

LifeKeeper for Linux

Dovecot Generic ARK スクリプト仕様書

第7版



目次

1. 概要	3
2. サポート環境	3
2.1. Dovecot Generic ARK のサポートポリシー	3
3. 前提条件	5
4. 基本動作	6
4.1. 起動処理(restore)	6
4.2. 停止処理(remove)	6
4.3. 監視処理(quickCheck)	6
4.4. 回復処理(recover)	6
5. 障害時の切り替え動作	7
5.1. Dovecot リソース障害時	7
5.2. ノード障害時	7
6. スクリプト仕様	8
6.1. 監視スクリプト(quickCheck)	8
6.2. 起動スクリプト(restore)	8
6.3. 停止スクリプト(remove)	9
6.4. 回復スクリプト(recover)	10
7. スクリプトの実行時間	10
7.1. restore、remove、recover	10
7.2. quickCheck	10
8. スクリプトのパラメーター一覧	12
9. 免責事項	13

改版履歴

2013年 6月 5日	第1版
2013年 11月 18日	第2版
2014年 1月 27日	第3版
2014年 3月 6日	第4版
2015年 4月 1日	第5版
2017年 5月 31日	第6版
2017年 6月 12日	第7版

1. 概要

本書は Dovecot サービス向けの Generic ARK スクリプトの動作を解説します。あらゆる環境において正常に動作することを保証するものではありません。本書に記載されている動作環境とは異なる環境で使用する場合は、必要に応じてスクリプト内容を修正し、ご利用ください。以降、本文書内でスクリプトと記載している箇所は、Dovecot サービス向けの Generic ARK スクリプトを指すものとします。

2. サポート環境

2.1. Dovecot Generic ARK のサポートポリシー

Dovecot Generic ARK のサポートポリシーは以下の通りです。

[サポート対象 OS]

LifeKeeper のサポートに準じます。

[サポート対象 Dovecot]

Dovecot v1

Dovecot v2

[留意点]

- 本 Generic ARK は、OS が提供している Dovecot の RPM パッケージを使用して設定した Dovecot 環境で動作検証を行っています。Dovecot のソースファイルからコンパイルした環境は動作確認していません。
- 動作検証が行われていない方法でインストールされた Dovecot 環境において、Dovecot に関連した問題が発生した場合は、OS が提供している RPM からインストールした Dovecot に変更することをお願いする可能性があります。予めご了承ください。
- Amazon EC2 環境で本 Generic ARK をご利用いただくことができます。なお、Elastic IP を利用している場合は、保護対象のアプリケーションをすべての IP に対して Listen する設定が必要です。
- LifeKeeper Single Server Protection for Linux で、本 Generic ARK をご利用いただくことができます。

[参考情報]

Dovecot Generic ARK の各スクリプトは以下の環境で動作検証を行っております

- RHEL5.9(kernel-2.6.18-348.el5) + Dovecot(dovecot-1.0.7-7.el5_7.1)
+ LifeKeeper for Linux v8.1.2
- RHEL5.9(kernel-2.6.18-348.el5) + Dovecot(dovecot-1.0.7-7.el5_7.1)
+ LifeKeeper for Linux v8.2
- RHEL6.3(kernel-2.6.32-279.el6) + Dovecot(dovecot-2.0.9-2.el6_1.1)
+ LifeKeeper for Linux v8.1.2
- RHEL6.4(kernel-2.6.32-358.el6.x86_64) + Dovecot(dovecot-2.0.9 - 5.el6)
+ LifeKeeper for Linux v8.2
- RHEL6.5(kernel-2.6.32-431.el6.x86_64) + Dovecot(dovecot-2.0.9 - 7.el6)
+ LifeKeeper for Linux v8.2.1
- RHEL7.0(kernel- 3.10.0-123.el7.x86_64) + Dovecot(dovecot-2.2.10-4.el7)
+ LifeKeeper for Linux v8.4
- RHEL7.3(kernel-3.10.0-514.el7.x86_64) + Dovecot(dovecot-2.2.10-7.el7)
+ LifeKeeper for Linux v9.1.2
- SLES12SP2(kernel- 4.4.21-69-default) + Dovecot(dovecot22-2.2.13-2.7)
+ LifeKeeper for Linux v9.1.2

