

BigFix Patch for CentOS ユーザーズ・ガイド



Special notice

Before using this information and the product it supports, read the information in [Notices \(on page lxxii\)](#).

Edition notice

This edition applies to BigFix version 11 and to all subsequent releases and modifications until otherwise indicated in new editions.

目次

Special notice.....	ii
Edition notice.....	iii
第 1 章. 概要.....	6
本更新リリースの新機能.....	6
サポートされるプラットフォームおよび更新.....	10
サポートされる CentOS リポジトリ.....	12
サイトのサブスクリプション.....	13
パッチ適用方式.....	13
第 2 章. ダウンロード・プラグインの使用.....	16
「ダウンロード・プラグインの管理」ダッシュボードの概要.....	17
CentOS ダウンロード・プラグイン R2 の登録.....	18
CentOS ダウンロード・プラグイン R2 の基本設定の構成.....	19
CentOS ダウンロード・プラグイン R2 の詳細設定の構成.....	21
CentOS ダウンロード・プラグイン R2 の登録解除.....	23
CentOS ダウンロード・プラグイン R2 のアップグレード.....	24
CentOS ダウンロード・プラグイン R2 の拡張.....	24
第 3 章. ダウンロード・キャッシャーの使用.....	29
CentOS ダウンロード・キャッシャー R2 の使用情報.....	29
隔離された環境での CentOS ダウンロード・キャッシャー R2 の使用.....	35
sha1 フォルダーでのパッケージのキャッシング.....	35
ローカル・キャッシュ・フォルダーでのパッケージのキャッシング.....	37
第 4 章. BigFix Patch for CentOS の使用 Linux™.....	38
Fixlet を使用したパッチの適用.....	38
RPM 適用ウィザードの使用.....	40
置き換え.....	41
第 5 章. 複数パッケージのベースラインのインストール.....	42
ベースラインでの複数パッケージのインストール.....	44
第 6 章. カスタム・リポジトリの管理.....	47
CentOS カスタム・リポジトリ管理ダッシュボード.....	47
リポジトリの追加.....	48
リポジトリへの CentOS エンドポイントの登録.....	48

リポジトリからの CentOS エンドポイントの登録解除.....	49
リポジトリの削除.....	50
リポジトリのインポート.....	50
YUM のパッケージ更新の確認.....	50
カスタム・リポジトリからのパッケージのインストール.....	51
第 7 章. YUM トランザクション管理.....	53
YUM トランザクションのロールバック.....	55
YUM トランザクションの取り消し.....	55
YUM トランザクションのやり直し.....	56
付録 A. サポート.....	58
付録 B. トラブルシューティング.....	59
付録 C. よくある質問.....	63
Notices.....	lxxii

第1章. 概要

BigFix Patch for CentOS Linux™ を使用すると、Linux™ クライアントを常に最新の更新およびサービス・パックが適用された状態にすることができます。

BigFix は、新規のパッチまたは更新が入手できるようになるたびに Fixlet® をリリースします。Fixlet により、企業内でそのパッチまたは更新を必要とするすべてのコンピューターが識別され、修正が適用されます。BigFix コンソールのオペレーターは、キーを数回押すだけで、関係するすべてのコンピューターにパッチを適用し、ネットワーク全体の適用の進行状況を視覚化することができます。BigFix エージェントは、オペレーティング・システムのバージョン、ファイルのバージョン、システムの言語、および他の関連する要因をチェックして、パッチが必要かどうかおよびいつ必要となるかを判別します。

BigFix は、リリース前に各 Fixlet® を研究所でテストし、多くの場合に問題を検出します。問題には、Fixlet® に追加のメモを添付するという方法で対処します。通常、そのメモにより、コンソール・オペレーターは問題を回避することができ、パッチの適用プロセスに付加価値を与えるものとなっています。BigFix は、ユーザー・フィードバックもメモに反映して最新の情報を提供します。

Fixlet により、大量の更新およびパッチを比較的簡単に管理することができ、どのようなスケジュールの下でも、自動化され、対象が絞り込まれた適用作業を実行できます。ネットワーク帯域幅を最適化するために大容量のダウンロードを段階的に実行でき、インベントリーまたは監査の制御のために、適用プロセス全体をモニター、グラフ化、および記録できます。

図 1. 「パッチ管理」ナビゲーション・ツリー

本更新リリースの新機能

本リリースの BigFix Patch for CentOS には、パッチ適用のさまざまな機能拡張が含まれています。

CentOS 8 コンテンツのサポート

CentOS Linux™ Enterprise 8 (x86_64) コンテンツが、「CentOS 8 向けパッチ」という新規サイトに公開されています。

このリリースでは、パッケージ・マネージャーのネイティブのコマンド行インターフェースである DNF がサポートされます。DNF を使用すると、CentOS 8 システムにパッチを適用できます。DNF は YUM 上のラッパーであり、これにより、依存関係の問題が削減され、パフォーマンスが向上し、セキュリティ・パッチをより確実にインストールできます。

CentOS ダウンロード・プラグイン R2 機能拡張

CentOS ダウンロード・プラグイン R2 v1.0.0.2 以降は、CentOS R2 ダウンロード・キャッシュの `download_dir` (`plugin.ini` ファイルでは `localCache` と呼ばれる) によってキャッシュされたパッケージを使用できると同時に、インターネットからパッケージを取得することもできます。

これまで CentOS ダウンロード・プラグイン R2 は、BigFix サーバーにおいて、次のいずれかのシナリオで使用されてきました。

- BigFix サーバーは完全に隔離されており、CentOS ダウンロード・キャッシュ R2 を使用する。あるいは、
- BigFix サーバーはインターネット対応であり、CentOS ダウンロード・キャッシュ R2 を使用しない。

この機能拡張により、パッケージをオフラインでキャッシングして、ダウンロード時間を節約することができます。

詳しくは、「[ダウンロード・キャッシュの設定 \(ページ 22\)](#)」を参照してください。

CentOS ダウンロード・キャッシュ R2 機能拡張

CentOS ダウンロード・キャッシュ R2 v1.0.0.2 以降には、以下の機能拡張が組み込まれています。

パッケージ sha1 ダウンロード・サポート

CentOS ダウンロード・キャッシュ R2 は、`--sha1_download_dir` を使用することで、RPM 形式ではなく sha1 ファイルとしてパッケージをダウンロードできるようになりました。

これまでは、ダウンロード・キャッシュで「`buildRepo -key centos-7-x64`」を使用すると、CentOS リポジトリ「`centos-7-x64`」の構造はオフラインでミラーリングされていました。この方法だと、パッケージが複数のリポジトリにある場合、パッケージが重複してしまう可能性があります。

`--sha1_download_dir` を使用することにより、すべてのパッケージが sha1 のファイル名を持つファイルとして、すべてのリポジトリ (キー) から単一のフラット・ディレクトリにダウンロードされます。

リポジトリ・アクセス検査

BigFix でサポートされる CentOS 基本リポジトリおよびサブリポジトリにアクセスできるかどうかを検証するための新規コマンドが導入されました。`check-baserepos` および `check-allrepos`

ストレージ・スペース所要量のチェック

`builRepo` コマンドを使用するときにストレージ・スペース所要量を計算して調べるための新規コマンド「`check-storagereq`」が導入されました。

このコマンドは、`--sha1_download_dir` オプションを使用する場合と使用しない場合の、リポジトリ・メタデータおよびパッケージのダウンロードに必要なスペースを出力します。

スペース節約のベンチマーク

`--sha1_download_dir` オプションを使用することで、スペース節約のベンチマークが設定されています。

`--sha1_download_dir` オプションを使用することで、同じ CentOS バージョンの複数のリポジトリをキャッシュする際のストレージ・サイズ、ダウンロード・サイズ、および時間が大幅に削減されます。これは、同じ CentOS バージョン (たとえば、`centos-6.8-x64`、`centos-6.7-x64`、`centos-6.6-x64`) のリポジトリ間で、多くのパッケージが重複しているためです。CentOS バージョン (たとえば、`centos-6.8-x64`、`centos-7.1-x64`) ごとにリポジトリを 1 つだけキャッシュする場合、スペースは節約されません。

詳しくは、以下のトピックを参照してください。

- [CentOS ダウンロード・キャッシャー R2 の使用情報 \(\(ページ\) 29\)](#)
- [隔離された環境での CentOS ダウンロード・キャッシャー R2 の使用 \(\(ページ\) 35\)](#)
- [sha1 フォルダーでのパッケージのキャッシング \(\(ページ\) 35\)](#)
- [ローカル・キャッシュ・フォルダーでのパッケージのキャッシング \(\(ページ\) 37\)](#)

表 1. 以前の更新

機能拡張	説明	リソース
複数パッケージのベースラインのインストール	BigFix Patch には、ベースラインに含まれる複数パッケージの更新のインストールを単一のタスクに結合できるソリューションが用意されています。これにより、ベースラインの実行時間を効果的に短縮できます。 このソリューションは、「CentOS6 プラグイン R2 向けパッチ (Patches for CentOS6 Plugin R2)」サイトおよび「CentOS7 プラグイン R2 向けパッチ (Patches for CentOS7 Plugin R2)」サイトの Fixlet に対して使用できます。	複数パッケージのベースラインのインストール ((ページ) 42)
ロギングとエラー処理の強化	エラー・ログとデバッグ出力が設計し直され、エラー報告がより分かりやすくなりました。これにより、問題のトラブルシューティングやデバッグにかかる時間が短縮されます。 この機能拡張は、「CentOS6 プラグイン R2 向けパッチ (Patches for CentOS6 Plugin R2)」サイトおよび「CentOS7 プラグイン R2 向けパッチ (Patches for CentOS7 Plugin R2)」サイトの Fixlet に対して使用できます。	ロギング・レベルの設定 ((ページ) 22)
カスタム・リポジトリのサポート	カスタム・リポジトリのサポートにより、YUM を使用してパッチをローカル・リポジトリからダウンロードし、CentOS エンドポイントに配布できます。また、カスタム・リポジトリを使用して、BigFix からカスタム・ソフトウェアを配信できます。	カスタム・リポジトリの管理 ((ページ) 47)
「YUM トランザクション履歴」ダッシュボード	BigFix では、単一のダッシュボードで YUM 関連のすべてのトランザクションを表示し、エンドポイントに対して実行された操作およびコマンドをモニターできます。また、このダッシュボードは、「ロールバック」、「元に戻す」および「やり直し」などのトランザクション管理機能を備えています。	YUM トランザクション管理 ((ページ) 53)

表 1. 以前の更新 (続く)

機能拡張	説明	リソース
CentOS 7 コンテンツのサポート	CentOS Linux™ Enterprise 7 (x86_64) コンテンツが、「CentOS 7 向けパッチ」と呼ばれる新規サイトに公開されています。 このリリースでは、パッケージ・マネージャーのネイティブ・コマンドライン・インターフェースであり、CentOS 7 システムにパッチを適用することができる YUM がサポートされています。YUM により、依存関係の問題が削減され、パフォーマンスが向上し、セキュリティ・パッチをより確実にインストールできます。 「CentOS 7 向けパッチ」サイトのパッチを使用するには、エンドポイントに bzip2 ユーティリティをインストールする必要があります。bzip2 ユーティリティは、すべてのパッケージ情報を含むメタデータ・ファイルをエンドポイントに解凍します。  注: bzip2 ユーティリティは CentOS 7 の基本インストールには含まれていない可能性があるため、この圧縮ユーティリティを手動でインストールする必要があります。 Install bzip2 (ID #1) と呼ばれる Fixlet を使用して、bzip2 ユーティリティのインストールが必要なエンドポイントを識別します。ベンダーの Web サイトからインストール・パッケージをダウンロードし、RPM 適用ウィザード (「Linux RPM パッチ」サイトから入手可能) を使用して、さまざまなエンドポイントにパッケージをデプロイします。	サポートされるプラットフォームおよび更新 ((ページ) 10) パッチ適用方式 ((ページ) 13) RPM 適用ウィザードの使用 ((ページ) 40)
CentOS 8 コンテンツのサポート	CentOS Linux™ Enterprise 8 (x86_64) コンテンツが、「CentOS 8 向けパッチ」という新規サイトに公開されています。 このリリースでは、パッケージ・マネージャーのネイティブのコマンド行インターフェースである DNF がサポートされます。DNF を使用すると、CentOS 8 システムにパッチを適用できます。DNF は YUM 上のラッパーであり、これにより、依存関係の問題が削減され、パフォーマンスが向	CentOS 8 向けパッチのサイトは、BaseOS と AppStream リポジトリのみをサポートしています。詳しくは、「サポートされている CentOS リポジトリ」を参照してください。

表 1. 以前の更新 (続く)

機能拡張

説明

リソース

上し、セキュリティー・パッチをより確実にインストールできます。

アナウンス

CentOS 8 will not make announcements on its official site as in earlier versions. BigFix Patch team retrieves the package information from <https://feeds.centos.org/> for generating the content based on the changes mentioned in [BaseOS](#) and [AppStream](#) change list.

Red Hat API's are used to get all the dependent packages and errata information.

置き換え

CentOS 8 repositories metadata now contains information related to latest available packages, hence all the superseded fixlets are expected to fail with an error `No matching Package found`.

Once the fixlet is superseded, the fixlet is made non-relevant by default. The value(client) of `_BESClient_CentOS_EnableSupersededEval` is removed for superseded fixlets and relevance for superseded fixlets will be set to false `<Relevance>False</Relevance>`. Please deploy only the latest available patches.

In order to deploy superseded fixlet using custom site, the relevant repository metadata and packages must be precedently cached.



注: ベースラインの Fixlet を置き換えるとエラーが発生してインストールが失敗するため、インストールを続行する前にすべてのベースラインを同期する必要があります。最新の使用可能なパッケージがベースラインとともにデプロイされます。

サポートされるプラットフォームおよび更新

BigFix Patch for CentOS Linux™ では、CentOS Linux™ 6 および 7 の各プラットフォームで CentOS のセキュリティー・アドバイザリー、修正のアドバイザリー、および改善に関するアドバイザリー用の Fixlet が提供されます。

表 2. BigFix Patch for CentOS でサポートされるプラットフォームおよびパッチ

以下の表に、サポートされるプラットフォームと、パッチ適用の Fixlet が含まれる対応サイトをリストします。

サポートされるプラットフォーム	Fixlet サイト名	更新のタイプ
CentOS Linux 6 (x86、x86_64)	CentOS6 プラグイン R2 向けパッチ (Patches for CentOS6 Plugin R2)	サポートされるパッケージを含むリポジトリのリストを確認するには、 サポートされる CentOS リポジトリ (ページ 12) を参照。
CentOS Linux 7 (x86_64)	CentOS7 プラグイン R2 向けパッチ (Patches for CentOS7 Plugin R2)	サポートされるパッケージを含むリポジトリのリストを確認するには、 サポートされる CentOS リポジトリ (ページ 12) を参照。
CentOS Linux 8 (x86_64)	CentOS 8 向けパッチ	サポートされるパッケージを含むリポジトリのリストを確認するには、 サポートされる CentOS リポジトリ (ページ 12) を参照。



注:

- **CentOS 8** は 2021 年 12 月 31 日に終了しました。つまり、**CentOS** は **CentOS Linux 8** のアップデートを今後提供しません。これにより、BigFix はそのパッチ・コンテンツを提供しなくなります。近日中にこのサイトの廃止を予定しています。詳しくは、「<https://www.centos.org/centos-linux-eol/>」を参照してください。
- **CentOS 8 Stream** は BigFix Patch ではサポートされていません。
- 置き換えられる Fixlet が **CentOS 8 向けパッチ**のサイトの一部である場合、CentOS 8 のメタデータに古いパッケージ情報がなく、最新バージョンにのみ更新されるため、Fixlet のインストールに失敗します。
- **CentOS 8** が告知情報の公開を停止したため、すべての Fixlet が CentOS によって公開されたフィードに基づいて作成されます。
- 置き換えられて 1 年以上になる Fixlet コンテンツは、外部サイトから削除され、別のサイトにアーカイブされます。置き換えられたコンテンツにアクセスする必要がある場合は、HCL の技術営業担当員にお問い合わせください。
- 「**CentOS 6 ネイティブ・ツール向けパッチ**」サイトおよび「**CentOS 7 向けパッチ**」サイトは、2017 年 9 月 29 日で非推奨になっています。BigFix Patch は、非推奨になった日付以降は、これらのサイトに対して Fixlet のコンテンツもサポートも提供しなくなります。



- 「CentOS 5 ネイティブ・ツール向けパッチ」サイトは、2017 年 7 月 31 日で非推奨になっています。BigFix Patch は、非推奨になった日付以降は、CentOS Linux 5 に対してコンテンツもサポートも提供しなくなります。
- 「CentOS 5 向けパッチ」サイトおよび「CentOS 6 向けパッチ」サイトは、2016 年 12 月 31 日で非推奨になっています。BigFix Patch では、これらのサイトに関するコンテンツおよびサポートを提供していません。

BigFix では、以下の発表メーリング・リストで公開される CentOS 勧告に基づいて、Fixlet コンテンツが生成されます。

CentOS-announce

セキュリティ情報および一般情報 (正誤表パッケージなど) が含まれる CentOS の発表。このメーリング・リストをサブスクライブするには、<https://lists.centos.org/hyperkitty/list/announce@lists.centos.org/> にアクセスします。

CentOS-CR-announce

さまざまなバージョンの CR リポジトリに関する CentOS の発表。このメーリング・リストをサブスクライブするには、<https://lists.centos.org/hyperkitty/list/announce@lists.centos.org/> にアクセスします。



重要: BigFix では、正式な CentOS 勧告なしにリリースされたパッケージのコンテンツは生成されません。

サポートされる CentOS リポジトリ

BigFix Patch for CentOS は、いくつかの CentOS リポジトリのパッケージをサポートしています。

次の表に、各 CentOS バージョンでサポートされるパッケージを含むリポジトリのリストを示します。

表 3. サポートされる CentOS リポジトリ

CentOS バージョン	リポジトリ
CentOS 6 (x86)	Base: http://mirror.centos.org/centos/6/os/i386/ 更新: http://mirror.centos.org/centos/6/updates/i386/ Extras: http://mirror.centos.org/centos/6/extras/i386/ ファーストトラック: http://mirror.centos.org/centos/6/fasttrack/i386/
CentOS 6 (x86_64)	Base: http://mirror.centos.org/centos/6/os/x86_64/ 更新: http://mirror.centos.org/centos/6/updates/x86_64/ Extras: http://mirror.centos.org/centos/6/extras/x86_64/

