コピー）した場合、なにか処置が必要でしょうか？
A3.	作成（コピー）した仮想マシンごとにコードワードの取得、登録が必要です。 また、サーバのコピーによって、ProcessSaver で監視対象としているプロセスのプロセス名が変更となる場合には、変更されたプロセス名にあわせて監視定義ファイル（pfile）の設定を修正していただく必要があります。 コードワードについての詳細は、製品添付資料の『コードワードについて』を参照してください。

3　運用段階における質問

3.1　起動、終了について

Q1.	pcheck が起動できない、もしくは起動するがすぐに終了してしまいます。 なにが原因でしょうか？
A1.	syslog に pcheck 関連のメッセージが出力されていないか確認してください。 以下に代表的なメッセージとそのメッセージに対する原因および対処方法を記載します。 1. pfile init fail. pfile = ‘pfile 名’ [原因] pfile の初期化に失敗しました。 [対処] pfile の定義に不備がある可能性がありますので直前のログメッセージを確認し、対処してください。 2. fopen(3) fail. pfile = ‘pfile 名’ [原因] pcheck 起動時に指定された pfile を開くことができません。 [対処] ・指定したパスに pfile が存在するか確認してください。 ・pfile が存在する場合、パーミッションや所有権が正しいか確認してください。 【 次ページ に続く 】

【 前ページからの続き 】

3. pcheck is already execute.

[原因]

指定された pfile 名で pcheck がすでに起動しているか、他の pfile とIPCKEY (共有メモリキー) が重複している可能性があります。

[対処]

すでに起動中の pcheck を停止し、起動してください。

もしくは、同じ IPCKEY を使用している場合は IPCKEY を変更し pcheck を起動してください。

4. shmget(2) fail. errno = ‘エラーNo’

[原因]

共有メモリの確保に失敗しました。

[対処]

空きメモリが不足している可能性があります。

ProcessSaver の動作に必要なメモリを確保してから pcheck を起動してください。

5. illegal option.

[原因]

無効なオプションが指定されました。

[対処]

マニュアルを確認し正しいコマンドを再投入してください。

その他のメッセージについては、製品添付資料の『syslog メッセージ一覧』を参照してください。

Q2.	OS 起動時 pcheck を自動起動したのですが、起動後しばらくすると pcheck が終了してしまいます。 なにが原因でしょうか？
A2.	OS 起動時に pcheck を自動起動する場合、rc スクリプトを作成・登録し自動起動をおこいますが、OS の自動起動処理では、処理の最後にシグナルを送り無用な残存プロセスの停止を行います。 これを受け付けないように rc スクリプトに trap 処理を追加する必要があります。 以下のように rc ファイルに trap 処理を行う行を追加してください。 [rc ファイル記述例] #!/bin/sh # ***trap "" 1 2 3 13 15*** *下線の行を rc スクリプトに追加してください。 詳細は、製品添付資料の『ユーザーズガイド』『3.6.起動、終了ファイルの導入について』を参照してください。

Q3.	syslog に以下のようなメッセージが出力され、pcheck が異常終了してしまいます。なにが原因でしょうか？ lockf(F_LOCK) fail. errno = xx
A3.	システム上のロックファイルの総数が、カーネルパラメータの file locks を超えており、pcheck がファイルのロックを行えないため、pcheck が異常終了しています。 他の製品で使用しているロックを解除するか、file locks のリソース制限を拡張してください。 リソース制限値は、以下のコマンドで確認できます。 [sh もしくは bash の場合] # ulimit –a [csh もしくは tcsh の場合] # limit * マシン（OS） のリソースが不足している状態では、正しく動作しない可能性があるため ProcessSaver の動作の保障ができません。 リソース（ファイルロック数・メモリ・空き容量など）不足にならないようにマシン（OS）の調整を行ってください。