3. 前提条件

Dovecot リソースを作成する前に以下の前提条件を満たす必要があります。

- 各ノードに同一バージョンの LifeKeeper for Linux(ライセンスを含む)がインストールされていること
- 各ノードに同一バージョンの Dovecot がインストールされていること
- 各ノード上の Dovecot の設定が同一であること
- 各ノードにおいて、OS 起動時の Dovecot の自動起動が無効にされていること
- Dovecot リソース導入前に、各ノードで Dovecot の起動、停止が手動で行えるように Dovecot の設定が行なわれていること
- クラスタシステムに Dovecot で使用する IP アドレスが確保されていること(初期値ではループバックアドレス 127.0.0.1 を使用します。IP リソースによる切り替え可能な仮想 IP アドレスを指定する場合は、依存関係を作成し、IP リソースを Dovecot リソースの下位に配置する必要があります。)
- Dovecot の動作に必要なポートが開放されていること(初期値では TCP の 143 ポートで接続を待ち受けます。)

4. 基本動作

Dovecot リソースでは、下記の 4 つの動作を実行します。

4.1. 起動処理(restore)

Dovecot リソースを ISP(サービス稼働状態)にするために、Dovecot サービスの起動処理を実行します。この動作は Dovecot リソースの起動処理、スイッチオーバーやフェールオーバーの切り替え動作が該当します。起動処理に失敗した場合は、Dovecot リソースは OSF(サービス障害状態)になります。Dovecot リソースの上位に依存関係を持つリソースが存在する場合は、そのリソースは起動されません。この動作の詳細は「6.スクリプト仕様」をご参照ください。

4.2. 停止処理(remove)

Dovecot リソースを OSU(サービス停止状態)にするために、Dovecot サービスの停止処理を実行します。この動作は Dovecot リソースの停止処理、スイッチオーバーやフェールオーバーの切り替え動作が該当します。停止処理に失敗した場合は、Dovecot リソースは OSF(サービス障害状態)になります。Dovecot リソースの下位に依存関係を持つリソースが存在する場合は、そのリソースは停止されません。

LifeKeeper for Linux v7.1 以降はフェールオーバー時、リソースの停止に失敗すると、稼働系ノードの強制再起動し、フェールオーバーを継続させます。この動作の詳細は「6.スクリプト仕様」をご参照ください。

4.3. 監視処理(quickCheck)

Dovecot リソースが ISP のときに、Dovecot の稼働状態の判定を行いません。監視処理で障害を検出すると回復処理に状態遷移します。この動作の詳細は「6.のスクリプト仕様」をご参照ください。

4.4. 回復処理(recover)

監視処理で障害を検出した場合に限り実行します。監視処理で障害を検出したノード上で Dovecot サービスの再起動を実行します。回復処理に失敗した場合、待機系へのフェールオーバーを開始します。この動作の詳細は「6.スクリプト仕様」をご参照ください。フェールオーバー動作の詳細は「5.障害時の切り替え動作」をご参照ください。

5. 障害時の切り替え動作

スクリプトは以下のようなケースで切り替えが発生することを目的に作成しております。Dovecot リソースは起動(restore)スクリプト、停止(remove)スクリプト、監視(quickCheck)スクリプト、回復(recover)スクリプトがそれぞれ指定されているものとします。

5.1. Dovecot リソース障害時

Dovecot リソースの障害を検知した場合は、以下のように動作します。

- (1). Dovecot リソースの quickCheck 処理で障害を検知します。
- (2). Dovecot リソースの recover 処理を実行され、回復処理を実行します。
 - Dovecot リソースの回復処理に成功した場合は、通常のサービス状態に戻ります。
 - Dovecot リソースの回復処理に失敗した場合は、待機系ノードへフェールオーバーします。

5.2. ノード障害時

コミュニケーションパスが全て DEAD 状態になった場合は、待機系ノードでフェールオーバーを開始します。このフェールオーバーでは待機系ノードで各リソースを順次、起動処理を行いません。この時の Dovecot リソース起動時の動作としては、「3.起動処理(restore)」を実行します。

6. スクリプト仕様

各スクリプトの仕様を以下に記載いたします。

6.1. 監視スクリプト(quickCheck)