表 3. サポートされる CentOS リポジトリ (続く)

CentOS バージョン	リポジトリ
	ファーストトラック: http://mirror.centos.org/centos/6/fasttrack/x86_64/
CentOS 7 (x86_64)	Base: http://mirror.centos.org/centos/7/os/x86_64/ 更新: http://mirror.centos.org/centos/7/updates/x86_64/ Extras: http://mirror.centos.org/centos/7/extras/x86_64/ ファーストトラック: http://mirror.centos.org/centos/7/fasttrack/x86_64/
CentOS 8 (x86_64)	ベース: http://mirror.centos.org/centos/8/BaseOS/x86_64/os/ AppStream: http://mirror.centos.org/centos/8/AppStream/x86_64/os/



注: パッケージが、関連する CentOS 勧告なしでリポジトリで公開される場合があります。そのような場合、BigFix により Fixlet は提供されません。

サイトのサブスクリプション

サイトとは、ユーザー、HCL、またはベンダーにより内部的に作成される Fixlet メッセージの集合です。

実装環境内のシステムにパッチを適用するために、サイトにサブスクライブして Fixlet メッセージにアクセスします。

サイトのサブスクリプションを追加するには、ベンダーまたは HCL からマストヘッド・ファイルを入手するか、またはライセンス・ダッシュボードを使用します。Fixlet サイトへのサブスクライブ方法について詳しくは、『*BigFix インストール・ガイド*』を参照してください。

サイトについて詳しくは、『*BigFix コンソール・オペレーター・ガイド*』を参照してください。

パッチ適用方式

BigFix では、ネイティブ・ツールを使用することで、パッチ管理ソリューションの柔軟性を向上します。

すべての CentOS コンテンツの Fixlet では、CentOS のデフォルトのパッチ・マネージャーである Yellow dog Updater Modified (YUM) が使用されます。YUM は、Red Hat Package Manager (RPM) パッケージを更新、インストール、および削除するパッケージ管理ツールです。YUM ではコマンド・ライン・インターフェースを使用して、パッケージのインストール、アンインストール、および更新の処理を単純化します。これらの処理では YUM リポジトリへのアクセス権限が必要です。

以前は、BigFix Patch for CentOS サイトでは Endpoint Dependency Resolver (EDR) と呼ばれるユーティリティのセットを使用して、エンドポイント上のパッケージ依存関係を処理していました。YUM はこれらの EDR ユーティリ

ティアーに置き換わり、より柔軟なパッチの適用を可能にし、Red Hat ソリューションおよび CentOS ソリューションに応じた結果を提供します。以下のサイトが使用可能です。

YUM ユーティリティーの設定

YUM ユーティリティーを適用する「BigFix Patch for CentOS」サイトでは、`/etc/yum.conf` 内の Fixlet 設定を使用します。ただし、以下の YUM 設定は除きます。

- cachedir
- keepcache
- プラグイン
- reposdir
- pluginpath
- pluginconfpath
- metadata_expire
- installonlypkgs

ネイティブ・ツールのコンテンツでのファイル関連度の識別

このネイティブ・ツールは、EDR と同じ方法でファイル関連度をキャプチャーします。どちらの方法でも、関連句 `exist lower version of a package, but not exist higher version of it` がチェックされます。同じ適用環境に両方のツールが適用される場合、関連度結果は同じです。

パッチ適用方式一覧

以下の表に、CentOS エンドポイントの管理に使用可能な各パッチ適用方法に対して適用できるサイトと機能をリストします。

パッチ適用方式	適用可能サイト	適用可能機能
エンドポイントの依存解決 (EDR)	• Linux RPM パッチ • CentOS 5 向けパッチ* • CentOS 6 向けパッチ*	• ダウンロード・プラグイン • RPM デプロイメント • 優先リスト
ネイティブ・ツール (YUM)	• パッチ・サポート • CentOS 5 ネイティブ・ツール向けパッチ • CentOS 6 ネイティブ・ツール向けパッチ • CentOS7 プラグイン R2 向けパッチ (Patches for CentOS7 Plugin R2)	ダウンロード・プラグイン

パッチ適用方式	適用可能サイト	適用可能機能
ネイティブ・ツール (DNF)	CentOS 8 向けパッチ	ダウンロード・プラグイン

* 「CentOS 5 向けパッチ」サイトおよび「CentOS 6 向けパッチ」サイトは、2016 年 12 月 31 日で非推奨になっています。BigFix Patch では、これらのサイトに関するコンテンツおよびサポートを提供していません。

第 2 章. ダウンロード・プラグインの使用

ダウンロード・プラグインである CentOS プラグインおよび CentOS プラグイン R2 は、関連パッケージをパッチのベンダーから直接ダウンロードする実行可能プログラムです。Fixlet は、内部プロトコルを使用してダウンロード・プラグインと通信することで、ファイルをダウンロードします。これらの Fixlet は、ベンダーが作成する更新を基盤としています。

Fixlet が前述のプロトコルを使用できるようにするには、BigFix サーバーでダウンロード・プラグインを登録します。該当するプラグインを登録するには、「ダウンロード・プラグインの管理 (Manage Download Plug-ins)」ダッシュボードを使用します。



注::

- ダウンロード・プラグインでは、基本認証のみがサポートされます。
- null エラーを回避するために、BigFix サーバーと BigFix クライアントは必ず同じバージョンにしてください。

表 4. CentOS ダウンロード・プラグイン

ダウンロード・プラグイン名	適用可能サイト
CentOS プラグイン R2	CentOS6 プラグイン R2 向けパッチ (Patches for CentOS6 Plugin R2) CentOS7 プラグイン R2 向けパッチ (Patches for CentOS7 Plugin R2) CentOS 8 向けパッチ
CentOS プラグイン	CentOS 5 ネイティブ・ツール向けパッチ CentOS 6 ネイティブ・ツール向けパッチ CentOS 7 向けパッチ



重要: 「CentOS プラグイン」、「CentOS 6 ネイティブ・ツール向けパッチ」サイト、および「CentOS 7 向けパッチ」サイトは、2017 年 9 月で非推奨になります。パッチ適用を継続的に実行し、BigFix コンテンツに途切れなくアクセスできるようにするには、上記コンポーネントの **CentOS プラグイン R2** 出来るだけ早く切り替える必要があります。

CentOS プラグイン R2 は、ベンダーの Web サイトから BigFix サーバーにパッチを直接ダウンロードしキャッシュします。これにより、パッケージ依存関係の解決およびリポジトリに対するサポートの正確性と信頼性が向上します。



注: CentOS プラグイン R2 は、BigFix 管理ツールの「**SHA-256 ダウンロードが必要**」オプションが有効になっている場合は機能しません。このオプションが有効になっていると、すべてのダウンロードの検証で、SHA-256 アルゴリズムのみが使用されます。しかし、リポジトリ内には、プラグインで使用するパッケージの SHA-256 値が含まれていない、ベンダーがらの特定のリポジトリ・メタデータが存在します。

パッチを正常にデプロイするために、「**SHA-256 ダウンロードが必要**」オプションを無効にすることを確認してください。パッケージの GPG シグニチャーを使用して別の層の検査および検証が実行されるため、セキュリティおよびパッケージの整合性が低下することはありません。ダウンロード・オプションについて詳しくは、BigFix Platform インストール・ガイド (https://help.hcl-software.com/bigfix/9.5/platform/Platform/Installation/c_security_settings.html) を参照してください。

ダウンロード・プラグインは拡張性および堅牢性が非常に高く、以下のようなことを可能にします。

- ユーザーによる拡張が可能なリポジトリ・リストを通じて、使用可能なリポジトリをカスタマイズする。
- インストールおよび依存関係解決の対象を、出荷時にサポートされているリポジトリだけでなく、すべてのリポジトリに簡単な操作で拡張できる。
- お客様やサービス・チームが簡単に機能を拡張できる。

これにより、大量のパッケージをダウンロードする際のパフォーマンスも向上するため、結果としてパッチ適用のターンアラウンド・タイムを短縮できます。

改訂された CentOS プラグインは、以下の URL にアクセスします。

- <http://mirror.centos.org>
- <http://vault.centos.org>

「ダウンロード・プラグインの管理」ダッシュボードの概要

「ダウンロード・プラグインの管理」ダッシュボードを使用して、デプロイメント内のダウンロード・プラグインの監視と管理を行います。

「ダウンロード・プラグインの管理」ダッシュボードを使用すると、さまざまなパッチ・ベンダーのダウンロード・プラグインの登録、登録解除、構成、およびアップグレードを実行できます。

「パッチ・サポート (Patching Support)」サイトにサブスクライブして、ダッシュボードにアクセスする必要があります。「ダウンロード・プラグインの管理」ダッシュボードを表示するには、「**パッチ管理ドメイン**」 > 「**すべてのパッチの管理**」 > 「**ダッシュボード**」 > 「**ダウンロード・プラグインの管理**」に進みます。

図 2. 「パッチ管理」ナビゲーション・ツリー

ダッシュボードには、デプロイメント内のすべてのサーバーとリレー (Windows のみ) が表示されます。サーバーまたはリレーを選択すると、そのコンピューターのすべてのプラグインが表示されます。ダッシュボードには、統合された 1 つのビューに、各プラグインのバージョンと状況も表示されます。

図 3. 「ダウンロード・プラグインの管理」ダッシュボード

プラグインは、以下のいずれかの状態にあります。

- インストールされていません
- 新規バージョンが使用可能
- 最新
- サポートされません

このダッシュボードには、ライブ・キーワード検索機能が用意されています。サーバー、リレー、およびプラグインの命名規則に基づいて検索できます。



注: ダウンロード・プラグインを BigFix リレーにインストールする場合、ダウンロードの問題を回避するためにダウンロード・プラグインを BigFix サーバーにもインストールする必要があります。

CentOS ダウンロード・プラグイン R2 の登録

「ダウンロード・プラグインの管理」ダッシュボードを使用して、「CentOS6 プラグイン R2 向けパッチ (Patches for CentOS6 Plugin R2)」サイトおよび「CentOS7 プラグイン R2 向けパッチ (Patches for CentOS7 Plugin R2)」サイトから CentOS パッチをインストールするための CentOS ダウンロード・プラグイン R2 を登録します。

以下のタスクを実行する必要があります。

- Ensure that the BigFix server and the BigFix client are on the same version to avoid a null error.
- 「**パッチ・サポート (Patching Support)**」サイトにサブスクライブして、「ダウンロード・プラグインの管理」ダッシュボードにアクセスします。
- 「**BES サポート**」サイトから使用できる「**クライアントの暗号化分析 (Encryption Analysis for Clients)**」分析をアクティブにします。
- 「**パッチ・サポート (Patching Support)**」サイトから使用できる「**ダウンロード・プラグインのバージョン (Download Plug-in Versions)**」分析をアクティブにします。
- エンドポイントを暗号化する場合、「**BES サポート**」サイトから使用できる「**クライアントの暗号化の有効化 (Enable Encryption for Clients)**」Fixlet を適用します。

ダウンロード・プラグインを、そのプラグインがインストールされていないコンピューターに登録すると、プラグインは自動的にインストールされ、構成ファイルが作成されます。

ダウンロード・プラグインがすでにコンピューターにインストールされている場合、構成ファイルは上書きされません。

1. 「パッチ管理」ドメインから、「すべてのパッチの管理」 > 「ダッシュボード」 > 「ダウンロード・プラグインの管理」ダッシュボードをクリックします。
2. 「サーバーとリレー」テーブルから、ダウンロード・プラグインを登録するサーバーを選択します。

 **重要:** 常にダウンロード・プラグインを BigFix サーバー上に登録する必要があります。

3. 「プラグイン」テーブルから、「CentOS プラグイン R2」を選択します。
4. 「登録」をクリックします。
「CentOS プラグイン R2 の登録 (Register CentOS Plug-in R2)」ウィザードが表示されます。

図 4. 「CentOS プラグイン R2 の登録 (Register CentOS Plug-in R2)」ウィザード

5. プロキシ・サーバーを経由してダウンロードする必要がある場合は、プロキシ・パラメーターを入力してください。

 **注:** 基本認証のみがサポートされています。

プロキシ URL

プロキシ・サーバーの URL。プロトコルとホスト名が含まれた整形式の URL である必要があります。この URL は通常、プロキシ・サーバーの IP アドレスまたは DNS 名とそのポートを、コロンで区切ったものです。例: `http://192.168.100.10:8080`。

プロキシ・ユーザー名

プロキシ・サーバーで認証が必要な場合のプロキシ・ユーザー名。

プロキシ・パスワード

プロキシ・サーバーで認証が必要な場合のプロキシ・パスワード。

「プロキシ・パスワードの確認」

確認用のプロキシ・パスワード。

6. 「OK」をクリックします。
「アクションの実行」ダイアログが表示されます。
7. ターゲット・コンピューターを選択します。
8. 「OK」をクリックします。

CentOS ダウンロード・プラグイン R2 が正常に登録されました。

CentOS ダウンロード・プラグイン R2 の基本設定の構成

「ダウンロード・プラグインの管理」ダッシュボードを使用して、CentOS プラグイン R2 のプロキシ設定を構成します。



注: Ensure that the BigFix server and the BigFix client are on the same version to avoid a null error.

このタスクの適用範囲は、BigFix コンソールから行う基本的な CentOS プラグイン R2 の構成に限られます。CentOS プラグイン R2 の詳細設定を構成するには、`plugin.ini` ファイルを編集する必要があります。この実行方法について詳しくは、[CentOS ダウンロード・プラグイン R2 の詳細設定の構成 \(ページ 21\)](#)を参照してください。

ダウンロード・プラグイン用の既存の構成内容をメモしておくことをお勧めします。ダウンロード・プラグインを構成すると既存の構成が上書きされます。

1. 「パッチ管理」ドメインから、「**すべてのパッチの管理**」>「**ダッシュボード**」>「**ダウンロード・プラグインの管理**」**ダッシュボード**をクリックします。
2. 「サーバーとリレー」テーブルから、ダウンロード・プラグインを構成するサーバーを選択します。
3. 「プラグイン」テーブルから、「**CentOS プラグイン R2**」を選択します。
4. 「**構成**」をクリックします。
「CentOS プラグイン R2 の構成 (Configure CentOS Plug-in R2)」ウィザードが表示されます。

図 5. 「CentOS プラグイン R2 の構成 (Configure CentOS Plug-in R2)」ウィザード

5. プロキシ・サーバーを経由してダウンロードする必要がある場合は、プロキシ・パラメーターを入力してください。



注: 基本認証のみがサポートされています。

プロキシ URL

プロキシ・サーバーの URL。プロトコルとホスト名が含まれた整形式の URL である必要があります。この URL は通常、プロキシ・サーバーの IP アドレスまたは DNS 名とそのポートを、コロンで区切ったものです。例: `http://192.168.100.10:8080`。

プロキシ・ユーザー名

プロキシ・サーバーで認証が必要な場合のプロキシ・ユーザー名。

プロキシ・パスワード

プロキシ・サーバーで認証が必要な場合のプロキシ・パスワード。

「プロキシ・パスワードの確認」

確認用のプロキシ・パスワード。

6. 「**OK**」をクリックします。
「アクションの実行」ダイアログが表示されます。
7. ターゲット・コンピューターを選択します。
8. 「**OK**」をクリックします。

CentOS ダウンロード・プラグイン R2 が正常に構成されました。

CentOS ダウンロード・プラグイン R2 の詳細設定の構成

詳細な構成を行うには、`plugin.ini`という CentOS ダウンロード・プラグイン R2 の構成ファイルを手動で編集します。

`plugin.ini` ファイルは、「ダウンロード・プラグインの管理 (Manage Download Plug-in)」ダッシュボードからダウンロード・プラグインが登録されたときに、自動的に作成されます。このファイルには、ロギングやキャッシュに関する設定の他、リポジトリ・リスト・ファイルを拡張するためのカスタム構成も含まれています。

Linux システムでは、このファイルはダウンロード・プラグインが格納されているルート・ディレクトリーのツリー内にあります。たとえば、`/var/opt/BESServer/DownloadPlugins/CentOSR2Protocol` です。

Windows システムでは、このファイルは BigFix サーバーのインストール・ディレクトリー内にあります。例えば、`%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins\CentOSR2Protocol` です。

図 6. CentOS ダウンロード・プラグイン R2 構成ファイルの例

```
[Logger]
file = logs/CentOSPluginR2.log
level = INFO

[UA]
proxy =
proxyUser =
proxyPass =

primaryRepoListFile = C:\Program Files (x86)\BigFix Enterprise\BES Server\
    GatherDBData\gather\Patching Support\CurrentSiteData\DLCentOSRepoList.json

extendedRepoListFile =
onlyUseExtendedRepoListFile = no

localCache =
localCacheOnly = no
```



注: `plugin.ini` は、大括弧で示される複数のセクションに分かれています。オプションは必ず正しいセクションで指定するようにしてください。オプションを別のセクションに移動すると、エラーが発生することがあります。

パスを指定する必要があるオプション `file`、`primaryRepoListFile`、`extendedRepoListFile`、および `localCache` では、絶対パスと相対パスのどちらでも使用できます。相対パスは、ダウンロード・プラグインの実行可能ディレクトリーを基準とする相対パスです。デフォルトでは、実行可能ファイルは `DownloadPlugins\CentOSR2Protocol` フォルダーにあります。

ロギング・レベルの設定

ロギング・レベルにより、CentOSPluginR2.log ファイルに書き込まれる詳細情報の量が決まります。

使用可能なロギング・レベルは以下のとおりです。

ERROR

ダウンロード・プラグインの実行に関連するエラーが出力されます。このようなエラーは、致命的エラーが発生する直前であることを示している場合があります。

WARNING

ダウンロードの失敗と、失敗の理由に関する情報が出力されます。

INFO

ダウンロードの進行状況とダウンロードの成功に関する一般情報が、最小限のトレース情報とともに出力されます。

DEBUG

問題のトラブルシューティングに使用される詳細情報が出力されます。これは、使用可能なレベルの中で最も詳細なレベルです。

ロギング・レベル・オプションは、[Logger] ファイルの plugin.ini セクションで変更できます。

```
[Logger]
file = logs/CentOSPluginR2.log
level = INFO
```