Q4.	監視対象プロセスの起動コマンドを実行後にpcheck を起動しましたが、 syslog に以下のメッセージが出力されて監視が正常に行えません。 なにが原因でしょうか？ pcheck[XXX]: Set pid fail (xxx)
A4.	このメッセージは、pcheck 監視開始時に監視対象プロセスが存在しない場合に出力されます。 Oracle や Java アプリケーションなどの起動に時間がかかるプロセスの場合、起動が完了する前に pcheck の監視が開始されている可能性があります。 アプリケーションが確実に起動したことを確認してから pcheck の監視を開始するようにしてください。 監視対象プロセスおよび pcheck を自動起動している場合には、以下の方法が有効です。 - pcheck の自動待ち合わせ昨日を使用し、監視対象プロセスが起動するまで一定時間自動的に待ち合わせを行う。 詳細は、製品添付資料の『ユーザーズガイド』『4.10. pcheck 起動時の自動待ち合わせ時間を変更する手順』を参照してください。 - pcheck に -w オプションを付与して起動し、pcheck 起動後に一定時間待ち合わせてから監視を開始する。 [30 秒後に監視を開始する場合の例] # /opt/HA/PS/bin/pcheck -f ＜pfile 名＞ -w 30

Q5.	pcheck の起動時に監視対象プロセスが存在するにもかかわらず、プロセスの消滅を検知し、以下のメッセージを syslog に出力しました。なにが原因でしょうか？ pcheck[XXX]: Set pid fail (xxx)
A5.	起動しているプロセスが大量かつ、プロセスの起動 ・ 停止が頻繁に発生している場合、稀に監視対象プロセスの情報が OS のプロセス情報から取得できない場合があります。 本事象が発生する環境では次のような対処をすることで事象の発生を抑えることができます。 環境変数 HAPS_GETPROC_COUNT を現在起動しているプロセス総数以上に設定し、pcheck を再起動してください。 ・ 起動しているプロセス総数が約 28000 個の場合は、HAPS_GETPROC_COUNT を 30000 に設定してください。 * HAPS_GETPROC_COUNT を設定しない場合、HAPS_GETPROC_COUNT が 1000 で動作します。 * HAPS_GETPROC_COUNT を増やすほど、pcheck 内部で一時的に確保するメモリ量が増えます。例えば 15000 プロセス指定時は約 15 MB 程度、pcheck 起動時などの監視対象プロセス特定処理時にのみ一時的に増加します。 環境変数 HAPS_GETPROC_COUNT の詳細は、製品添付資料の『ユーザーズガイド』『4.12. 監視対象選択時のプロセス情報取得量を変更する手順』を参照してください。

Q6.	すべての監視を停止した状態で、pcheck を起動することは可能でしょうか？
A6.	可能です。 以下の方法で行うことができます。 [設定方法] ProcessSaver の起動前に、あらかじめ監視停止した状態で起動させるための制御ファイルを作成しておきます。 # touch /opt/HA/PS/bin/pcheck.ignore [確認方法] ProcessSaver を起動すると、syslog に監視停止中のメッセージが表示されます。 [出力メッセージ] syslog: pcheck wait... by /opt/HA/PS/bin/pcheck.ignore [解除方法] 監視を再開する場合は制御ファイルを削除します。 # rm /opt/HA/PS/bin/pcheck.ignore * pcheck が起動した状態で制御ファイルを削除すると最大 60 秒後に監視を開始します。 ただし、複数の pcheck を起動する場合は、すべての pcheck の監視が停止状態で起動されますので注意が必要です。 特定の pcheck のみを監視停止状態で起動することはできません。

Q7.	pcheck の監視を停止するための制御ファイル (/opt/HA/PS/bin/pcheck.ignore) を pcheck 実行中に作成した場合、監視は停止されるのでしょうか？
A7.	監視は停止されません。 pcheck 起動時に制御ファイル(/opt/HA/PS/bin/pcheck.ignore) が作成されている場合のみ、監視を停止します。 pcheck 実行中に制御ファイル(/opt/HA/PS/bin/pcheck.ignore)を作成した場合は、次回の pcheck 起動時から有効となります。 制御ファイル(/opt/HA/PS/bin/pcheck.ignore) を作成後に起動した pcheck は、rm コマンドなどで一時ファイルを削除しない限り監視停止状態を継続します。