以下の手順で Dovecot を監視します。

(1).Dovecot のプロセスの確認

Dovecot のプロセス（プロセス名：“dovecot”）の存在を確認します。

(2).ソケット通信の確認

Dovecot で設定されている IP アドレス/ポート¹に対して接続の確認をします。

(3).Dovecot の稼働状況の確認

Dovecot のステータスコマンドを実行して、Dovecot の稼働状態を確認します。

```
# service dovecot status
```

上記の監視項目が全て成功すれば Dovecot が稼働状態であると判定し、上記の監視項目が 1 つでも失敗した場合は、Dovecot が停止状態であると判定します。

6.2. 起動スクリプト(restore)

以下の手順で Dovecot の起動を実行します。

(1).Dovecot の稼働状態の確認

Dovecot の稼働状態²を確認します。Dovecot が稼働状態であると判定した場合、起動スクリプトを終了します。

(2).Dovecot の起動

Dovecot の起動コマンドを実行します。

```
# service dovecot start
```

(3).Dovecot の稼働状態の確認

Dovecot の起動コマンドを実行後、sleep コマンドによる 3 秒³間隔をあけてから Dovecot の稼働状態を確認します。Dovecot が稼働状態であると判定した場合、起動スクリプトを終了し、Dovecot が停止状態である場合は、(2)~(3)の処理を最大 3 回⁴まで繰り返し実行します。Dovecot の繰り返し起動処理が

¹ IP アドレス/ポート番号は、APP_DOVECOT_IP_LIST プロパティで変更可能です。

² 稼働状態の確認項目は、「6.1 監視スクリプト(quickCheck)」の 3 つの確認項目を実行します。

³ sleep 秒数は APP_DOVECOT_RESTORE_WAIT パラメータで指定可能です。

⁴ 起動処理の繰り返し回数は APP_DOVECOT_RESTORE_CHECK_RETRY パラメータで指定可能です。

実行された場合は、下記のログが記載されます。

<日時> Dovecot process <found/not found>,Port <connected/cannot connected>, Status <alive/dead>, Retry count: <回数>.

6.3. 停止スクリプト(remove)

以下の手順で Dovecot の停止を実行します。

(1).Dovecot の稼働状態の確認

Dovecot の稼働状態⁵を確認します。Dovecot が停止状態であると判定した場合、停止スクリプトを終了します。

(2).Dovecot の停止

Dovecot の停止コマンドを実行します。
#service dovecot stop

(3).Dovecot の稼働状態を確認

Dovecot の停止コマンドを実行後、sleep による 3 秒⁶の間隔をあけてから Dovecot の稼働状態を確認します。Dovecot が停止状態であると判定した場合、停止スクリプトを終了し、Dovecot が稼働状態である判定した場合は(2)~(3)の処理を最大 3 回⁷まで繰り返し実行します。

(4).(2)~(3)で Dovecot の停止コマンドを実行後でも Dovecot が稼働状態である場合は、killall コマンドを使用して強制的に Dovecot のプロセスを停止させます。Dovecot プロセスの強制停止を実行後、sleep による 3 秒の間隔をあけてから Dovecot の稼働状態を確認します。Dovecot が停止状態であると判定した場合、停止スクリプトを終了します。

⁵ 稼働状態の確認項目は、「6.1 監視スクリプト(quickCheck) 」の「(1) Dovecot のプロセスの確認」を実行します。

⁶ sleep 秒数は、APP_DOVECOT_REMOVE_CHECK_RETRY パラメータで変更可能です。

⁷ 停止処理の繰り返し回数は APP_DOVECOT_REMOVE_CHECK_RETRY パラメータで指定可能です。

6.4. 回復スクリプト(recover)

Dovecot の停止処理および起動処理を実行し、Dovecot のサービスの回復を実行します。回復スクリプトは、内部的に停止スクリプトと起動スクリプトの処理を呼び出すことで、Dovecot の再起動処理を実行します。

- (1).停止スクリプトの処理を呼び出します。詳細は 6.2 起動スクリプト(restore)を参照してください。
- (2).起動スクリプトの処理を呼び出します。詳細は 6.3 停止スクリプト(remove)を参照してください。
- (3).(2)の起動スクリプトに成功した場合は回復スクリプトを終了し、失敗した場合は待機ノードへフェールオーバーが開始されます。

7. スクリプトの実行時間

スクリプトの実行時間およびタイムアウトは下記の通りです。

7.1.restore、remove、recover

これらのスクリプトの処理に要する時間に制限はありません。スクリプトの終了時間はスクリプト内部で使用されている OS のコマンドや Dovecot の各コマンドの実行処理時間に依存します。導入するシステムの CPU、Memory、I/O 等のシステムリソースの使用状況により終了時間に遅延が発生する場合があります。