たとえば、ロギングが INFO に設定されている場合は、そのレベルと、そのレベルの上に記載されているすべてのレベルのログが、ロガーによって出力されます。この場合は、INFO、WARNING、および ERROR の各ログが出力されます。



注: ロギング・レベルを DEBUG に設定すると、ログに記録する情報の量が増えるため、パフォーマンスに影響が及ぶ可能性があります。ロギング・レベルを DEBUG に上げるのは、問題を調査するときだけにとどめ、問題の解決後は INFO または WARNING に戻しておいてください。

拡張リポジトリ・リスト・ファイルの追加

CentOS ダウンロード・プラグイン R2 は、必要に応じて、BigFix が正式にサポートしていないリポジトリと連動するように構成できます。

そのようなリポジトリをサポートするようにダウンロード・プラグインを構成する方法について詳しくは、[CentOS ダウンロード・プラグイン R2 の拡張 \(ページ 24\)](#)を参照してください。

ダウンロード・キャッシュの設定

ダウンロード・キャッシュ・ツールを使用すると、指定した場所にパッケージとリポジトリ・メタデータをダウンロードできます。

ダウンロード・キャッシャーを構成できるシナリオとして、次の 3 つが考えられます。

隔離された環境での Sha1 ダウンロード機能

ダウンロード・キャッシャー・ツールは主に、隔離された環境用に設計されています。このような環境では、セキュアなネットワークが必要であるために、ベンダーのサイトから直接ファイルを直接ダウンロードするためにインターネットにアクセスすることができません。

sha1 ダウンロード機能により、BigFix サーバーの sha1 フォルダーにパッケージが直接キャッシングされるので、パフォーマンスが向上します。

詳しくは、[隔離された環境での CentOS ダウンロード・キャッシャー R2 の使用 \(\(ページ\) 35\)](#)を参照してください。

インターネット対応 BigFix サーバーでの Sha1 ダウンロード機能

この方法は、インターネット対応の BigFix サーバーが含まれている環境にパッケージをキャッシュする場合のベスト・プラクティスと考えられます。sha1 ダウンロード機能により、BigFix サーバーの sha1 フォルダーにパッケージが直接キャッシングされるので、パフォーマンスが向上します。

詳しくは、[sha1 フォルダーでのパッケージのキャッシング \(\(ページ\) 35\)](#)を参照してください。

インターネット対応の BigFix サーバーで sha1 ダウンロード機能を使用しない

何らかの理由で、sha1 ダウンロード機能による BigFix サーバーの sha1 フォルダーへのパッケージのキャッシングを行わないことにした場合は、ローカル・キャッシュを使用できます。詳しくは、[ローカル・キャッシュ・フォルダーでのパッケージのキャッシング \(\(ページ\) 37\)](#)を参照してください。

CentOS ダウンロード・プラグイン R2 の登録解除

「ダウンロード・プラグインの管理」ダッシュボードを使用して、CentOS ダウンロード・プラグイン R2 を登録解除します。

1. 「パッチ管理」ドメインから、「**すべてのパッチの管理**」 > 「**ダッシュボード**」 > 「**ダウンロード・プラグインの管理**」ダッシュボードをクリックします。
2. 「サーバーとリレー」テーブルから、ダウンロード・プラグインを登録解除するサーバーを選択します。
3. 「プラグイン」テーブルから、「**CentOS プラグイン R2**」を選択します。
4. 「**登録解除 (Unregister)**」をクリックします。

図 7. CentOS ダウンロード・プラグイン R2 の登録解除

「アクションの実行」ダイアログが表示されます。

5. ターゲット・コンピューターを選択します。
6. 「**OK**」をクリックします。

CentOS ダウンロード・プラグインが正常に登録解除されました。

CentOS ダウンロード・プラグイン R2 のアップグレード

「ダウンロード・プラグインの管理」ダッシュボードを使用して、CentOS ダウンロード・プラグイン R2 を新しいバージョンにアップグレードします。

1. 「パッチ管理」ドメインから、「**すべてのパッチの管理**」 > 「**ダッシュボード**」 > 「**ダウンロード・プラグインの管理**」ダッシュボードをクリックします。
2. 「サーバーとリレー」テーブルから、ダウンロード・プラグインをアップグレードするサーバーを選択します。
3. 「プラグイン」テーブルから、「**CentOS プラグイン R2**」を選択します。
4. 「**アップグレード**」をクリックします。
「アクションの実行」ダイアログが表示されます。
5. ターゲット・コンピューターを選択します。
6. 「**OK**」をクリックします。



注: ダウンロード・プラグインを再構成する必要があります。



注: 最新バージョンのダウンロード・プラグインは、プロキシ・パスワードとベンダー・パスワードの保存に関するセキュリティを增強するために強化されています。

これで、CentOS ダウンロード・プラグイン R2 の最新バージョンがインストールされました。

CentOS ダウンロード・プラグイン R2 の拡張

CentOS ダウンロード・プラグイン R2 は、BigFix が正式にサポートしていないリポジトリからパッケージをダウンロードしてキャッシュするように構成できます。CentOS プラグイン R2 は、サポートされていない CentOS リポジトリのパッケージ依存関係を解決できるため、サポートされていない依存パッケージをインストールすることができます。この拡張ソリューションにより、BigFix を使用して、出荷時にサポートされていないリポジトリからパッケージを適用できます。

操作を始める前に、以下の作業を完了しておく必要があります。

- 適切なライセンスが登録されていること、および CentOS サブスクリプションがアクティブであることを確認します。
- BigFix Patch for CentOS のサポート情報を調べて、サポートされるリポジトリのリストを確認します。
 - [サポートされるプラットフォームおよび更新 \(ページ 10\)](#)
 - [サポートされる CentOS リポジトリ \(ページ 12\)](#)

BigFix で正式にサポートされていないリポジトリを使用するようにプラグインを拡張するには、以下の手順に注意深く従う必要があります。

1. 拡張リポジトリ・リスト・ファイルを作成する ((ページ) 25)。
2. CentOS ダウンロード・プラグイン R2 構成ファイルを更新する ((ページ) 27)。
3. 拡張リポジトリのセットアップを検証する ((ページ) 28)。

ステップ 1: 拡張リポジトリ・リスト・ファイルの作成

CentOS プラグイン R2 では、サブスクリプションに組み込まれている基本製品と拡張製品の両方の追加リポジトリが含まれた、拡張リポジトリ・リスト・ファイルを使用できます。

拡張リポジトリ・リスト・ファイルは、以下の基準を満たしている必要があります。

- これは有効な `.json` ファイルでなければなりません。
- CentOS プラグイン R2 からアクセスできること。管理者ユーザーまたは root ユーザーがファイルを読み取ることができるようにしてください。
- 以下のフォーマットが含まれていること。

```
{CentOS Plug-in R2
  "<OS_key>": [
    {"name": "<name>", "baseurl": "<base_url>"},
    {"name": "<name>", "baseurl": "<base_url>"}
  ],
  "<OS_key_1>": [
    {"name": "<name>", "baseurl": "<base_url>"}
  ]
}
```



注: このファイルは、`DLCentOSRepoList.json` ファイルの `repoListFile` で設定される `plugin.ini` ファイルと同じ形式を使用します。

不等号括弧 `<>` で囲まれたプレースホルダーは、実際値に置き換える必要があります。

OS_key

BigFix の CentOS パッチ適用コンテンツでサポートされていないリポジトリを使用するには、`DLCentOSRepoList.json` ファイルでリストされている以下の OS キーを使用します。

- centos-8.1.1911-x64
- centos-8.0.1905-x64
- centos-8-x64
- centos-7.6.1810-x64
- centos-7.5.1804-x64
- centos-7.4.1708-x64
- centos-7.3.1611-x64
- centos-7.2.1511-x64

- centos-7.1.1503-x64
- centos-7.0.1406-x64
- centos-7-x64
- centos-6.8-x64
- centos-6.7-x64
- centos-6.6-x64
- centos-6.5-x64
- centos-6.4-x64
- centos-6.3-x64
- centos-6.2-x64
- centos-6.1-x64
- centos-6.0-x64
- centos-6-x64
- centos-6.8-x32
- centos-6.7-x32
- centos-6.6-x32
- centos-6.5-x32
- centos-6.4-x32
- centos-6.3-x32
- centos-6.2-x32
- centos-6.1-x32
- centos-6.0-x32
- centos-6-x32

ダウンロード・プラグインの新バージョンがリリースされているため、これは完全なリストではない可能性があります。最新版の完全なリポジトリ・リストを入手するには、以下の操作を実行します。

1. エンドポイントが最新のダウンロード・プラグインに登録されているかどうかを確認します。
プラグインのダウンロード管理ダッシュボードに、プラグインが最新であるか、新しいバージョンが利用可能であるかが表示されます。
2. 以下の場所から `DLCentOSRepoList.json` ファイルを表示します。

Windows システムの場合

```
%PROGRAM FILES%\BigFix Enterprise\BES Server\GatherDBData
\gather\Patching Support\CurrentSiteData
```

Linux システムの場合

```
/var/opt/BEServer/gatherDBData/gather/Patching Support/
CurrentSiteData
```

 **重要:** ダウンロード時や依存関係の解決時に問題が起こらないようにするために、各リポジトリに対応する正しい OS キーを使用してください。

base_url

基本 URL は、CentOS のミラーおよびボルト・リポジトリ・リストから形成されます。

```
http://mirror.centos.org/centos/<CentOS_version>/<repo_name>/<architecture>
```

```
http://vault.centos.org/<CentOS_version>/<repo_name>/<architecture>
```

以下は、指定された形式の `.json` ファイルの例です。

```
{
  "centos-7-x64": [
    {"name": "EPEL 7", "baseurl": "https://dl.fedoraproject.org/pub/epel/7/x86_64"}
  ],
  "centos-6-x64": [
    {"name": "EPEL 6", "baseurl": "https://dl.fedoraproject.org/pub/epel/6/x86_64"}
  ]
}
```

このファイルには「`epel_repos.json`」という名前を付け、このセクション全体の例で使用します。

ステップ 2: CentOS ダウンロード・プラグイン R2 構成ファイルの更新

拡張リポジトリ・リスト・ファイルを使用するように CentOS プラグイン R2 を構成します。「ダウンロード・プラグインの管理」ダッシュボードから CentOS プラグイン R2 の登録解除や構成を行うと、`plugin.ini` 構成ファイルが上書きされます。構成ファイルで行った変更はすべて失われるため、以前に行った変更を書き留めておいてください。

1. テキスト・エディターを使用して、`plugin.ini` ファイルを以下の場所から開きます。

Windows システムの場合

```
%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins
\CENTOSR2Protocol
```

Linux システムの場合

```
/var/opt/BESServer/DownloadPlugins/CENTOSR2Protocol
```

2. `extendedRepoListFile` フィールドに、拡張リポジトリ・リスト・ファイルの絶対パスまたは相対パスを入力します。相対パスに設定する場合は、CentOS プラグイン R2 の実行可能プログラムの場所を基準としたパスを指定する必要があります。

例:

```
extendedRepoListFile = ./epel_repos.json
```

この例は、`epel_repos.json` ファイルがダウンロード・プラグイン・バイナリーと同じディレクトリーにあることを示しています。

3. 拡張リポジトリ・リストだけを使用するように CentOS プラグイン R2 を設定するには、**onlyUseExtendedRepoListFile** フィールドを **yes** に設定します。

例:

```
onlyUseExtendedRepoListFile= yes
```

両方のリポジトリ・リスト・ファイルを使用するように CentOS プラグイン R2 を設定するには、この設定を **no** に構成します。両方のファイルで同じ OS キーが使用されている場合は、リポジトリが結合されます。

4. ファイルを保存します。

ステップ 3: 拡張リポジトリのセットアップの検証

CentOS ダウンロード・キャッシャー R2 を使用して、拡張リポジトリが正しくセットアップされているかどうか、およびそれが使用可能であることを検証します。キャッシャーの詳細については、[CentOS ダウンロード・キャッシャー R2 の使用情報 \(ページ 29\)](#) を参照してください。

これを行うには、`CentOSR2DownloadCacher.exe --check-allrepos` コマンドを実行します。

セットアップに問題がない場合は、次のような出力になります。

```
3488      : 2018-02-28 17:23:36 : INFO      : Testing access to: centos-6-x64
3488      : 2018-02-28 17:23:36 : INFO      : EPEL_6
3488      : 2018-02-28 17:23:36 : INFO      : Success!
3488      : 2018-02-28 17:23:36 : INFO      : CentOS_-_OS
3488      : 2018-02-28 17:23:36 : INFO      : Success!

3488      : 2018-02-28 17:23:36 : INFO      : Testing access to: centos-7-x64
3488      : 2018-02-28 17:23:36 : INFO      : EPEL_7
3488      : 2018-02-28 17:23:36 : INFO      : Success!
3488      : 2018-02-28 17:23:36 : INFO      : CentOS_-_OS
3488      : 2018-02-28 17:23:36 : INFO      : Success!
```

第 3 章. ダウンロード・キャッシュの使用

ダウンロード・キャッシュは、必要なファイルをダウンロードしてキャッシュすることを目的として設計された、スタンドアロンのキャッシング用コマンド・ライン・ツールです。事前キャッシュされたファイルは、ダウンロード・プラグインがエンドポイントにパッチを適用するために使用することができます。

ダウンロード・キャッシュは、隔離された環境を対象に使用することを目的として設計されています。このツールを使用して、Fixlet で必要となる大量のパッケージをダウンロードおよびキャッシュします。ファイルを事前キャッシュすることで、BigFix クライアントにファイルを配布する前にそれらのファイルをインターネットからダウンロードする必要がなくなるため、アクションの実行速度が向上します。



注: BigFix サーバーがインターネットにアクセスできる場合は、ダウンロード・プラグインを使用します。「ダウンロード・プラグインの管理」ダッシュボードからダウンロード・プラグインを登録する必要があります。

ダウンロードしてこのツールにアクセスし、手動で実行することができます。

CentOS ダウンロード・キャッシュ R2 の使用情報

CentOS ダウンロード・キャッシュ R2 を使用すると、隔離された環境で CentOS パッチをダウンロードし、キャッシュに入れることができます。このツールは、「**CentOS6 プラグイン R2 向けパッチ (Patches for CentOS6 Plugin R2)**」サイトおよび「**CentOS7 プラグイン R2 向けパッチ (Patches for CentOS7 Plugin R2)**」サイトをサポートします。

CentOS ダウンロード・キャッシュ R2 は、<http://mirror.centos.org> および <http://vault.centos.org> にアクセスして、更新用のパッケージをダウンロードします。ダウンロード・エラーを回避するため、リストされているホストでのトラフィックを許可するようにホワイトリスト、ファイアウォール、またはプロキシーを構成してください。

CentOS ダウンロード・キャッシュ R2 は、Windows システムまたは Linux システム上で実行できます。要件については、「[BigFix 10.0 - System Requirements](#)」を参照してください。

最新の CentOS ダウンロード・キャッシュ R2 は、以下の BigFix サポート・サイトから入手できます。

- Windows システムを使用する場合は、<https://software.bigfix.com/download/bes/util/CentOSR2DownloadCacher.exe> でこのツールをダウンロードしてください。
- Linux システムを使用する場合は、<https://software.bigfix.com/download/bes/util/CentOSR2DownloadCacher-linux.tar.gz> でこのツールをダウンロードしてください。このツールは、x86-64 (64 ビット) システムでサポートされます。



注: このツールを正しく使用するには、以下のパッケージとその依存関係を必ずインストールしてください。



- GLIBC バージョン 2.2.5 以降
- GLIBC バージョン 2.3 以降
- GNU/Linux カーネルのバージョン 2.6.31 以降

説明のために、このセクションでは、Windows での CentOS ダウンロード・キャッシュ R2 の実行手順を示します。ただし、CentOS ダウンロード・キャッシュ R2 を実行するためのパラメーターとサブコマンドは、Windows システムでも Linux システムでも同じです。

ツール `CentOSR2DownloadCacher.exe` により、追加の操作を実行することができます。このツールをコマンド・プロンプトから実行するには、以下のコマンドを使用します。

```
CentOSR2DownloadCacher.exe [-h] [parameters...] {subcommand} [subparameters...]
```

各コマンドの説明。

-h

コマンドを実行せずに、そのコマンドのヘルプ・メッセージを表示するよう指定します。

parameters

ダウンロード・キャッシュを構成するために使用するオプションのパラメーターを指定します。

--proxyServer

使用するプロキシ・サーバーの URL を指定します。これは、プロトコルとホスト名を含む整形形式の URL でなければなりません。この URL は通常、プロキシ・サーバーの IP アドレスまたは DNS 名とそのポートを、コロンで区切ったものです。例:
http://192.168.100.10:8080

--proxyUser

プロキシ・サーバーで認証が必要な場合は、プロキシ・ユーザー名を指定します。

--proxyPass

プロキシ・サーバーで認証が必要な場合は、プロキシのパスワードを指定します。

基本認証のみがサポートされています。

--download_dir

リポジトリ・メタデータ・ファイルがキャッシュされるディレクトリーを指定します。

このパラメーターが定義されていない場合は、ダウンロード・キャッシュの実行可能ディレクトリーを基準とした相対ディレクトリーにファイルがダウンロードされます。

キャッシュされたファイルを使用するように CentOS ダウンロード・プラグイン R2 を構成できます。これは、`localCache` ファイルに `plugin.ini` を設定することで行います。

--sha1_download_dir

パッケージを sha1 のファイル名で単一のフラット・ディレクトリーにキャッシュする場合のディレクトリーを指定します。キャッシュャーは、すべてのリポジトリー (キー) からすべてのパッケージを、指定されたディレクトリーにファイルとしてダウンロードします。

パッケージのみが `sha1_download_dir` に保管されます。各リポジトリー・メタデータは、`download_dir` に保管され、CentOS リポジトリーのディレクトリー構造が維持されます。