Q8.	pcheck で Oracle のリスナープロセス (tnslsnr) を監視しているのですが、対象のプロセスが存在するにもかかわらず消滅を検知しました。なにが原因でしょうか？
A8.	Oracle リスナープロセス (tnslsnr) の監視を行う場合、Oracle の接続要求を受け付けた瞬間に pcheck が監視を開始するとプロセス消滅を検知してしまう場合があります。 これは、監視対象の Oracle リスナープロセス (tnslsnr) は、接続要求を受け付けた瞬間に同名のプロセスを最大 2 個生成する仕様となっており、その生成されたプロセスはその後すぐに消滅します。 同名プロセスが複数存在した場合 pcheck は、ルールに則って 1 つのプロセスのみを監視対象プロセスとして選択します。 そのため、すぐに消滅する Oracle リスナープロセス (tnslsnr) を監視対象プロセスとして選択し、監視開始しプロセスの異常終了を検知すると本事象のような動作となります。 対処としては Oracle への接続要求がない状態で Oracle リスナープロセス (tnslsnr) の監視を開始してください。

Q9.	pcheck が突然終了しました。 syslog にメッセージも出力されていません。なにが原因でしょうか？
A9.	通常 pcheck プロセスの終了時は、必ず syslog にメッセージを出力します。 メッセージが出力されないケースとしては外部からのシグナル（SIGKILL）受信による強制停止が考えられます。 以下のケースに合致していないか確認してください。 - オペレータからの kill -9 コマンド発行や業務スクリプトなどからの kill -9 コマンド発行を行っており、本来意図しないプロセス (pcheck) の強制停止を行っている。 - pcheck を起動するためにミドルウェア製品を利用していた場合、ジョブなどの終了時に、それまでに起動したプロセス (pcheck) に対して SIGKILL を送信し、残存プロセスの回収処理を行っている。 - ファイルシステムをアンマウントする前の処理として fuser –k コマンドを実行しており、そのため該当ディレクトリ配下にて起動しているプロセス (pcheck) に対して SIGKILL が発行されている。

Q10.	pcheck が以下のメッセージを syslog に出力し終了しました。 なにが原因でしょうか？ Something wrong with Sharedmemory(incorrect ipckey)
A10.	本メッセージは、ProcessSaver が使用している共有メモリの情報が不正な場合に出力されます。 ProcessSaver が使用している共有メモリの情報が手動で削除されたもしくは、他のアプリケーションにより情報を書き換えられた可能性があります。 他のアプリケーションと共有メモリが競合している可能性があるため、IPCKEY を再調整していただく必要があります。 * IPCKEY を小さい値にした場合、他のプロセスが使用する共有メモリキーと重複する可能性が高くなると考えられます。IPCKEY の値は 0x1f000000 以降の大きな値を指定することを推奨します。

3.2 運用管理コマンド（padmin）について

Q1.	padmin コマンド -l オプションで pcheck の一覧表示を行いましたが、なにも表示されません。 なにが原因でしょうか？
A1.	以下の 2 つの原因が考えられます - pcheck が起動していない。 - ProcessSaver 管理ファイルが存在しない。 まず原因を切り分けるために ps コマンドを実行し起動中の pcheck プロセスが表示されるか確認してください。 以下のコマンドを実行してください。 # ps -ef | grep pcheck * grep コマンドの引数は、システムにあったプロセス名を指定してください。 [pcheck プロセスが表示されない場合] pcheck が起動していません。 pcheck を起動してから、padmin コマンド -l オプションを実行し確認してください。 [pcheck プロセスが表示された場合] ProcessSaver 管理ファイルが削除されている可能性があります。 管理ファイルが削除されている場合には、運用管理コマンドの一覧表示（padmin –l）を実行しても表示されません。 通常は、削除されることのないファイルのため手動での削除などなんらかの原因で削除されたと考えられます。 ただし、プロセス監視自体には影響ありません。 以下のどちらかの手順を実行することで一覧に表示されるようになります。 ・ pcheck を再起動する。 ・ pcheck の reload を実行する。 # /opt/HA/PS/bin/padmin -f ＜pfile 名＞ -c reload