7.2.quickCheck

LifeKeeper for Linux v9.0.2 までは、quickCheck スクリプトは初期値で 22 秒のタイムアウト値が設定されています。quickCheck スクリプトのタイムアウト値は/etc/default/LifeKeeper ファイルに\${CONVTAG}_TIMEOUT パラメータを以下のように追加することで調整が可能です。\$CONVTAG は Dovecot リソースのタグ名で、英数字以外の全ての記号を"_"に置き換えた文字列です。

```
${CONVTAG}_TIMEOUT=<秒数>
```

LifeKeeper for Linux v9.1.0 以降では、初期値ではタイムアウトが設定されません。タイムアウトを設定する場合は、/etc/default/LifeKeeper ファイルに\${CONVTAG}_QUICKCHECK_TIMEOUT パラメータを以下のように追加することで調整が可能です。\$CONVTAG は Dovecot リソースのタグ名で、英数字以外の全ての記号を"_"に置き換えた文字列です。

```
${CONVTAG}_QUICKCHECK_TIMEOUT=<秒数>
```

quickCheck スクリプトの処理がタイムアウトした場合は、監視処理のタイムアウトを意味する下記のメッセージがログに出力され、処理が終了します。quickCheck スクリプトのタイムアウトは障害とは見なさないため、タイムアウト後にローカルリカバリやフェールオーバーの動作には遷移しません。

Script has hung checking "\$TAG". Forcibly terminating.

\$TAG は Dovecot リソースのタグ名です。

8. スクリプトのパラメーター一覧

設定可能なパラメーター一覧を下記に記載します。これらのパラメータは設定ファイル /etc/default/LifeKeeper に記載することで調整可能です。パラメータの変更は両ノードで行ってください。パラメータ変更後、直ちに設定が有効になり、LifeKeeper の再起動は必要ありません。

パラメータ	機能
APP_DOVECOT_DEBUG	デバックログの有効(1)・無効(0)の設定をします。初期値は無効(0)です。
APP_DOVECOT_RESTORE_CHECK_RETRY	起動コマンドのリトライ回数を指定します。初期値は3回です。
APP_DOVECOT_RESTORE_WAIT	起動コマンドのsleep時間(秒)を指定します。初期値は3秒です。
APP_DOVECOT_REMOVE_CHECK_RETRY	停止コマンドのリトライ回数を指定します。初期値は3回です。
APP_DOVECOT_REMOVE_WAIT	停止コマンドのsleep時間(秒)を指定します。初期値は3秒です。
APP_DOVECOT_IP_LIST	使用するIPアドレス:ポート番号:TCPタイムアウト値(秒)を指定します。初期値は127.0.0.1:143:2です。

9. 免責事項

- 本書に記載された情報は予告なしに変更、削除される場合があります。最新のものをご確認ください。
- 本書に記載された情報は、全て慎重に作成され、記載されていますが、本書をもって、その妥当性や正確性についていかなる種類の保証もするものではありません。
- 本書に含まれた誤りに起因して、本書の利用者に生じた損害については、サイオステクノロジー株式会社は一切の責任を負うものではありません。
- 第三者による本書の記載事項の変更、削除、ホームページ及び本書等に対する不正なアクセス、その他第三者の行為により本書の利用者に応じた一切の損害について、サイオステクノロジー株式会社は一切の責任を負うものではありません。
- システム障害などの原因によりメールフォームからのお問い合わせが届かず、または延着する場合がありますので、あらかじめご了承ください。お問い合わせの不着及び延着に関し、サイオステクノロジー株式会社は一切の責任を負うものではありません。

【著作権】

本書に記載されているコンテンツ(情報・資料・画像等種類を問わず)に関する知的財産権は、サイオステクノロジー株式会社に帰属します。その全部、一部を問わず、サイオステクノロジー株式会社の許可なく本書を複製、転用、公衆への送信、販売、翻案その他の二次利用をすることはいずれも禁止されます。またコンテンツの改変、削除についても一切認められません。

本書では、製品名、ロゴなど、他社が保有する商標もしくは登録商標を使用しています。

サイオステクノロジー株式会社

〒106-0047 東京都港区南麻布 2-12-3 サイオスビル
<http://www.sios.com>