`--sha1_download_dir` サブコマンドで `check-storage` を使用することで、スペース節約のベンチマークが設定されます。`--sha1_download_dir` を使用することで、同じ CentOS バージョンの複数のリポジトリーをキャッシュする際のストレージ・サイズ、ダウンロード・サイズ、および時間が大幅に削減されます。これは、同じ CentOS バージョン (たとえば、`centos-6.8-x64`、`centos-6.7-x64`、`centos-6.6-x64`) のリポジトリー間で、多くのパッケージが重複しているためです。CentOS バージョン (たとえば、`centos-6.8-x64`、`centos-7.1-x64`) ごとにリポジトリーを 1 つだけキャッシュする場合、スペースは節約されません。



注: このパラメーターを使用する場合は、BigFix サーバーの sha1 ファイル・フォルダーのキャッシュ制限を考慮してください。

--redownload

RPM ファイルを再ダウンロードして、ダウンロード・ディレクトリー内の既存の RPM ファイルを上書きすることを示すフラグを指定します。

このパラメーターが定義されていない場合は、RPM ファイルが再ダウンロードされません。ただし、メタデータはデフォルトでダウンロードされ上書きされます。

--verifyExistingPkgChecksum

「`buildRepo`」、 「`downloadPkg`」、または「`downloadbypatchid`」の各サブコマンドを使用してパッケージのダウンロードを試みる際に、既存の RPM ファイルにチェックサム・チェックを強制するフラグを指定します。



注: チェックサムは、デフォルトで「オフ」に設定されています。

--loglevel

ログ・レベルを指定します。「`DEBUG`」、「`INFO`」、「`WARNING`」、または「`ERROR`」の中から選択できます。デフォルトでは、この値は「`INFO`」に設定されています。

INFO

ダウンロードの進行状況とダウンロードの成功に関する一般情報が、最小限のトレース情報とともに出力されます。

WARNING

ダウンロードの失敗と、失敗の理由に関する情報が出力されます。

ERROR

ダウンロード・プラグインの実行に関連するエラーが出力されます。このようなエラーは、致命的エラーが発生する直前であることを示している場合があります。

DEBUG

問題のトラブルシューティングに使用される詳細情報が出力されます。これは、使用可能なレベルの中で最も詳細なレベルです。

--help

コマンドを実行せずに、そのコマンドの完全な説明とヘルプを表示するよう指定します。

subcommand subparameter

ダウンロード・キャッシュを実行するために使用するサブコマンドとサブパラメーターを指定します。



注: サブコマンドとサブパラメーターの名前では、大/小文字が区別されます。

subparameter は以下のように *subcommand* によって異なります。

check-baserepos

BigFix でサポートされる CentOS 基本リポジトリにアクセスできるかどうかを検査します。結果はコマンド・プロンプトに表示され、**<cache directory>\logs\CentOSR2DownloadCacher.log** ファイルに格納されます。

check-allrepos

BigFix でサポートされる CentOS 基本リポジトリおよびサブリポジトリにアクセスできるかどうかを検査します。結果はコマンド・プロンプトに表示され、**<cache directory>\logs\CentOSR2DownloadCacher.log** ファイルに格納されます。

check-storagereq

builRepo コマンドで `--sha1_download_dir` オプションを指定する場合と指定しない場合のストレージ・スペース所要量を調べます。結果はコマンド・プロンプトに表示され、**<cache directory>\logs\CentOSR2DownloadCacher.log** ファイルに格納されます。

showKeys

サポートされるリポジトリの OS キーのリストを **<cachier directory>\logs**
\CentOSR2DownloadCacher.log ファイルに出力します。OS キーは、単一 CentOS リ
 ポジトリの CentOS オペレーティング・システムのバージョンおよびアーキテクチャー
 を示します。

このサブコマンドを実行するための構文は以下のとおりです。

```
CentOSR2DownloadCacher.exe --download_dir <download_dir>
[parameters] showsKeys
```

例: CentOSR2DownloadCacher.exe --download_dir C:\downloads showKeys

buildRepo

指定された OS キーに基づいて、ローカルのミラー・リポジトリを作成し、すべての関
 連ファイルをダウンロードします。

このサブコマンドを実行するための構文は以下のとおりです。

```
CentOSR2DownloadCacher.exe --download_dir <download_dir>
--sha1_download_dir <sha1_download_dir> [parameters]
buildRepo --key <OS_key1,OS_key2,...>
```

例: CentOSR2DownloadCacher.exe --download_dir C:\downloads --sha1_download_dir C:
 \sha1_downloads buildRepo --key centos-7-x64, centos-6-x32, centos-6.8-x32

各コマンドの説明。

--key OS_key1,OS_key2,...

CentOS オペレーティング・システムのバージョンおよびアーキテクチャー
 を指定します。項目はコンマで区切る必要があり、スペースを含めることは
 できません。以下のフォーマットを使用する必要があります。

```
<product>-<version_number>-<architecture>
```

例えば、`--key centos-7-x64` です。

downloadMetadataOnly

指定された OS キーのメタデータをダウンロードします。

このサブコマンドを実行するための構文は以下のとおりです。

```
CentOSR2DownloadCacher.exe --download_dir <download_dir>
[parameters] downloadMetadataOnly --key <OS_key1,OS_key2,...>
```

例:

```
CentOSR2DownloadCacher.exe --download_dir C:\downloads downloadMetadataOnly
--key centos-7-x64
```

各コマンドの説明。

--key OS_key1,OS_key2,...

CentOS オペレーティング・システムのバージョンおよびアーキテクチャーを指定します。項目はコンマで区切る必要があり、スペースを含めることはできません。以下のフォーマットを使用する必要があります。

```
<product>-<version_number>-<architecture>
```

例えば、`--key centos-7-x64` です。

downloadPkg

指定された OS キーのリストされた RPM ファイルをダウンロードします。



注: ダウンロードするパッケージに依存関係がある場合は、代わりに buildrepo を使用して依存関係の問題を回避することをお勧めします。

このサブコマンドを実行するための構文は以下のとおりです。

```
CentOSR2DownloadCacher.exe --download_dir <download_dir>
[parameters] downloadPkg --key <OS_key1, OS_key2...>
--pkg <pkg1, pkg2, ...>
```

例:

```
CentOSR2DownloadCacher.exe --download_dir C:\temp --redownload downloadPkg
--key centos-7-x64 --pkg python-qr-code-core-5.0.1-1.el7.noarch.rpm
```

各コマンドの説明。

--key OS_key1,OS_key2,...

CentOS オペレーティング・システムのバージョンおよびアーキテクチャーを指定します。項目はコンマで区切る必要があり、スペースを含めることはできません。以下のフォーマットを使用する必要があります。

```
<product>-<version_number>-<architecture>
```

例えば、`--key python-qr-code-core-5.0.1-1.el7.noarch.rpm` です。

--pkg pkg1,pkg2,...

パッケージ名を示します。

各項目はコンマで区切る必要があり、スペースを含めることはできません。例えば、`--pkg`

```
liblcms1-1.17-77.12.1.x86_64.rpm, liblcms1-32bit-1.17-77.12.1.x86_64.rpm
```

です。

隔離された環境での CentOS ダウンロード・キャッシュ R2 の使用

buildRepo サブコマンドを使用してリポジトリのすべてのパッチを指定のディレクトリにダウンロードすることにより、隔離された環境で CentOS ダウンロード・キャッシュ R2 を使用できます。

- BigFix でサポートされる CentOS 基本リポジトリおよびサブリポジトリにアクセスできることを確認します。確認するには、`check-allrepos` サブコマンドを実行します。
- リポジトリ・メタデータおよびパッケージをダウンロードするための十分なスペースがあることを確認します。必要なストレージ・スペースを調べるには、サブコマンド `check-storagereq` を実行します。
- 以下の手順を実行して、BigFix サーバーの sha1 フォルダー・サイズの制限を大きくします。
 1. BigFix コンソールから、コンピューターを右クリックして、「**コンピューター設定の編集**」を選択します。
 2. **_BESGather_Download_CacheLimitMB** のサイズを増やします。

推奨されるサイズは、現行の BigFix サーバーの sha1 フォルダー・サイズに `sha1_download_dir` のサイズを加えた値です。

`sha1_download_dir` のサイズを事前に決定できない場合、推奨される `sha1_download_dir` のサイズは、リポジトリごとに最小で 20GB です。最小サイズの 20GB は、時間の経過とともに増加される可能性があることに注意してください。

1. buildRepo ファイルで `CentOSR2DownloadCacher.exe` サブコマンドを使用して、リポジトリのすべてのファイルを指定のディレクトリにダウンロードします。例:

```
CentOSR2DownloadCacher.exe --download_dir C:\downloads
--sha1_download_dir C:\sha1_downloads
buildRepo --key centos-6.8-x64, centos-6.7-x64
```

この例で、ダウンロード・キャッシュ・ツールは、「centos-6.8-x64」リポジトリと「centos-6.7-x64」リポジトリの両方について、パッケージを `sha1_download_dir` にダウンロードし、リポジトリ・メタデータを `download_dir` にダウンロードします。

2. `download_dir` を隔離された BigFix サーバーに転送し、`sha1_download_dir` の sha1 ファイルを BigFix Server の sha1 フォルダーに転送します。
3. CentOS ダウンロード・キャッシュ R2 の構成ファイル `plugin.ini` を開きます。`plugin.ini` ファイルに以下の設定を構成して、BigFix サーバーがリポジトリ・メタデータおよび必要なパッケージをオンラインではなくローカル・キャッシュから取得するように設定します。

```
localCache = <download_directory_specified_for_the_--download_dir_parameter>
localCacheOnly = yes
```

sha1 フォルダーでのパッケージのキャッシング

CentOS ダウンロード・キャッシュ R2 を使用して、インターネット対応の BigFix サーバーの sha1 フォルダーにパッケージを直接キャッシングすることにより、Fixlet 適用時のパフォーマンスを改善できます。

- BigFix でサポートされる CentOS 基本リポジトリおよびサブリポジトリにアクセスできることを確認します。確認するには、`check-allrepos` サブコマンドを実行します。
- リポジトリ・メタデータおよびパッケージをダウンロードするための十分なスペースがあることを確認します。必要なストレージ・スペースを調べるには、サブコマンド `check-storagereq` を実行します。
- 以下の手順を実行して、BigFix サーバーの sha1 フォルダー・サイズの制限を大きくします。
 1. BigFix コンソールから、コンピューターを右クリックして、「**コンピューター設定の編集**」を選択します。
 2. **_BESGather_Download_CacheLimitMB** のサイズを増やします。

推奨されるサイズは、現行の BigFix サーバーの sha1 フォルダー・サイズに `sha1_download_dir` のサイズを加えた値です。

`sha1_download_dir` のサイズを事前に決定できない場合、推奨される `sha1_download_dir` のサイズは、リポジトリごとに最小で 20GB です。最小サイズの 20GB は、時間の経過とともに増加される可能性があることに注意してください。

このタスクにより、BigFix サーバーはインターネットを活用して、リポジトリ・メタデータやパッケージなど、必要なファイルが Fixlet 適用時に必ず使用できるようにします。

1. `buildRepo` ファイルで `CentOSR2DownloadCacher.exe` サブコマンドを使用して、リポジトリから BigFix サーバーの sha1 フォルダーにパッケージをダウンロードします。例:

```
CentOSR2DownloadCacher.exe --download_dir C:\downloads --sha1_download_dir
C:\Program Files (x86)\BigFix Enterprise\BES Server\wwwrootbes\bfmirror\downloads\sha1
buildRepo --key centos-7-x64, centos-6-x32
```

この例で、ダウンロード・キャッシュ・ツールは、BigFix サーバーの sha1 フォルダーにパッケージを直接ダウンロードします。

2. CentOS ダウンロード・キャッシュ R2 の構成ファイル `plugin.ini` を開きます。 `plugin.ini` ファイルに以下の設定を構成して、BigFix サーバーがその sha1 フォルダーからパッケージを取得するように設定します。

```
localCache =
localCacheOnly = no
```

BigFix サーバーがインターネットに対応していて、パッケージが BigFix サーバーの sha1 フォルダーに直接保存されている場合は、`localCache` 設定に値を指定しないでください。

最初に BigFix サーバーは、必要なファイルをインターネットから取得する前に、それらがダウンロード・プラグインのキャッシュ・フォルダーおよび sha1 フォルダーにあるかどうかを確認します。ダウンロード・プラグインのキャッシュ・フォルダー内のリポジトリ・メタデータが期限切れの場合は、新しいリポジトリ・メタデータがオンラインでダウンロードされます。必要なパッケージが BigFix サーバーの sha1 フォルダーに存在しない場合、パッケージはインターネットからダウンロードされます。

ローカル・キャッシュ・フォルダーでのパッケージのキャッシング

CentOS ダウンロード・キャッシュ R2 を使用して、インターネット対応の BigFix サーバー上にパッケージをキャッシングすることができます。BigFix サーバーがキャッシュ・フォルダーとインターネットの両方からリポジトリ・メタデータおよびパッケージを使用するように構成できます。

フォルダー・サイズが増大する場合に備えて、BigFix サーバーの sha1 フォルダーではなく別のフォルダーにパッケージを保管します。BigFix サーバーは最新のダウンロードを保管するだけなので、BigFix サーバーの sha1 フォルダー・サイズの制限が小さすぎると、保管されたパッケージは新しいファイルによって置き換えられることがあります。

ローカル・キャッシュにパッケージを保管すると、CentOS ダウンロード・プラグイン R2 は、インターネットから取得せずにそのパッケージを使用します。

BigFix サーバーがローカル・キャッシュからのパッケージを要求する場合、パッケージは BigFix サーバーの sha1 フォルダーにもキャッシュされるため、追加のスペースが必要になる可能性があります。

1. `buildRepo` ファイルで `CentOSR2DownloadCacher.exe` サブコマンドを使用して、リポジトリから BigFix サーバーの sha1 フォルダーにパッケージをダウンロードします。例:

```
CentOSR2DownloadCacher.exe --download_dir C:\downloads  
buildRepo --key centos-7-x64, centos-6-x32
```

この例で、ダウンロード・キャッシュ・ツールは、指定されたダウンロード・フォルダーにパッケージをダウンロードします。

2. CentOS ダウンロード・キャッシュ R2 の構成ファイル `plugin.ini` を開きます。 `plugin.ini` ファイルに以下の設定を構成します。

```
localCache = <location of the transferred download_dir>  
localCacheOnly = no
```

この設定により、BigFix サーバーは最初にダウンロード・プラグインのキャッシュ・フォルダー内のリポジトリ・メタデータを検査します。これが期限切れでなければ、BigFix サーバーはそのメタデータを使用します。それ以外の場合、BigFix サーバーはインターネットからリポジトリ・メタデータを取得します。パッケージの場合、BigFix サーバーは最初に、パッケージが sha1 フォルダーに存在するかどうかを検査し、次に `localCache` を検査してから、インターネットに進みます。

第 4 章. BigFix Patch for CentOS の使用 Linux™

「Linux™ RPM パッチ」サイトおよび各種の「CentOS 向けパッチ」サイトにある Fixlet を使用して、デプロイメント環境にパッチを適用します。

CentOS Linux Enterprise 向けの使用可能な Fixlet サイトについて詳しくは、『[サポートされるプラットフォームおよび更新 \(\(ページ\) 10\)](#)』を参照してください。

パッチ・コンテンツのキャッシングは、エアー・ギャップ環境またはカスタム・リポジトリを使用している場合を除いて、ダウンロード・プラグイン経由で行われなければなりません。詳しくは、以下のトピックを参照してください。

- [ダウンロード・プラグインの使用 \(\(ページ\) 16\)](#)
- [ダウンロード・キャッシュの使用 \(\(ページ\) 29\)](#)

Fixlet を使用したパッチの適用

CentOS Fixlet サイトから入手可能な Fixlet を使用して、CentOS Linux™ パッチをデプロイメントに適用できます。

- 適切な Fixlet サイトをサブスクライブします。
- 「パッチ・サポート」サイトまたは Fixlet サイトから、必要な分析 (「**エンドポイントの依存解決 - 適用結果**」分析など) をアクティブ化して、パッチの適用結果を表示します。
- エンドポイントに GPG キーをインストールし、有効化します。「**RPM-GPG-KEY-centos-release のインポート (Import RPM-GPG-KEY-centos-release)**」タスクを使用すると、必要なキーをインポートできます。
- クライアントの **/var** ディレクトリについて、適切な構成を設定します。
 - **/var** ディレクトリが `noexec` オプションに設定された場合は、パッチ適用の実行可能ファイルを実行するために別のディレクトリを指定してください。そうしないと、パッチ適用は失敗します。「**_BESClient_LinuxPatch_executable_directory のパスの設定 (Set the path for _BESClient_LinuxPatch_executable_directory)**」Fixlet を実行し、代替ディレクトリ名を指定することができます。
 - **/var** ディレクトリのディスク・スペースが限られている場合は、パッチ・メタデータをキャッシングできるスペースがより多いディレクトリを指定してください。「**_BESClient_LinuxPatch_metadata_directory のパスの設定 (Set the path for _BESClient_LinuxPatch_metadata_directory)**」Fixlet を実行し、代替ディレクトリ名を指定することができます。



注: `_BESClient_LinuxPatch_executable_directory` と `_BESClient_LinuxPatch_metadata_directory` 両方の設定で、指定したディレクトリ・パスは有効な絶対パス名でなければなりません。使用できるのは英数字、スラッシュ、および下線のみです。

- リポジトリを使用している場合は、パッチが含まれるリポジトリにエンドポイントを登録します。YUM または「CentOS カスタム・リポジトリ管理」ダッシュボードを使用すると、リポジトリをパッケー

ジ・ソースとして追加できます。このダッシュボードについて詳しくは、[CentOS カスタム・リポジトリ管理 \(\(ページ\) 47\)](#)を参照してください。

- リポジトリを使用していない場合は、適切な CentOS ダウンロード・プラグインを登録します。ダウンロード・プラグインについて詳しくは、[ダウンロード・プラグインの使用 \(\(ページ\) 16\)](#)を参照してください。

CentOS Linux™ パッチ Fixlet® を実行する場合、パッチの適用前にテスト実行を適用することもできます。「適用の結果」分析を表示して、依存関係が正常に解決したかどうか、およびインストールが成功するかどうかを判断できます。