Q2.	padmin コマンド実行時、コンソール画面に以下のエラーメッセージが出力されました。 なにが原因でしょうか？ Can not access SharedMemory(incorrect ipckey)
A2.	このメッセージは、padmin コマンド実行時に対象の pcheck プロセスが共有メモリを確保していない状態でアクセスした場合に出力されます。 以下のケースに合致していないか確認し、対処してください。 - 対象の pcheck プロセスが存在しない状態で、padmin コマンドを実行した場合 対象の pcheck プロセスが起動しているか確認してください。 起動していない場合は pcheck を起動後、再度 padmin コマンドを実行してください。 - 対象の pcheck の監視を待ち合わせている状態で、padmin コマンドを実行した場合 対象の pcheck が -w オプション指定で起動された場合、pcheck プロセスは指定された待ち合わせ時間後に共有メモリを確保するため、待ち合わせ時間中は、padmin コマンドを実行することができません。待ち合わせ時間経過後、padmin コマンドを再実行してください。 - 対象の pcheck が制御ファイル(/opt/HA/PS/bin/pcheck.ignore)作成による監視停止の状態で padmin コマンドを実行した場合 制御ファイル(/opt/HA/PS/bin/pcheck.ignore) が存在する状態で起動された pcheck プロセスは、起動後に共有メモリは確保せずに監視停止状態となっています。このため監視停止状態の間は、padmin コマンドを実行することができません。制御ファイルを削除し、pcheck を監視状態としたあとに padmin コマンドを再実行してください。

Q3.	監視定義ファイル（pfile）のプロセス（pent）単位でリトライ回数をリセットすることはできますか？
A3.	監視定義ファイル（pfile）のプロセス（pent）単位でリトライ回数をリセットする方法はありません。 padmin コマンド -c reload オプションを実行することで監視定義ファイル（pfile）の再読み込みを実施して、監視定義ファイル（pfile）に定義されたすべてのプロセス（pent）のリトライ回数をクリアすることは可能です。 [実行例] # /opt/HA/PS/bin/padmin -f ＜pfile 名＞ -c reload

Q4.	padmin コマンドを実行したところ、以下のメッセージがコンソール画面に出力されました。 なにが原因でしょうか？ Operation timeout.
A4.	padmin コマンドが時間内に終了できずにタイムアウトしたことを示すメッセージです。一時的な高負荷やリソース不足が原因で発生する可能性があります。 padmin コマンドは、一時的な高負荷やリソース不足が原因で反映に時間がかかってしまう場合に制御が戻らない状態となることを防止するために、タイムアウト値を設定しています。 本事象が発生した場合でも pcheck での監視は、正常に継続されます。 pcheck で処理が継続されていることを確認するため padmin コマンドの show pent オプションなどを実行し、pcheck の状態を確認してください。 確認した結果、padmin コマンドで要求した処理が反映できていない場合は padmin コマンドを再実行することで対処することが可能です。

Q5.	ps コマンドで pcheck プロセスを確認しましたが、存在しない状態でした。 その状態で padmin コマンド -l オプション(起動中の pcheck 一覧を表示する)を実行すると pcheck の情報が表示されました。 pcheck プロセスが起動していないのに表示されるのはなにが原因でしょうか？
A5.	なんらかの原因で pcheck プロセスが強制終了されたために ProcessSaver の管理ファイルと pcheck で使用した共有メモリが残ったままになっていることが原因です。 その場合に padmin -l コマンドを実行すると以下のような表示結果となります。 [表示例] # /opt/HA/PS/bin/padmin -l ipckey = 0x1f000100 mypid = --- myname = --- pfile = /var/opt/HA/PS/conf/bin/pfile_os message = start このような状態を解消するためには、残った共有メモリを削除するために以下の処理を実施してください。 # ipcrm -M ***IPCKEY*** ***IPCKEY*** は、監視定義ファイル（pfile）内の IPCKEY 設定している値(16 進数)を指定してください。

Q6.	padmin コマンドの -c stop オプションで監視を一時停止したあと、監視対象プロセスを停止しました。 その後、監視対象プロセスを起動せずに padmin コマンドの restart オプションで監視を再開しましたが ProcessSaver で監視対象プロセスの異常検知、再起動が行われていません。 なにが原因でしょうか？
A6.	監視対象プロセスが存在しない状態で padmin コマンドの -c restart オプションを実行した場合は、監視が再開されません。 padmin コマンドの restart オプションで監視を再開する場合は、監視対象プロセスが起動されていることを確認してから実施してください。