注: テスト実行機能を使用すると、テストに失敗した場合でも、アクションは「修正済み」と報告します。

kernel Fixlet には、すべての kernel パッケージをアップグレードまたはインストールするオプションが用意されています。アップグレード・オプションを使用すると、既存の kernel パッケージがそれより後のバージョンで置き換えられます。インストール・オプションでは、以前のバージョンの次に新しい kernel パッケージがインストールされます。kernel 更新のデフォルトの動作は、パッケージの横並びでのインストールです。また、各 kernel 更新 Fixlet® には、これらの各オプションをテストする機能が用意されています。

1. 「パッチ管理」ドメインから、「**OS ベンダー**」 > 「**CentOS**」をクリックし、ドメイン・ノードを使用してパッチ・コンテンツに移動します。

図 8. 「パッチ管理」ナビゲーション・ツリー

2. リスト・パネルに表示されたコンテンツの中から、適用する Fixlet を選択します。作業域で Fixlet が開きます。
3. Fixlet の詳細を確認するには、ウィンドウの上部にあるタブをクリックします。
4. 「**アクションの実行**」をクリックして、Fixlet を適用します。



注: パッケージ・ソースとなるリポジトリを使用している場合は、Fixlet アクションを実行する前に、「**カスタム・リポジトリ・サポートの有効化 - CentOS**」タスクを実行します。このタスクにより、YUM を使用してエンドポイント上の構成済みリポジトリから必要なダウンロードを実行するためのアクションが構成されます。カスタム・リポジトリ・サポートを有効にすると、Fixlet は Bigfix インフラストラクチャーによるメタデータおよびパッケージのダウンロードを停止し、必要なファイルを YUM でダウンロードできるようにします。

「アクション」ボックスで、以下の該当するリンクをクリックすることもできます。

- デプロイメント・プロセスを開始できます。
 - パッチを適用する前にテスト実行を適用できます。「**エンドポイントの依存解決 - 適用結果**」分析を表示して、依存関係が正常に解決したかどうか、およびインストールが成功するかどうかを判断します。
 - 特定の Fixlet の Red Hat セキュリティー情報を表示することができます。パッチ・ページを表示するには、「**こちらをクリックして、パッチ ページを表示**」アクションを選択します。
5. 「アクションの実行」ダイアログでは、追加のパラメーターを設定することができます。
- 「アクションの実行」ダイアログを使用したパラメーターの設定について詳しくは、『[BigFix コンソール・オペレーター・ガイド](#)』を参照してください。
6. 「**OK**」をクリックします。

RPM 適用ウィザードの使用

RPM 適用ウィザードを使用することで、RPM パッケージの適用が容易になります。



注: Linux RPM パッチ・サイトは非推奨となったため、RPM 適用ウィザードは削除されます。詳しくは <https://forum.bigfix.com/t/ibm-bigfix-patch-to-deprecate-and-end-support-for-linux-rpm-patching-site-on-september-29-2017/22405> を参照してください。

このウィザードは、依存関係を解決することではなく、単純な適用のケースのみを処理するためのものです。ナビゲーション・ツリーの「コンテンツ・フィルター」部で、「**OS ベンダー**」 > 「**CentOS Linux**」 > 「**RPM 適用ウィザード**」の順にクリックします。

図 9. ナビゲーション・ペイン内の RPM 適用ウィザード

ウィザードが開いたら、適用する RPM パッケージを入力します。

図 10. ソース RPM パッケージの指定

1 回限りのアクションを作成する場合はチェック・ボックスを選択し、再使用可能な Fixlet® を作成する場合は空白のままにします。完了したら、「**作成**」をクリックします。次のような進行状況画面が表示されます。

図 11. 進行状況画面 - アーカイブの作成

最初の「タスク」ウィンドウが開いたら、「**OK**」をクリックして「秘密鍵のパスワード」に値を入力します。次に、「アクション」ボックス内をクリックして適用を開始します。

図 12. 「タスク」ウィンドウ - 「アクション」ボックス

「アクションの実行」ダイアログが開いたら、ウィンドウ上部にある各タブを使用してこのアクションのパラメーターを設定します。「OK」をクリックし、「秘密鍵のパスワード」に値を入力します。ウィザードがこのアクションをクライアントに適用します。

置き換え

置き換えの概念について詳しくは、「Windows 以外での置き換え ((ページ))」を参照してください。

CentOS 8 向けパッチのサイトは、[BaseOS](#) と [AppStream](#) リポジトリのみをサポートしています。詳しくは、「[サポートされている CentOS リポジトリ](#)」を参照してください。

CentOS 8 リポジトリのメタデータには、最新の使用可能なパッケージにのみ関連する情報が含まれるようになったため、置き換えられるすべての Fixlet がエラー `No matching Package found` で失敗するはずです。

Fixlet は、置き換えられると、デフォルトでは無関係になります。

置き換えられる Fixlet ではクライアントの設定「`_BESClient_CentOS_EnableSupersededEval`」の値が削除され、置き換えられる Fixlet の関連度が **false**`<Relevance>False</Relevance>` に設定されます。最新の使用可能なパッチのみをデプロイする必要があります。

カスタム・サイトを使用して置き換えられる Fixlet をデプロイするには、関連するリポジトリ・メタデータおよびパッケージを先行してキャッシュに入れておく必要があります。



注: ベースラインの Fixlet を置き換えるとエラーが発生してインストールが失敗するため、インストールを続ける前にすべてのベースラインを同期することをお勧めします。ベースラインとともにデプロイできるのは、最新の使用可能なパッケージのみです。

第 5 章. 複数パッケージのベースラインのインストール

BigFix パッチには、ベースラインに含まれる複数パッケージの更新のインストールを、単一のタスクに結合するためのソリューションが用意されており、それによってベースラインの実行時間を短縮できます。

ベースラインを使用すると、複数の Fixlet をグループにまとめて、任意の組み合わせのターゲット・コンピューターにすぐに適用することが可能になります。これは、特定のアクション・グループをネットワーク全体に適用するための強力な方法です。ただし、ベースライン内の各 Fixlet は、ベースラインの実行時に、個別の YUM 更新トランザクションを作成します。単一のベースラインで膨大な数の YUM 呼び出しが行われる可能性があり、その場合はすべてのトランザクションが完了するまでに時間がかかるため、パフォーマンスに深刻な影響が及ぶことがあります。

「複数パッケージのベースラインのインストール (multiple-package baseline installation)」ソリューションを使用すると、依存関係の解決やパッケージのインストールが Fixlet ごとに個別に行われることに伴うパフォーマンスの低下に対処できます。このソリューションでは、ベースラインの先頭でこの機能を有効化して、関連パッケージを単一の YUM 呼び出しからインストールするためのインストール・タスクを付加する必要があります。

パッケージをインストールするのではなく、リストに追加するように Fixlet に指示するフラグを設定するには、「**複数パッケージのベースラインのインストール機能を有効化 - CentOS 6**」タスク、「**複数パッケージのベースラインのインストール機能を有効化 - CentOS 7**」タスク、または「**複数パッケージのベースラインのインストール機能を有効化 - CentOS 8**」タスクを使用します。このフラグは、ベースラインの完了後にクリアされます。複数のパッケージを単一のコマンドからインストールできるようにするには、該当するタスクをベースラインの先頭に追加する必要があります。



注:

- CentOS Vault に保管されているパッケージが含まれる Fixlet (Fixlet タイトルにラベル ((`vault x.y`)) で示される) は、複数パッケージのベースラインのインストール機能を使用して適用することはできません。このような Fixlet は、個別に適用するか、または通常のベースラインで適用する必要があります。
- 複数パッケージのベースラインのインストール機能では、「**アクションの実行**」の事前キャッシュ・オプション「**すべての制約を満たす前のダウンロードの開始 (Start download before all constraints are met)**」はサポートされません。

複数パッケージのインストール・タスクは、それぞれの CentOS オペレーティング・システムのバージョンおよびアーキテクチャーに対して有効化されます。依存関係の解決を行い、パッケージをダウンロードして、それらをエンドポイントにインストールするには、該当するインストール・タスクをベースラインの最後に追加する必要があります。



重要: 「複数パッケージのベースラインのインストール機能を有効化」タスクおよび「複数パッケージのベースラインのインストール」タスクは、同じベースライン内に存在する必要があります。

「CentOS6 プラグイン R2 向けパッチ (Patches for CentOS6 Plugin R2)」サイトから使用可能:

- 複数パッケージのベースラインのインストール - CentOS 6 - x86 (Multiple-Package Baseline Installation - CentOS 6 - x86)
- 複数パッケージのベースラインのインストール - CentOS 6 - x86_64 (Multiple-Package Baseline Installation - CentOS 6 - x86_64)

「CentOS7 プラグイン R2 向けパッチ (Patches for CentOS7 Plugin R2)」サイトから使用可能なタスク:

- 複数パッケージのベースラインのインストール - CentOS 7 - x86_64 (Multiple-Package Baseline Installation - CentOS 7 - x86_64)

CentOS8 サイトのパッチから使用可能:

複数パッケージのベースラインのインストール - CentOS 8 - x86_64 (Multiple-Package Baseline Installation - CentOS 8 - x86_64)

ベースライン全体の依存関係の解決とパッケージのインストールを単一のインスタンスで実行するには、これらのタスクがベースラインの最後に実行される必要があります。

インストールの予行演習を行うことで、パッケージに対する変更をプレビューして、意図しないパッケージの更新によって発生する可能性のある依存関係の破損を防ぐこともできます。テスト・アクションは、[/var/opt/BESClient/EDRDeployData](#) にある以下のファイルに結果を出力します。

PkgToInstallList.txt file

このファイルには、依存関係チェック後にインストールされるパッケージが記載されます。

PkgToRemoveList.txt file

このファイルには、ターゲット CentOS エンドポイントから削除されるパッケージが記載されます。

BigFix パッチには、インストールを促進するために、以下のコンテンツも用意されています。

複数パッケージのベースラインのインストール用 CentOS 6 パッケージ・リスト・ファイルを削除 (Delete CentOS 6 Package List File for Multiple-Package Baseline Installation)

ターゲットの CentOS Linux 6 コンピューターからパッケージ・リスト・ファイルを削除します。

パッケージの依存関係の解決に関連して発生する可能性のある問題を防ぐために、このタスクをベースラインの開始時に追加します。

複数パッケージのベースラインのインストール用 CentOS 7 パッケージ・リスト・ファイルを削除 (Delete CentOS 7 Package List File for Multiple-Package Baseline Installation)

ターゲットの CentOS Linux 7 コンピューターからパッケージ・リスト・ファイルを削除します。

パッケージの依存関係の解決に関連して発生する可能性のある問題を防ぐために、このタスクをベースラインの開始時に追加します。

複数パッケージのベースラインのインストール用 CentOS 8 パッケージ・リスト・ファイルを削除 (Delete CentOS 8 Package List File for Multiple-Package Baseline Installation)

ターゲットの CentOS Linux 8 コンピューターからパッケージ・リスト・ファイルを削除します。

パッケージの依存関係の解決に関連して発生する可能性のある問題を防ぐために、このタスクをベースラインの開始時に追加します。

複数パッケージのインストール機能の使用について詳しくは、[ベースラインでの複数パッケージのインストール \(\(ページ\) 44\)](#)を参照してください。

ベースラインでの複数パッケージのインストール

複数パッケージのベースラインのインストール機能を使用すると、複数の固有パッケージを含む Fixlet を 1 つのベースラインから適用するときに、時間を短縮することができます。

- クライアント・ダウンロード事前キャッシュ・サイズを 2GB 以上に設定します。_BESClient_Download_PreCacheStageDiskLimitMB 設定を使用して、事前キャッシュの推奨サイズを設定してください。
- クライアント・ダウンロード・キャッシュ・サイズを 2GB 以上に設定します。_BESClient_Download_DownloadsCacheLimitMB 設定を使用して、キャッシュの推奨サイズを設定してください。
- 各エンドポイントの空きディスク・スペースが 4GB 以上あることを確認します。
- エンドポイントに GPG キーをインストールし、有効化します。Fixlet サイトから入手可能な「**RPM-GPG-KEY-centos-release のインポート (Import RPM-GPG-KEY-centos-release)**」タスク (ID# 301) を使用すると、必要なキーをインポートできます。

単一の YUM 呼び出しを実行することによってベースラインのすべての Fixlet のパッケージをインストールまたは更新するには、その機能を有効化するタスクと、適切な複数パッケージのベースラインのインストール・タスクを、ベースラインに追加する必要があります。使用可能なタスクのリストについては、[複数パッケージのベースラインのインストール \(\(ページ\) 42\)](#)を参照してください。



注::

- CentOS 8 の場合、ベースラインの Fixlet を置き換えるとエラーが発生してインストールが失敗するため、インストールを続行する前にすべてのベースラインを同期する必要があります。最新の使用可能なパッケージのみがベースラインとともにデプロイされます。
- 依存関係が破損したパッケージは、可能な限りスキップされます。ただし、パッケージをスキップできず、インストールがキャンセルされる場合があります。既知のケースを以下にいくつか示します。
 - CentOS 製品の依存関係に関する問題が含まれるパッケージ。
 - 依存関係エラーのあるパッケージはインストール中に発生します。これは通常、以下のエラー・メッセージによって示されます。File conflicts happen when two packages attempt to install files with the same name but different contents。



- タイトルに (vault x.y) が含まれる Fixlet は、複数パッケージのインストールのベースライン内では動作しません。
- 複数パッケージのベースラインのインストール機能では、「**アクションの実行**」の事前キャッシュ・オプション「**すべての制約を満たす前のダウンロードの開始 (Start download before all constraints are met)**」はサポートされません。



重要: この機能が動作するためには、「**複数パッケージのベースラインのインストール機能を有効化**」タスクおよび「**複数パッケージのベースラインのインストール**」タスクを、同じベースライン内に追加する必要があります。さらに、タスクと Fixlet の順序がベースライン・インストールにとって重要であるため、手順を正しい順序で実行するように注意してください。

1. ベースラインを作成します。

Fixlet サイトの Fixlet を強調表示して、コンテキスト・メニューから「**新しいベースラインに追加**」を選択します。「**ツール**」メニューから「**ベースラインの新規作成**」を選択することもできます。

2. オプション: いずれかの Fixlet を追加する前に、「**複数パッケージのベースラインのインストール用 CentOS 6 パッケージ・リスト・ファイルを削除**」タスク、「**複数パッケージのベースラインのインストール用 CentOS 7 パッケージ・リスト・ファイルを削除**」タスク、または「**複数パッケージのベースラインのインストール用 CentOS 8 パッケージ・リスト・ファイルを削除**」タスクを追加します。

「このコンポーネントが該当する場合はベースラインも適用可能なコンピューターに該当する (Baseline will be relevant on applicable computers where this component is relevant)」オプションが選択されていないことを確認します。

図 13. ベースライン・コンポーネントのオプション

3. 該当する「**複数パッケージのベースラインのインストール機能を有効化 (Enable the Multiple-Package Baseline Installation feature)**」タスクを追加します。

「このコンポーネントが該当する場合はベースラインも適用可能なコンピューターに該当する (Baseline will be relevant on applicable computers where this component is relevant)」オプションが選択されていないことを確認します。

4. 目的のパッチ Fixlet をベースラインに追加します。

すべての Fixlet について、「**このコンポーネントが関連する適用可能なコンピューター上では、ベースラインが関連します**」オプションが選択されていることを確認します。



注: 同一パッケージの異なるバージョンに影響する複数の Fixlet をベースラインに追加すると、インストール・タスクは古い方のバージョンのパッケージをスキップし、最新バージョンのみインストールします。

5. 該当する「**複数パッケージのベースラインのインストール (Multiple-Package Baseline Installation)**」タスクを、ベースラインの最後に追加します。このタスクにより、以下のいずれかのアクションを適用できます。
- 発生する可能性のある問題を確認するために、パッケージを実際にインストールすることなくインストールのプレビューを実行する。
 - すべての RPM を単一の YUM トランザクションでインストールする。
- 「このコンポーネントが該当する場合はベースラインも適用可能なコンピューターに該当する (Baseline will be relevant on applicable computers where this component is relevant)」オプションが選択されていないことを確認します。

ベースラインを実行する前に、以下の要件を満たしておく必要があります。

- エンドポイントで登録されているリポジトリに、対象パッケージと必要なすべての依存パッケージが格納されている必要があります。
- 同じ Fixlet を個別に適用する前に、複数パッケージのインストール方式を使用している Fixlet が、すべての YUM トランザクションを完了し、エンドポイントのステータスを更新するための十分な時間を空けてください。
- 同一エンドポイントに対して、同一サイトから複数のベースラインを実行しないでください。
- 技術情報に記載されたベースラインのベスト・プラクティスに従ってください。 https://hclpnpsupport.service-now.com/csm?id=kb_article&sys_id=d288c2021b098c9477761fc58d4bcbdf



注: ベースラインを適用すると、すべてのパッチ Fixlet の最初のサブアクション・ステータスに、Fixlet が失敗したことが示されます。これは予想された動作です。ベースラインでパッケージをダウンロードおよびインストールするプロセスは、Fixlet アクション・レベルで実行されるのではなく、「**複数パッケージのベースラインのインストール (Multiple-Package Baseline Installation)**」タスクで実行されます。ベースラインが完了すると、Fixlet のベースライン・サブアクション・ステータスに、各パッチ・インストールの最終状態が反映されます。

第 6 章. カスタム・リポジトリの管理

CentOS Linux バージョン 5 以降向けのパッチを管理するために、カスタム・リポジトリをセットアップできます。このソリューションでは、適用環境全体で複数のリポジトリが許可されます。

カスタム・リポジトリのサポートにより、CentOS ネイティブ・ツール・サイトの Fixlet で YUM を使用してカスタム・リポジトリから直接パッケージをダウンロードできます。mirror.centos.org および [Vault .centos.org](https://vault.centos.org) でパッケージ・ソースにアクセスする必要はありません。カスタム・リポジトリ・アーキテクチャーでは、帯域幅スロットリングはサポートされません。

カスタム・リポジトリを使用すると、ご使用の適用環境のエンドポイントに適用できる内容を柔軟に制御できます。たとえば、カスタム・リポジトリ内でホストしているカスタム・ソフトウェアを適用し、その後、「**パッチ・サポート**」サイトから「**YUM を使用したパッケージのインストール**」タスクを使用して、ソフトウェアをエンドポイントにインストールすることができます。詳しくは、「[カスタム・リポジトリからのパッケージのインストール \(ページ 51 \)](#)」を参照してください。

「**CentOS カスタム・リポジトリ管理 (CentOS Custom Repository Management)**」ダッシュボードを使用すると、カスタム・リポジトリ・ソリューションを簡単に統合できます。ただし、このダッシュボードでは、物理リポジトリは追加されません。このアクションは別個に行う必要があります。リポジトリの作成方法については、<https://wiki.centos.org/HowTos/CreateLocalRepos> を参照してください。

! **重要:** ダウンロード・エラーを回避するために、カスタム・リポジトリを使用して CentOS R2 ダウンロード・プラグインを登録する必要があります。「**ダウンロード・プラグインの管理 (Manage Download Plugins)**」ダッシュボードでダウンロード・プラグインを登録できます。

CentOS カスタム・リポジトリ管理ダッシュボード

「CentOS カスタム・リポジトリ管理」ダッシュボードを使用すると、既存のリポジトリ・ソリューションを BigFix パッチ管理ソリューションと簡単に統合できます。

「CentOS カスタム・リポジトリ管理」ダッシュボードを使用すると、Fixlet は、標準の BigFix ダウンロード・インフラストラクチャーを使用する代わりに、YUM を使用してダウンロードを実行することができます。このダッシュボードでは、エンドポイントにパッケージをインストールする際に、リポジトリを登録して YUM コマンドを使用することもできます。

このダッシュボードにアクセスするには、「**パッチ・サポート**」サイトをサブスクライブします。パッチ管理ドメインから、「**すべてのパッチの管理**」>「**ダッシュボード**」>「**CentOS カスタム・リポジトリ管理**」をクリックします。「**リポジトリ構成 - CentOS**」分析をアクティブにして、ダッシュボード上でコンテンツを表示します。

! **重要:** このダッシュボードを使用する前に、必要なメタデータとヘッダーを使用してカスタム・リポジトリを事前構成する必要があります。

「CentOS カスタム・リポジトリ管理」ダッシュボードを使用して、パッチ管理用の以下のアクションを実行します。

- リポジトリへのエンドポイントの登録と登録解除
- リポジトリ・ダッシュボード・リストでのローカル・リポジトリの追加、削除、およびインポート



注: 「CentOS カスタム・リポジトリ管理」ダッシュボードでは、物理的なリポジトリ・サーバーの作成はサポートされません。このリポジトリは、個別に作成する必要があります。リポジトリの作成方法については、以下のリソースを参照してください。 <https://wiki.centos.org/HowTos/CreateLocalRepos>



重要: ダウンロード・エラーを回避するために、カスタム・リポジトリを使用して CentOS R2 ダウンロード・プラグインを登録する必要があります。「ダウンロード・プラグインの管理 (Manage Download Plugins)」ダッシュボードでダウンロード・プラグインを登録できます。

リポジトリの追加

リポジトリを登録してエンドポイントに接続できるようにするために、「CentOS カスタム・リポジトリ管理」ダッシュボードを使用して、リポジトリをリポジトリ・リストに追加します。

「**リポジトリ構成 - CentOS**」分析をアクティブ化して、ダッシュボードにエンドポイントおよびリポジトリの情報を取り込みます。

1. 「**CentOS カスタム・リポジトリ管理 (CentOS Custom Repository Management)**」ダッシュボードで、「**リポジトリ**」タブをクリックします。
2. 「**追加**」をクリックします。
3. 「**新規リポジトリの追加 (Add a New Repository)**」ダイアログで、以下のフィールドに値を入力します。
 - **リポジトリ名**
 - **リポジトリ URL**



注: リポジトリ設定がリポジトリ・サーバー構成と一致することを確認する。

4. 「**保存**」をクリックします。

追加したリポジトリをエンドポイントに接続するには、[リポジトリへの CentOS エンドポイントの登録 \(ページ 48\)](#)を参照してください。

エンドポイントの既知のすべての既存リポジトリをこのダッシュボードのリポジトリ・リストに追加する場合は、インポート機能を使用します。詳しくは、[リポジトリのインポート \(ページ 50\)](#)を参照してください。

リポジトリへの CentOS エンドポイントの登録

「CentOS カスタム・リポジトリ管理」ダッシュボードを使用すると、エンドポイントにパッチを適用するときのパッケージ・ソースとしてリポジトリを追加できます。

- リポジトリ設定がリポジトリ・サーバー構成と一致することを確認する。
- 「**リポジトリ構成 - CentOS**」分析をまだアクティブ化していない場合はアクティブ化する。

1. 「CentOS カスタム・リポジトリ管理 (CentOS Custom Repository Management)」ダッシュボードで「エンドポイント」タブをクリックします。
2. 最初のテーブルから、リポジトリに登録するエンドポイントを選択します。選択したエンドポイントのリポジトリが 2 番目のテーブルにリストされます。



注: 名前が「未指定」になっているリポジトリは、ダッシュボードのリポジトリ・リストにリストされていません。

3. 「新規リポジトリの登録 (Register a new repository)」をクリックします。
4. 「新規リポジトリの登録 (Register a New Repository)」ダイアログで、リポジトリを選択して、「次へ」をクリックします。
後続のウィンドウに、エンドポイントの登録先リポジトリの名前と URL が表示されます。
5. オプション: 「追加フィールド (Additional Fields)」に詳細な構成情報を追加できます。
ベンダー・サイトの単なるミラーではないカスタム・リポジトリを持つユーザーは、「追加フィールド (Additional Fields)」に `gpgcheck=0` を追加する必要があります。gpg シグニチャー・ファイルが除外されると、rpm ファイルに対して認証チェックが行われず、インストールが失敗する可能性があります。
6. 「保存」をクリックします。この情報は、YUM 構成ファイルに保存されます。
7. 「アクションの実行」ダイアログで、コンピューターを選択し、「OK」をクリックしてアクションを適用します。

選択されたエンドポイントのパッケージ・ソースとしてリポジトリが正常に割り当てられました。

パッチ適用時にこのリポジトリを使用するには、「カスタム・リポジトリ・サポートの有効化 - CentOS」タスクを実行します。

リポジトリからの CentOS エンドポイントの登録解除

「CentOS カスタム・リポジトリ管理」ダッシュボードを使用して、関連性がなくなったリポジトリからエンドポイントを登録解除します。

リポジトリを登録解除すると、YUM 構成ファイルは削除されず、無効化のみが行われます。また、「CentOS カスタム・リポジトリ管理」ダッシュボードは、選択したコンピューターからシステム ID ファイルを削除します。

1. 「CentOS カスタム・リポジトリ管理 (CentOS Custom Repository Management)」ダッシュボードで「エンドポイント」タブをクリックします。
2. リポジトリを登録解除するエンドポイントを選択します。
3. 「新規リポジトリの登録解除 (Unregister a new repository)」をクリックします。
4. 「新規リポジトリの登録解除 (Unregister a New Repository)」ダイアログで、リポジトリを選択して「保存」をクリックします。
5. 「アクションの実行」ダイアログで、コンピューターを選択し、「OK」をクリックしてアクションを適用します。

リポジトリの削除

ダッシュボードのリポジトリ・リストの管理を容易にするために、適用環境にもはや存在していないリポジトリを削除します。

1. 「**CentOS カスタム・リポジトリ管理 (CentOS Custom Repository Management)**」ダッシュボードで、「**リポジトリ**」タブをクリックします。
2. 削除するリポジトリを選択し、「**削除**」をクリックします。削除を確認するダイアログが表示されます。
3. 「**はい**」をクリックして確認し、選択したリポジトリの削除を進めます。

選択したリポジトリがリストから削除されます。

リポジトリのインポート

「CentOS カスタム・リポジトリ管理」ダッシュボードのインポート機能を使用して、エンドポイントの既知のすべての既存リポジトリをこのダッシュボードのリポジトリ・リストに追加します。

「**リポジトリ構成 - CentOS**」分析をアクティブ化して、ダッシュボードにエンドポイントおよびリポジトリの情報を取り込みます。

既存のリポジトリをインポートする場合は、`.repo` ファイルで以下のエントリーが以下の順序で入力されているようにしてください。

```
name=  
baseurl=  
enabled=  
gpgcheck=
```

1. 「**CentOS カスタム・リポジトリ管理 (CentOS Custom Repository Management)**」ダッシュボードで、「**リポジトリ**」タブをクリックします。
2. 「**インポート**」をクリックします。
3. 「**既存のリポジトリのインポート (Import Existing Repositories)**」ダイアログで、ダッシュボードのリポジトリ・リストに追加するリポジトリを選択します。
4. リポジトリの名前を入力します。
5. 「**保存**」をクリックします。

これでリポジトリがインポートされ、ダッシュボードのリポジトリ・リストに追加されました。

YUM のパッケージ更新の確認

「**YUM: 使用可能なパッケージ更新の確認**」タスクを使用して、CentOS Linux エンドポイントに、インストール用の適用可能な YUM パッケージ更新を識別して判別します。

「**YUM: 使用可能なパッケージ更新の確認**」タスクは、`yum check-update` コマンドを使用して、エンドポイントにインストールされているパッケージに使用可能な更新を判別します。これは、「**パッチ・サポート**」サイトで使用可能なタスクです。

このタスクを実行する前に、**YUM ログ分析** (ID #28) がアクティブ化されていることを確認します。エンドポイントに適用可能な YUM パッケージ更新の結果の詳細は、分析内の「**YUM check-update 出力 (YUM check-update output)**」という名前の新規の列にリストされます。

このタスクは適用環境内のリポジトリを使用するため、このタスクの実行時に YUM パッケージがリポジトリ内で使用可能であることを確認します。

カスタム・リポジトリからのパッケージのインストール

BigFix には、カスタム・リポジトリに登録されているバージョン 5 以降の CentOS Linux のエンドポイントで、パッケージのインストールと更新を簡単に実行できるタスクが用意されています。

- 「**パッチ・サポート**」サイトをサブスクライブして、「**YUM を使用したパッケージのインストール**」という名前のインストール・タスクにアクセスします。
- 「CentOS カスタム・リポジトリ管理」ダッシュボードで、カスタム・リポジトリを構成します。詳しくは、[CentOS カスタム・リポジトリ管理ダッシュボード \(\(ページ\) 47\)](#)を参照してください。
- 「ダウンロード・プラグインの管理 (Manage Download Plug-ins)」ダッシュボードで CentOS R2 ダウンロード・プラグインに登録します。ダウンロード・プラグインによって生成されたデータは使用されない場合がありますが、ダウンロード・エラーを回避するためにプラグインに登録することが重要です。
- 構成したリポジトリが最新の状態になっていて、必要なパッケージとメタデータが含まれていることを確認します。

エンドポイントでパッケージのインストールや更新を行うには、「**YUM を使用したパッケージのインストール**」タスクを使用します。

インストール用に選択したパッケージを指定するには、パッケージ名または Common Vulnerabilities and Exposures (CVE) ID 番号を使用します。

カスタム・リポジトリ内の有効な新しいバージョンを使用して、エンドポイントにインストールされているすべてのパッケージを更新することもできます。

使用可能な各アクションの YUM コマンドを以下に示します。

yum install <package_name1> <package_name2>

特定の名前のパッケージを更新またはインストールします。複数のパッケージの更新やインストールを行うことができます。パッケージ名を区切るために、スペースを使用してください。

yum update

エンドポイントのすべてのインストール済みパッケージを更新します。

yum patch --cve=<cve_number>

特定の CVE ID 番号を持つパッケージを更新します。CVE ID 番号が指定されておらず、単一の CVE 参照だけが許可されている場合、このコマンドは失敗します。

yum patch --date=<YYYY-MM-DD>

指定した日付までに発行されたすべてのパッチをインストールします。

`yum install` コマンドと `yum update` コマンドの場合のみ、コマンド・オプションが追加のフラグとしてサポートされます。詳しい使用方法については、yum の man ページを参照してください。

このタスクには、インストール用のパッケージをテストするためのアクションも用意されています。テストを実行するためにパッケージをエンドポイントにインストールする必要はありません。

1. 「パッチ管理」ドメインから、「**すべてのパッチの管理**」>「**Fixlet とタスク**」をクリックします。
2. 「**YUM を使用したパッケージのインストール**」タスクを選択して、エンドポイントにカスタム・パッケージをインストールします。
3. 「タスク」ペインで説明を確認し、「アクション」ボックスの指示に従ってアクションを適用します。
4. 選択したアクションに応じて必要な情報を入力し、「**OK**」をクリックします。



注: エンドポイントにインストールされているすべてのパッケージを更新するには、パッケージをインストールするためのアクションを選択します。ただし、パッケージ名は指定しないでください。

5. 「アクションの実行」ペインで、パッケージのインストール先となるエンドポイント、または更新するパッケージが存在するエンドポイントを選択します。
6. 「**OK**」をクリックします。

第7章. YUM トランザクション管理

「YUM トランザクション履歴」ダッシュボードを使用して、YUM トランザクション履歴を表示してトランザクションを管理します。このダッシュボードは、CentOS Linux バージョン 5、6、および 7 で動作します。



注: 「YUM トランザクション履歴」ダッシュボードは、CentOS 8 ではサポートされていません。

このダッシュボードには YUM トランザクション履歴が表示され、適用環境内でトランザクションをロールバックしたり、元に戻したり、やり直したりするのに使用できます。

ロールバック

ロールバック機能は、指定したトランザクションの時点までのトランザクションをすべて取り消します。

元に戻す

元に戻す機能は、選択したトランザクションのみを元に戻します。

やり直す

やり直し機能は最近のトランザクション・アクションを繰り返します。



注: アクティブ・カーネルをカーネル更新に対してロールバックすることはできません。

要件

「YUM トランザクション履歴」ダッシュボードを使用するには、以下の要件が満たされていることを確認してください。

- YUM バージョン 3.2.28 以降を使用している。YUM バージョン 3.2.28 から 3.2.29-22 では、ロールバック機能がサポートされません。



注: 選択したエンドポイントの YUM バージョンが 3.2.29-22 より前の場合、ダッシュボードでその YUM バージョンの隣に警告サインが表示され、ロールバック・アクションがサポートされないバージョンであることを示します。サポートされないバージョンの場合、すべてのトランザクションに対して「ロールバック」ボタンが無効になります。

- 「パッチ・サポート」サイトをサブスクライブする。
- YUM トランザクション履歴分析をアクティブ化する。

YUM トランザクション・アクション

「アクション」列は、ダッシュボードでの YUM トランザクション・アクションを示します。以下の表に、トランザクションごとのアクションの詳細を示します。

表 5. トランザクション・アクションの説明

アクション	略称	説明
ダウングレード	D	少なくとも 1 つのパッケージが以前のバージョンにダウングレードされました。
消去	E	少なくとも 1 つのパッケージが削除されました。
インストール	I	少なくとも 1 つの新しいパッケージがインストールされました。
廃止	O	少なくとも 1 つのパッケージが廃止としてマークを付けられました。
再インストール	R	少なくとも 1 つのパッケージが再インストールされました。
更新	U	少なくとも 1 つのパッケージが新しいバージョンに更新されました。

YUM 履歴について詳しくは、[Red Hat Product Documentation](#) サイトを参照してください。

YUM トランザクション分析

「YUM トランザクション履歴」ダッシュボードでは以下の分析を使用します。

YUM トランザクション履歴分析

BigFix は、「YUM トランザクション履歴」ダッシュボードで実行されるアクションの結果を記録するログを生成します。YUM 履歴トランザクション分析は、アクション・ログ `yum_history.log` の内容を取得します。ログは、`/var/opt/BESClient/EDRDeployData/yum_history.log` にあります。

:NONE.

YUM ログ分析

YUM ログは、YUM がデフォルトで `/var/log/yum.log` に生成する正式なログです。デフォルトの場所を変更するには、`/etc/yum.conf` のログ・ファイルの設定を変更します。YUM ログ分析は、トラブルシューティングの目的に非常に有効です。

分析には 2 つのプロパティがあります。

YUM ログ・プロパティ

実行された操作をすべてログに記録し、変更されたトランザクションを識別します。このログは、YUM ログ・ファイルの最後の 40 行を取得します。

YUM 履歴ダッシュボードのアクション・ログ

このログは、アクションからのやり直しの操作、元に戻す操作、およびロールバックの操作のアクション・ログをリストします。アクションは、`/var/opt/BESClient/EDRDeployData/yum_history.log` にログを書き込みます。YUM 履歴ダッシュボードのアクションは、5 件の最新アクションを記録します。

YUM トランザクションのロールバック

「YUM トランザクション履歴」ダッシュボードを使用して、指定したトランザクションの時点までのトランザクションをすべてロールバックします。

以下の要件を満たしていることを確認してください。

- YUM バージョン 3.2.28 以降を使用している。YUM バージョン 3.2.28 から 3.2.29-22 では、ロールバック機能がサポートされません。



注: 選択したエンドポイントの YUM バージョンが 3.2.29-22 より前の場合、ダッシュボードでその YUM バージョンの隣に警告サインが表示され、ロールバック・アクションがサポートされないバージョンであることを示します。サポートされないバージョンの場合、すべてのトランザクションに対して「**ロールバック**」ボタンが無効になります。

- 「パッチ・サポート」サイトをサブスクライブする。
- YUM トランザクション履歴分析をアクティブ化する。



注: アクティブ・カーネルをカーネル更新に対してロールバックすることはできません。

1. BigFix コンソールで、「外部サイト」 > 「パッチ・サポート」 > 「ダッシュボード」 > 「YUM トランザクション履歴」に移動します。
2. YUM 履歴を表示するエンドポイントを選択します。



注: 選択したエンドポイントの YUM バージョンが 3.2.29-22 より前の場合、そのバージョンはサポートされていないことがツールチップで示されます。

3. ロールバックするトランザクションを選択します。
4. 「**ロールバック**」をクリックします。
「トランザクションまでロールバック」ウィンドウが開きます。
5. **オプション:** 追加のフラグを入力して、ロールバック・アクションをさらにカスタマイズできます。
6. 「**適用**」をクリックします。
7. 「**アクションの実行**」ウィンドウで、コンピューターを選択し、「**OK**」をクリックしてアクションを実行します。