Q7.	ある監視対象プロセスが連続して異常となり、リトライオーバとなりました。 その後、監視対象プロセスの問題を取り除き、正常に起動できました。 この状態で ProcessSaver の監視を再開するために padmin コマンドの –c restart オプションを実行しましたが、監視が開始されません。 なにが原因でしょうか？
A7.	リトライオーバとなった監視対象プロセスの監視を再開する場合は、padmin コマンドの -c reload オプションで再開するか、pcheck の再起動を行ってください。 padmin コマンドの各オプションについては、製品添付資料の『ユーザーズガイド』『5.2.プロセスの運用管理について』を参照してください。

3.3 構成変更について

Q1.	ProcessSaver をインストール後、 IP アドレス、ホスト名が変更となった場合、なにか処置が必要でしょうか？
A1.	すでに登録していただいているコードワードに影響がある場合があります。 コードワードの取得時にご提供いただいたマシン情報により、対処が異なります。 - ホスト ID の場合 ホスト名や IP アドレスが変更された場合は、ホスト ID が変更される場合がありますので変更後のホスト ID を確認し、コードワード申請時と異なる場合は、コードワードの再取得、再登録が必要です。 - MAC アドレスの場合 特に対処は必要ありません。 どちらの場合も ProcessSaver 本体の再起動、再インストールなどの処置は不要です。 ただし IP アドレス、ホスト名の変更により、ProcessSaver で監視対象としているプロセスのプロセス名が変更となる場合には、変更されたプロセス名にあわせて監視定義ファイル（pfile）の設定を修正していただく必要があります。 コードワードの変更手順は製品添付の『コードワードについて』の手順を参照してください。

Q2.	ProcessSaver をインストール済みのサーバに接続されているストレージ構成を変更した場合、なにか処置が必要でしょうか？
A2.	ProcessSaver は、ストレージ構成が変更になった場合でも特に影響はありません。 ProcessSaver 本体の再起動、再インストールなどの処置は不要です。

4　主なメッセージ一覧

4.1　syslog メッセージ

メッセージ	参照場所
execvp(2) fail (xx).	『2.2 再起動スクリプトについて』『Q1』
fopen(3) fail. pfile = xxx.	『2.1 起動、終了について』『Q3』 『3.1 起動、終了について』『Q1』
illegal option.	『2.1 起動、終了について』『Q3』 『3.1 起動、終了について』『Q1』
lockf(F_LOCK) fail. errno = xxx.	『3.1 起動、終了について』『Q3』
pcheck is already execute.	『2.1 起動、終了について』『Q3』 『3.1 起動、終了について』『Q1』
pfile init fail. pfile = xxx.	『2.1 起動、終了について』『Q3』 『3.1 起動、終了について』『Q1』
Process [xxx, pid=yyy] Down.	『1.6 ログファイルについて』『Q3』
restart shell fail (exit status) (xxx).	『2.2 再起動スクリプトについて』『Q1』
Set pid fail (xxx).	『1.6 ログファイルについて』『Q3』 『2.1 起動、終了について』『Q2』 『3.1 起動、終了について』『Q4』 『3.1 起動、終了について』『Q5』
shmget(2) fail. errno = xxx.	『2.1 起動、終了について』『Q3』 『3.1 起動、終了について』『Q1』
shmget(2) use other process.	『2.1 起動、終了について』『Q4』
Something wrong with Sharedmemory.	『3.1 起動、終了について』『Q10』

4.2　コンソール画面出力メッセージ

メッセージ	参照場所
Can not access SharedMemory(incorrect ipckey).	『3.2 padmin コマンドについて』『Q2』
Operation timeout.	『3.2 padmin コマンドについて』『Q4』

CLUSTERPRO
MC ProcessSaver 2.0 for Linux
FAQ 集

2015 年 3 月 第 1 版

日本電気株式会社
東京都港区芝五丁目7番地1号
TEL (03) 3454-1111（代表）

P

保護用紙