YUM トランザクションの取り消し

「YUM トランザクション履歴」ダッシュボードを使用して、特定の単一のトランザクションまで戻します。

以下の要件を満たしていることを確認してください。

- YUM バージョン 3.2.28 以降を使用している。



注: 選択したエンドポイントの YUM バージョンが 3.2.29-22 より前の場合、ダッシュボードでその YUM バージョンの隣に警告サインが表示され、ロールバック・アクションがサポートされないバージョンであることを示します。サポートされていないバージョンの場合、すべてのトランザクションに対して「**元に戻す**」ボタンが無効になります。

- 「パッチ・サポート」サイトをサブスクライブする。
- YUM トランザクション履歴分析をアクティブ化する。



注: アクティブ・カーネルをカーネル更新に対してロールバックすることはできません。

1. BigFix コンソールで、「外部サイト」 > 「パッチ・サポート」 > 「ダッシュボード」 > 「YUM トランザクション履歴」に移動します。
2. YUM 履歴を表示するエンドポイントを選択します。
3. 元に戻すロールバックがあるトランザクションを選択します。
4. 「元に戻す」をクリックします。
5. 「トランザクションの取り消し」ウィンドウで「適用」をクリックします。
6. 「アクションの実行」ウィンドウで、コンピューターを選択し、「OK」をクリックしてアクションを実行します。

YUM トランザクションのやり直し

「YUM トランザクション履歴」ダッシュボードを使用して最近のトランザクション・アクションを繰り返します。

以下の要件を満たしていることを確認してください。

- YUM バージョン 3.2.28 以降を使用している。



注: 選択したエンドポイントの YUM バージョンが 3.2.29-22 より前の場合、ダッシュボードでその YUM バージョンの隣に警告サインが表示され、ロールバック・アクションがサポートされないバージョンであることを示します。サポートされていないバージョンの場合、すべてのトランザクションに対して「**やり直し**」ボタンが無効になります。

- 「パッチ・サポート」サイトをサブスクライブする。
- YUM トランザクション履歴分析をアクティブ化する。



注: アクティブ・カーネルをカーネル更新に対してロールバックすることはできません。

1. BigFix コンソールを使用して、「外部サイト」 > 「パッチ・サポート」 > 「ダッシュボード」 > 「YUM トランザクション履歴」に移動します。
2. YUM 履歴を表示するエンドポイントを選択します。

3. やり直すトランザクションを選択します。
4. 「やり直し」をクリックします。
5. 「トランザクションのやり直し (Redo Transaction)」ウィンドウで「適用」をクリックします。
6. 「アクションの実行」ウィンドウで、コンピューターを選択し、「OK」をクリックしてアクションを実行します。

付録 A. サポート

この製品について詳しくは、以下のリソースを参照してください。

- [BigFix サポート・ポータル](#)
- [BigFix Developer](#)
- [YouTube の BigFix プレイリスト](#)
- [YouTube の BigFix Tech Advisors チャンネル](#)
- [BigFix フォーラム](#)

付録 B. トラブルシューティング

CentOS エンドポイントへのパッチ適用時に問題が発生した場合は、ログ・ファイルを確認して、発生した問題およびエラーの修正方法を判断します。

ログ・ファイル

トラブルシューティングの精度を上げるために、エラー報告とエラー処理をより明確に示すことで、ロギングを強化しています。

CentOSPluginR2.log

CentOS ダウンロード・プラグイン R2 の実行に関連するダウンロードの結果がリストされています。情報量はロギング・レベルによって異なります。

このログの場所は以下のとおりです。

- Windows システムの場合: `%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins\CentOSR2Protocol`
- Linux システムの場合: `/var/opt/BESServer/DownloadPlugins/CentOSR2Protocol`

以下のログ・ファイルが、ディレクトリー `/var/opt/BESClient/EDRDeployData` のクライアント・フォルダーにあります。

EDR_DeploymentResults.txt

EDR デプロイメントの結果と YUM の出力がリストされています。

register-repo.log

「CentOS カスタム・リポジトリ管理」ダッシュボードのリポジトリ登録アクションの実行結果が記録されます。

unregister-repo.log

「CentOS カスタム・リポジトリ管理」ダッシュボードのリポジトリ登録解除アクションの実行結果が記録されます。

yum_history.log

「YUM トランザクション履歴」ダッシュボードからの実行結果が記録されます。

その他の有用なログ・ファイル:

yum.log

これは YUM がデフォルトで `/var/log/yum.log` に生成する正式なログです。YUM 関連のすべての操作およびトランザクションが記録されます。

ダウンロード・プラグインのロギング・レベル

ロギング・レベルは、CentOS ダウンロード・プラグイン R2 がログ・ファイルに書き込む詳細情報の量を決定します。`%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins\CentOSR2Protocol\plugin.ini` ファイルでロギング・レベルを設定します。



注: ロギング・レベルの値では、大/小文字が区別されます。

以下のロギング・レベルは、ログに記録される情報量が少ない順にリストされています。

ERROR

ダウンロード・プラグインの実行に関連するエラーが出力されます。このようなエラーは、致命的エラーが発生する直前であることを示している場合があります。

WARNING

ダウンロードの失敗と、失敗の理由に関する情報が出力されます。

INFO

ダウンロードの進行状況とダウンロードの成功に関する一般情報が、最小限のトレース情報とともに出力されます。

DEBUG

問題のトラブルシューティングに使用される詳細情報が出力されます。これは、使用可能なレベルの中で最も詳細なレベルです。



注: ロギング・レベルを DEBUG に設定すると、ログに記録する情報の量が増えるため、パフォーマンスに影響が及ぶ可能性があります。ロギング・レベルを DEBUG に上げるのは、問題を調査するときだけに与える必要があります。

クライアントのデバッグ・ログを有効にする

「パッチ・サポート」サイトから、「Linux パッチ適用のデバッグ・ログを有効化するようにクライアント設定を構成する (Configure the client setting to enable the Debug Log for Linux Patching)」Fixlet (ID #57) を使用します。

プリフェッチ・プラグインのエラー

アクション・スクリプトの `execute prefetch plugin-in` が含まれる行で失敗した Fixlet でアクションを実行した場合、同じプリフェッチ・プラグインに対するすべてのアクション・スクリプトからのそれ以降の呼び出しが、そのエンドポイントで失敗する可能性があります。スクリプトがブラックリストに登録された可能性があり、プリフェッチ・プラグインがエラーになります。

確認するには、クライアント・ログを調べます。プリフェッチ・プラグインを実行する Fixlet アクションに対して、以下のメッセージのいずれかが見つかります。

```
execute prefetch plug-in' didn't complete within 300 seconds. Black listing plug-ins
matching the sha1 hash of 'name of 'bash' until agent is restarted.
```

```
Execute prefetch plug-in attempting to reuse plug-in which took too long earlier.
```

この問題を解決するには、以下のアクションを実行します。

1. BigFix クライアントを再始動して、ブラックリストをクリアします。
2. `_BESClient_ActionManager_PrefetchPlugInTimeoutSeconds` クライアント構成設定に、パッチで依存関係をインストールして解決するための十分な時間を設定します。このクライアント設定は、クライアントがスクリプトをブラックリストに登録する前に待機する時間を示します。パッチ・サポート・サイトで使用できる「**プリフェッチ・プラグインのタイムアウトの変更 (Change Timeout for Prefetch Plugins)**」タスクを使用して、この設定を 30 分 (1800 秒) に設定できます。



注: `_BESClient_ActionManager_PrefetchPlugInTimeoutSeconds` 設定は、エンドポイントおよびインストール中の Fixlet によって異なります。最適な値を求めるには、最も遅いエンドポイントで設定を 3,000 秒などの高い値に設定し、大きな Fixlet を実行して所要時間を確認します。その時間に 2 を乗算した値を使用できます。あるいは、推奨値ではうまくいかない場合は、クライアント設定を 600 秒に設定し、適宜調整していきます。

/var を noexec としてマウントした場合のエラー

使用可能なすべての Fixlet は、デフォルトでは、エンドポイント上のパーティションである `/var` ディレクトリーから直接実行される実行可能ファイルを使用します。`/var` が `noexec` オプションを使用して設定されている場合、CentOS ダウンロード・プラグイン R2 またはカスタム・リポジトリ・ソリューションのいずれを使用しているかにかかわらず、Fixlet は動作しません。そのため、以下の手順を実行して、`/var` ディレクトリーが `noexec` オプションを使用して設定されないようにする必要があります。

1. クライアント・ログを確認して、プリフェッチ・プラグインから `exit code 126` が返されているかどうかを調べます。例:
2. root ユーザーとして `mount` を実行し、現在使用されているマウント・オプションを確認します。

```
[root@host ~]# mount
/dev/mapper/vg_data-lv_root on / type ext4 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
tmpfs on /dev/shm type tmpfs (rw)
/dev/sda1 on /boot type ext4 (rw,nodev)
/dev/mapper/vg_data-lv_var on /var type ext4 (rw,noexec,nosuid,nodev)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
```

`/var` が `noexec` に設定されている場合は、以下のいずれかのアクションを実行する必要があります。

- noexec マウント・オプションを解除する。
- `/var/opt/BESClient` を noexec でない別のパーティションに移動し、そこへのシンボリック・リンクを元の場所に作成する。
- 「`_BESClient_LinuxPatch_executable_directory` のパスの設定 (Set the path for `_BESClient_LinuxPatch_executable_directory`)」 Fixlet を実行して、パッチ適用の実行可能ファイルを実行するための代替ディレクトリーを指定します。ディレクトリー・パスは、有効な絶対パス名でなければなりません。使用できるのは英数字、スラッシュ、および下線のみです。

GPG キーがない

適用時の問題を回避するためには、パッチ Fixlet を適用する前に Fixlet サイトから入手可能な「**RPM-GPG-KEY-centos-release のインポート (Import RPM-GPG-KEY-centos-release)**」タスク (ID# 301) を適用します。

リポジトリ・メタデータが大きすぎる

パッチ適用で使用されるリポジトリ・メタデータは、ベンダーによって提供されるものであり、サイズが大きい場合があります。`/var` ディレクトリーにメタデータを保管するための十分なスペースがない場合は、そのエンドポイント上でメタデータを保管できるだけの十分なスペースを持つ代替ディレクトリーを設定します。「`_BESClient_LinuxPatch_metadata_directory` のパスの設定 (Set the path for `_BESClient_LinuxPatch_metadata_directory`)」を使用すると、リポジトリ・メタデータが作成されるディレクトリーを設定できます。

BigFix Patch ダウンロード・プラグインの構成時に Null エラーが発生する

Null エラーを回避するために、BigFix サーバーと BigFix サーバー上の BigFix クライアントは必ず同じバージョンにしてください。このエラーが発生するのは、BigFix サーバー 8.x バージョンと 9.x バージョンが異なる方法で暗号化を処理するためです。Bigfix サーバー上のクライアントのバージョンを使用して BigFix サーバーのバージョンが判別され、Bigfix サーバーと BigFix サーバー上のクライアントのバージョンが同じであると想定されます。

BigFix サーバーと BigFix サーバー上の BigFix クライアントのバージョンが一致することを確認して、ダウンロード・プラグインの構成時に Null エラーが発生しないようにしてください。少なくとも、バージョンは、同じメジャー・バージョン・レベル (たとえば、8.x または 9.x) でなければなりません。

付録 C. よくある質問

BigFix Patch for CentOS をよりよく理解するために、以下の質問と回答を確認してください。

「ダウンロード・プラグインの管理」ダッシュボードはデータを反映していません。どうすればよいでしょう。

この問題のトラブルシューティングのために実施できる手順を以下に示します。

- 最新の「パッチ・サポート」サイトを収集します。
- 「パッチ・サポート」サイトから使用できる「ダウンロード・プラグインのバージョン」分析をアクティブにします。
- BigFix コンソール・キャッシュをクリアします。

置き換えられるパッチとは何ですか？

置き換えられる Fixlet とは、古いパッケージが含まれた Fixlet のことです。Fixlet® が置き換えられると、新しいバージョンのパッケージが含まれた新規 Fixlet® が存在するようになります。新規 Fixlet® Fixlet の ID は、置き換えられた Fixlet® の説明で確認できます。

適用ログはエンドポイントのどこにありますか？

ログは、EDRDeployData にあるクライアント・フォルダー内の `/var/opt/BESClient/EDRDeployData` という名前のフォルダーにあります。

アクションがダウンロードの失敗として報告されるのはなぜですか？

ダウンロード・プラグインが最新バージョンに更新されており、正しい資格情報で登録されていることを確認してください。

アクションがインストール失敗のレポートを返した場合はどうすればよいですか？

競合がベンダー提供のパッケージによって発生したのかどうかを調べてください。インストールを実行するには、それらの競合を削除する必要があります。

`/var` ディレクトリーが `noexec` としてマウントされているエンドポイントの Fixlet にパッチを適用できません。どうすればよいでしょう。

回避策については、`/var` を `noexec` としてマウントした場合のエラー ([ページ 61](#)) を参照してください。

ダウンロード・プラグインが正しく登録されているかどうかは、どのように確認すればよいですか？

ダウンロード・プラグインが正しく登録されているかどうかを確認するには、アクション・タスクと共に、Fixlet を実行します。パッチのダウンロードが成功したことを確認します。成功していない場合には、ダウンロード・プラグインを登録解除してから再登録することが必要な場合があります。

ダウンロード・プラグインを登録する方法を教えてください。ダウンロード・プラグインの登録タスクまたは「ダウンロード・プラグインの管理」ダッシュボードのどちらを使用すればよいですか。

ダウンロード・プラグインを登録するには、「Patching Support」サイトの「ダウンロード・プラグインの管理」ダッシュボードを使用する必要があります。既存のダウンロード・プラグインの登録タスクは使用しないでください。プラグインの登録について詳しくは、『[CentOS ダウンロード・プラグイン R2 の登録 \(\[ページ 18\]\(#\) \)](#)』を参照してください。



注: ダウンロード・プラグインの登録解除、構成、およびアップグレードを行う際にも、「ダウンロード・プラグインの管理」ダッシュボードを使用する必要があります。既存のダウンロード・プラグインの登録解除および編集タスクは使用しないでください。ダッシュボードについて詳しくは、[BigFix Knowledge Center](#) の『「ダウンロード・プラグインの管理」ダッシュボード』のトピックを参照してください。

どのバージョンの YUM ネイティブ・ツールを使用する必要がありますか？

CentOS ネイティブ・ツール向けパッチ・サイトでは、バージョン 3.2.19-18 以降が必要です。

アクションが失敗し、ログには YUM 固有のエラーが表示されています。失敗したアクションはどのようにトラブルシューティングすればよいですか？

YUM および関連するエラーについて詳しくは、<http://yum.baseurl.org> の YUM の資料および Red Hat Customer Portal の YUM 関連の記事を参照してください。

CentOS カスタム・リポジトリ・ダッシュボードを使用するには、どのバージョンの YUM が必要ですか？

最小でも 3.2.19-18 の YUM バージョンが必要です。

CentOS カスタム・リポジトリ・ダッシュボードではどの CentOS Linux がサポートされますか？

このダッシュボードは、CentOS Linux バージョン 5、6 および 7 をサポートします。

CentOS カスタム・リポジトリ・ダッシュボードをサポートするのは、どのバージョンの BigFix サーバーですか？

CentOS カスタム・リポジトリ・ダッシュボードは、バージョン 9.0 以降の BigFix をサポートします。

パッチを適用するときには、既存の方法とカスタム・リポジトリを介する方法のどちらを使用すればよいですか？ この 2 つの方式は共存できますか？

この 2 つの方式は併存できます。ただし、単一クライアントのパッチを適用する場合は、ネイティブ・ツールとカスタム・リポジトリ方式のどちらを使用するのか選択する必要があります。

CentOS カスタム・リポジトリ・ダッシュボードを使用した場合、依存関係はどのようにして解決されますか？

YUM は、メタデータを使用して依存関係を解決し、必要なパッケージを認識します。

ソフトウェアのインストールでカスタム・リポジトリを使用できますか？

はい、ソフトウェアのインストールでカスタム・リポジトリを使用できます。ソフトウェアのインストールにカスタム・リポジトリを使用するには、以下のステップに従ってください。

1. クライアントがカスタム・リポジトリ・ダッシュボードを介して登録されていることを確認します。
2. アクション `yum install<space><package name>` を実行して、カスタム・サイトで Fixlet を作成します。適切な関連度または成功条件 (つまり Fixlet がそのクライアントまたはエンドポイントに対してアクションを実行すべきかどうか) を確実に設定します。Fixlet の作成について詳しくは、「コンソール・オペレーター・ガイド」を参照してください。



注: リポジトリが確実に更新されるようにすることは重要です。パッケージを使用できないと、アクションが失敗する可能性があります。

以前に構成したリポジトリを再び構成できますか？

はい、以前に構成したことのあるリポジトリを再び構成できます。

ログによって、リポジトリに対する通常の YUM プロセスを使用しているかどうか分かりますか？

はい、ログには、リポジトリ内の通常の YUM プロセスが使用されているかどうか記録されます。

リポジトリの登録とリポジトリのインポートの違いは何ですか？

ダッシュボードの「リポジトリ」リストに含まれていない既存のリポジトリがある場合は、インポート機能を使用します。「リポジトリ」リストにリポジトリがすでにある場合でも、そのリポジトリをエンドポイントにリンクする必要がある場合は、登録機能を使用します。

リポジトリにパッケージが含まれていないとどうなりますか？

パッケージが見つからない場合、Fixlet は失敗します。YUM 出力のログが記録される `EDR_DeploymentResult.txt` からトラブルシューティングを実行できます。

カスタム・リポジトリ・ソリューションで問題が発生するとどうなりますか？

カスタム・リポジトリの解決で問題が発生した場合、ユーザーは標準の BigFix サーバー・ソリューションに戻すことができます。ユーザーは、「カスタム・リポジトリ・サポートの無効化 - CentOS」と呼ばれるタスクを実行することができます。

「CentOS カスタム・リポジトリ管理」ダッシュボードの「エンドポイント」タブで、ウィンドウ下部にリストされているリポジトリは順番に使用されますか？

「エンドポイント」タブにリストされるリポジトリに順番はありません。YUM がリポジトリを照会すると、最初にフェッチ照会を受けたリポジトリが、パッケージとその依存関係を含めて応答を返します。

「CentOS カスタム・リポジトリ管理」ダッシュボードにより、製造元のサイトのミラーではないカスタム・リポジトリを介してパッチが適用されました。適用が失敗し、EDR ログに rpm ファイルを開けなかったことが記録されています。どうすればよいでしょう。

ベンダー・サイトのミラーではないカスタム・リポジトリが使用された場合は、インストールの一環としてデフォルトの gpgcheck が実行され、gpg シグニチャー・ファイルが含まれない可能性があります。rpm ファイルに対して認証チェックが行われず、インストールが失敗する可能性があります。この問題を解決するには、「CentOS カスタム・リポジトリ管理」ダッシュボードでエンドポイントを登録するときに、「追加フィールド (Additional Fields)」に「gpgcheck=0」を追加したことを確認してください。

「タスク: YUM を使用したパッケージのインストール (Task: Install packages by using YUM)」を使用して複数のパッケージをインストールできますか？

はい、このタスクで複数のパッケージをインストールできます。各 rpm 名をスペースで区切ってください。

「YUM トランザクション履歴」ダッシュボードがサポートする YUM のバージョンは何ですか？

最小でも 3.2.28 の YUM バージョンが必要です。

「YUM トランザクション履歴」ダッシュボードがサポートする CentOS Linux のバージョンは何ですか？

このダッシュボードは、CentOS Linux バージョン 5、6 および 7 をサポートします。

「YUM トランザクション履歴」ダッシュボードの「ロールバック」と「元に戻す」の違いは何ですか？

ロールバック・コマンドは、指定したトランザクションの時点までのトランザクションをすべて取り消します。元に戻すコマンドは、選択したトランザクションのみを元に戻します。

YUM トランザクション履歴ログと YUM ログ分析の違いは何ですか？

Patch Management for CentOS Linux は、「YUM トランザクション履歴」ダッシュボードで実行されるアクションの結果を記録する YUM トランザクション履歴ログを生成します。ログは、`/var/opt/BESClient/EDRDeployData/yum_history.log` にあります。

YUM ログは、YUM がデフォルトで `/var/log/yum.log` に生成する正式なログです。デフォルトの場所を変更するには、`/etc/yum.conf` のログ・ファイルの設定を変更します。

「YUM トランザクション履歴」ダッシュボードに表示されるのは `yum -update all` だけですか？

`yum -update all` だけでなく、ダッシュボードの「コマンド・ライン」列に、以下のインストール・コマンドなどのさまざまなトランザクションが表示されます。

- `install bzip2`
- `install net-tools`
- `install vim enhance`
- `install wget`

YUM コマンドについて詳しくは、Red Hat Enterprise Linux の Web サイト (<https://access.redhat.com/>) を参照してください。

BigFix クライアントが YUM リポジトリを使用するように構成されている場合、「YUM トランザクション履歴」ダッシュボードはどのように動作しますか？

BigFix クライアントが YUM リポジトリを使用するように構成している場合でも、「YUM トランザクション履歴」ダッシュボードが適用環境に悪影響を及ぼすことはありません。ローカルの YUM リポジトリを既に使用している場合は、ロールバック用のパッケージを指定の方が簡単な場合があります。

クライアント・ログに、Fixlet が正常に完了しないプリフェッチ・プラグイン・エラーが表示されています。エラーの原因は何でしょうか。どうすればよいでしょう。

エンドポイントで実行されていた ActionScript がブラックリストに登録され、プリフェッチ・プラグインの問題が発生した可能性があります。

この問題を解決するには、BigFix クライアントを再始動してブラックリストをクリアしてください。スクリプトがブラックリストに登録されないよう

に、`_BESClient_ActionManager_PrefetchPlugInTimeoutSeconds` クライアント構成設定に、パッチを処理するための十分な時間を設定してください。詳しくは、「[プリフェッチ・プラグインのエラー \(ページ 60\)](#)」を参照してください。

CentOS ダウンロード・プラグイン R2 のログ・レベルを設定することはできますか？

必要な情報のレベルに応じたログ・メッセージを生成するように、ダウンロード・プラグインを設定することができます。

ロギング・レベルは、CentOS プラグイン R2 がログ・ファイルに書き込む詳細情報の量を決定します。`%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins\CentOSR2Protocol\plugin.ini` ファイルでロギング・レベルを設定します。



注: ロギング・レベルの値では、大/小文字が区別されます。

以下のロギング・レベルは、ログに記録される情報量が少ない順にリストされています。

ERROR

ダウンロード・プラグインの実行に関連するエラーが出力されます。このようなエラーは、致命的エラーが発生する直前であることを示している場合があります。

WARNING

ダウンロードの失敗と、失敗の理由に関する情報が出力されます。

INFO

ダウンロードの進行状況とダウンロードの成功に関する一般情報が、最小限のトレース情報とともに出力されます。

DEBUG

問題のトラブルシューティングに使用される詳細情報が出力されます。これは、使用可能なレベルの中で最も詳細なレベルです。



注: ロギング・レベルを DEBUG に設定すると、ログに記録する情報の量が増えるため、パフォーマンスに影響が及ぶ可能性があります。ロギング・レベルを DEBUG に上げるのは、問題を調査するときだけにとどめる必要があります。

CentOS ダウンロード・プラグイン R2 のログはどこにありますか? 使用できるログ・レベルにはどのようなものがありますか?

ロギングは `plugin.ini` ファイルによって制御されます。このファイルは、ダウンロード・プラグインの実行可能プログラムと同じ場所にあります。デフォルトでは、Windows システム上の `%PROGRAM FILES%\BigFix Enterprise\BES Server\DownloadPlugins\CentOSR2Protocol` にあります。Linux システムの場合は、`/var/opt/BESServer/DownloadPlugins/CentOSR2Protocol` にあります。ログ・ファイルは 1 日ごとに循環して使用されます。つまり、新規ログ・ファイルが作成されると、古いログ・ファイルの名前がその作成開始日に変更されます。

新バージョンのダウンロード・プラグインがインストールされると、CentOS ダウンロード・プラグイン R2 の構成ファイル (plugin.ini) は上書きされますか？

いいえ。構成ファイルが上書きされることはありません。構成ファイルが上書きされるのは、ダウンロード・プラグインが再構成されたときだけです。

CentOS ダウンロード・プラグイン R2 の登録後に、プロキシを再構成することはできますか？

はい。「ダウンロード・プラグインの管理」ダッシュボードからダウンロード・プラグインを構成することで、プロキシ設定を更新できます。

DLCentOSRepoList.json ファイルを編集してリポジトリを追加するとどうなりますか？

BigFix が「パッチ・サポート (Patching Support)」サイトを更新すると、このファイルが上書きされるため、行った変更はこのときに削除されます。

拡張製品のサブスクリプションがあります。各製品に割り当てられたリポジトリにアクセスするように CentOS ダウンロード・プラグイン R2 を構成できますか？

はい、できます。詳しくは、「[CentOS ダウンロード・プラグイン R2 の拡張 \(ページ 24\)](#)」を参照してください。

拡張リポジトリ・リスト・ファイルはどこに保存すればよいですか？

ダウンロード・プラグインがアクセスできる場所であれば、どこに保管しても構いません。BigFix サーバーがその場所に対するアクセス権を保持していることを確認する必要があります。

CentOS プラグイン R2 にアップグレードした後、どのパッケージもインストールできなくなりました。すべてのタスクは、以下の行になります。「Failed add prefetch item {concatenation ";" of lines of file (parameter "EDR_PkgRequest")}」。何が問題なのでしょうか？

BigFix 拡張セキュリティー・オプション `-requireSHA256Downloads` または BigFix 管理ツールの「**SHA-256 ダウンロードが必要**」オプションが有効になっている可能性があります。このオプションは、すべてのダウンロード検証で SHA-256 アルゴリズムのみを使用するように構成します。リポジトリ内には、CentOS ダウンロード・プラグイン R2 が使用するパッケージの SHA-256 値が含まれていない、特定の CentOS リポジトリ・メタデータがあるため、このプラグインは失敗する可能性があります。

パッチを正常にデプロイするために、「**SHA-256 ダウンロードが必要**」オプションを無効にすることを検討してください。パッケージの GPG シグニチャーを使用して別の層の検査および検証が実行されるため、セキュリティーおよびパッケージの整合性が低下することはありません。ダウンロード・オプションについて詳しくは、BigFix Platform インストール・ガイド (https://help.hcl-software.com/bigfix/9.5/platform/Platform/Installation/c_security_settings.html) を参照してください。

拡張リポジトリ・リストだけを使用するように CentOS ダウンロード・プラグイン R2 を構成できますか？

はい、`onlyUseExtendedRepoListFile` の `plugin.ini` フラグを「yes」に設定します。

ベースライン内の複数の Fixlet を適用するときに、破損した依存関係をスキップして、残りのパッケージのインストールを続行することはできますか？

「**複数パッケージのベースラインのインストール (Multiple-Package Baseline Installation)**」タスクは、依存関係が破損したパッケージを可能な限りスキップします。ただし、CentOS の依存関係に関する問題が含まれるパッケージはスキップできません。パッケージがスキップされないもう 1 つのシナリオは、インストール中に依存関係エラーが発生した場合です。これは、次のエラー・メッセージで示されます。File conflicts happen when two packages attempt to install files with the same name but different contents。このような場合は、インストールが取り消され、エンドポイントにパッチがインストールされません。

複数パッケージのベースラインのインストール方式を使用したときに障害が発生する原因として考えられるものは何ですか？

Fixlet のインストールに失敗した理由は以下の可能性があります。

- カスタム・サイトに、「複数パッケージのベースラインのインストール (multiple-package baseline installation)」タスクを同時に実行するベースラインが複数存在していた。
- 複数の Fixlet で、同一パッケージの複数のバージョンを更新するよう要求された。
- 複数の Fixlet で、同一のパッケージ依存関係を更新するよう要求された。
- ベースラインが複数パッケージのインストール方式を実行した直後に Fixlet が適用された。複数パッケージのインストールで、すべての YUM トランザクションを完了し、エンドポイントのステータスを更新するための十分な時間が取られなかった。
- YUM ツールにパッチが適用されていないのに、ベースラインに CentOS リリース・パッケージが含まれている。
- 「**複数パッケージのベースラインのインストール機能を有効化 (Enable the Multiple-Package Baseline Installation feature)**」タスクは、残りのコンテンツと同じベースライン内にある必要があります。このタスクは、パッチ Fixlet および複数パッケージのインストール・タスクより前に追加する必要があります。
- 複数パッケージのベースラインのインストール機能は、有効化タスクとインストール・タスクの両方が同じベースライン内に存在する場合にのみ動作します。詳しくは、「[ベースラインでの複数パッケージのインストール \(ページ 44\)](#)」を参照してください。
- 以下のクリーンアップ・タスクのいずれかの後に、「**複数パッケージのベースラインのインストール機能を有効にする**」タスクを追加する必要があります。**複数パッケージのベースラインのインストール用に CentOS 6 パッケージ・リスト・ファイルを削除 (Delete CentOS 6 Package List File for Multiple-Package Baseline Installation)** または **トラブルシューティング: CentOS 7 パッチ・デプロイメント・ログ - クリーンアップ (TROUBLESHOOTING:CentOS 7 Patching Deployment Logs - Cleanup)**。
- 正しい CentOS ディストリビューション、オペレーティング・システム・バージョン、サービス・パック・レベル、およびアーキテクチャーを使用するインストール・タスクをベースラインの最後に追加する必要があります。

エンドポイントが隔離された環境にあります。これらのエンドポイントにパッチを適用するには、BigFix をどのように構成すればよいですか？

隔離された環境の場合は、エンドポイントにパッチを適用するために必要なパッケージをホストするサポート対象 CentOS リポジトリに対して、必ずミラーリングを行ってください。これを行うに

は、CentOS ダウンロード・キャッシャー R2 を使用して、BigFix サーバーからアクセスできる場所にローカル・リポジトリを作成します。この場所は、ローカル・キャッシュと呼ばれます。



注: 依存関係の解決時に問題が起こらないように、ローカル・キャッシュには必要なリポジトリがすべて含まれている必要があります

適用時にファイルをダウンロードする際にローカル・キャッシュを使用するには、ダウンロード・プラグインの構成ファイル `plugin.ini` を構成します。以下の手順に従ってください。

1. ローカル・キャッシュの構成 `localCache` を、ダウンロード・キャッシャー・ツールを使用してダウンロードしたファイルの場所に設定します。



注: この場所は、BigFix サーバーからアクセス可能でなければなりません。

2. ダウンロード・キャッシュからのファイルのみをダウンロードし、ベンダーのサイトからはダウンロードしないよう `localCacheOnly` フラグを `yes` に設定します。

「複数パッケージのベースラインのインストール (Multiple-Package Baseline Installation)」タスクを使用してベースラインを実行しました。そのベースラインで失敗した Fixlet のリストを確認するには、どうすればよいですか？

「**アクション情報の表示**」ダイアログを使用すると、ベースライン適用の全体的な進行状況をモニターすることや、各サブアクションの詳細なステータスを確認することができます。

このダイアログにアクセスするには、以下の手順を実行します。

1. ナビゲーション・ツリーで「**アクション**」アイコンをクリックします。
2. アクション・リスト・パネルでアクションを選択します。
3. 作業域で「**コンピューター**」タブを選択します。
4. リストの任意のコンピューターを右クリックします。
5. コンテキスト・メニューから「**アクション情報の表示**」を選択するか、「**編集**」メニューから「**アクション情報の表示**」を選択します。

失敗した Fixlet に関する詳細情報を調べるには、ターゲット・エンドポイントの `/var/opt/BESClient/___BESData/___Global/Logs` にあるクライアント・ログを確認してください。

複数パッケージのインストール・タスクを使用して実行したベースラインが正常に完了しましたが、まだ関連するものとして表示されるのはなぜですか？

依存関係が破損したパッケージを、Fixlet コンポーネントがインストールできなかったことが原因と考えられます。「**複数パッケージのベースラインのインストール (Multiple-Package Baseline Installation)**」タスクは、デフォルトでは、依存関係に問題がないパッケージがターゲット・エンドポイントに正常に適用されるように、破損した依存関係を無視します。

サポートされる CentOS リポジトリにアクセスできるかどうかを確認する方法はありますか？

コマンド・ラインから `--check-baserepos` コマンドと `--check-allrepos` コマンドを使用して、サポート対象の CentOS リポジトリにアクセスできるかどうかを確認できます。これらのコマンドについて詳しくは、[CentOS ダウンロード・キャッシュ R2 の使用情報 \(ページ 29\)](#)を参照してください。

BigFix がサポートする CentOS リポジトリのリストについては、[サポート・マトリックス](#)を参照してください。

CentOS ダウンロード・キャッシュ R2 ツールを使用してリポジトリ・メタデータおよびパッケージをキャッシングするときに必要なスペースを調べる方法がありますか？

ストレージ・スペース所要量を調べるには、`check-storagereq` サブコマンドを使用します。

CentOS ダウンロード・キャッシュ R2 ツールを使用してリポジトリ・メタデータおよびパッケージをキャッシングするときに、スペースを節約するにはどうすればよいですか？

`--sha1_download_dir` を使用することで、スペース節約のベンチマークが設定されます。

`--sha1_download_dir` を使用することで、同じ CentOS バージョンの複数のリポジトリをキャッシュする際のストレージ・サイズ、ダウンロード・サイズ、および時間が大幅に削減されます。これは、同じ CentOS バージョン (たとえば、centos-6.8-x64、centos-6.7-x64、centos-6.6-x64) のリポジトリ間で、多くのパッケージが重複しているためです。CentOS バージョン (たとえば、centos-6.8-x64、centos-7.1-x64) ごとにリポジトリを 1 つだけキャッシュする場合、スペースは節約されません。

出力例を以下に示します。

```
Caching centos-6.8-x64, centos-6.7-x64, centos-6.6-x64
(with --sha1_download_dir):
Total Repo Metadata and Packages will take up 28 GB of space
instead of 37 GB (23% space saved)
```

EDR ログに、何もインストールできないため、最新のカーネルを使用しているかどうか確認するよう警告するメッセージが表示され、Fixlet のインストールが失敗しました。この場合の対処方法を教えてください。

このメッセージは、カーネル・パッケージをデプロイする Fixlet の場合にのみ表示されます。エンドポイントにターゲット・カーネル・パッケージがインストールされていない場合、またはエンドポイントのアクティブ・カーネルがターゲット・カーネル・パッケージより低いバージョンである場合、カーネル Fixlet が関連状態になります。エンドポイントに最新のカーネルがインストールされているもののアクティブに使用していない場合も、カーネルの脆弱性の対象と見なされます。

To remediate the issue, restart the endpoint and ensure it is using the latest kernel available.

Notices

This information was developed for products and services offered in the US.

HCL may not offer the products, services, or features discussed in this document in other countries. Consult your local HCL representative for information on the products and services currently available in your area. Any reference to an HCL product, program, or service is not intended to state or imply that only that HCL product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any HCL intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-HCL product, program, or service.

HCL may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085
USA
Attention: Office of the General Counsel

For license inquiries regarding double-byte character set (DBCS) information, contact the HCL Intellectual Property Department in your country or send inquiries, in writing, to:

HCL
330 Potrero Ave.
Sunnyvale, CA 94085
USA
Attention: Office of the General Counsel

HCL TECHNOLOGIES LTD. PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. HCL may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-HCL websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this HCL product and use of those websites is at your own risk.

HCL may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

HCL

330 Potrero Ave.

Sunnyvale, CA 94085

USA

Attention: Office of the General Counsel

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by HCL under terms of the HCL Customer Agreement, HCL International Program License Agreement or any equivalent agreement between us.

The performance data discussed herein is presented as derived under specific operating conditions. Actual results may vary.

Information concerning non-HCL products was obtained from the suppliers of those products, their published announcements or other publicly available sources. HCL has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-HCL products. Questions on the capabilities of non-HCL products should be addressed to the suppliers of those products.

Statements regarding HCL's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to HCL, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. HCL, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS," without warranty of any kind. HCL shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work must include a copyright notice as follows:

© (your company name) (year).

Portions of this code are derived from HCL Ltd. Sample Programs.

Trademarks

HCL Technologies Ltd. and HCL Technologies Ltd. logo, and hcl.com are trademarks or registered trademarks of HCL Technologies Ltd., registered in many jurisdictions worldwide.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other product and service names might be trademarks of HCL or other companies.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the HCL website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of HCL.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of HCL.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

HCL reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by HCL, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

HCL MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.