



# JBoss Enterprise Application Platform 6.2

## 6.2.1 リリースノート

Red Hat JBoss Enterprise Application Platform 6.2 向け  
エディション 6.2.1



# JBoss Enterprise Application Platform 6.2 6.2.1 リリースノート

---

Red Hat JBoss Enterprise Application Platform 6.2 向け  
エディション 6.2.1

Nidhi Chaudhary

Lucas Costi

Russell Dickenson

Sande Gilda

Vikram Goyal

Eamon Logue

Darrin Mison

Scott Mumford

David Ryan

Misty Stanley-Jones

Keerat Verma

Tom Wells

## 法律上の通知

Copyright © 2014 Red Hat, Inc.57.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](#). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 概要

本リリースノートには、Red Hat JBoss Enterprise Application Platform 6.2.1. に関する重要な情報が含まれています。リリースノートをすべて読んでから製品をインストールしてください。

---

## 目次

1. RED HAT JBOSS ENTERPRISE APPLICATION PLATFORM 6 .....	2
2. メンテナンスリリースの概要 .....	2
3. パッチ更新の適用 .....	3
4. セキュリティーアドバイザリー .....	3
5. バグ修正 .....	3
A. 改訂履歴 .....	13

## 1. RED HAT JBOSS ENTERPRISE APPLICATION PLATFORM 6

Red Hat JBoss Enterprise Application Platform 6 (JBoss EAP 6) は、オープンな標準に基づき構築され、Java Enterprise Edition 6 の仕様に準拠する高速でセキュアな高性能ミドルウェアプラットフォームです。高可用性クラスタリング、強力なメッセージング、分散キャッシングなどの技術を JBoss Application Server 7 と統合し、安定したスケーラブルな高速プラットフォームを作り上げます。

新しいモジュラー構造により、必要な時だけサービスを有効にできるため、起動速度が大幅に向上します。管理コンソールと管理コマンドラインインターフェースを使用すると、XML 設定ファイルを手作業で編集する必要がなくなるため、スクリプトを作成して作業を自動化することが可能です。さらに、API と開発フレームワークも含まれており、これらを使用して堅牢で拡張性のある、セキュアな Java EE アプリケーションを迅速に開発することができます。

[バグを報告する](#)

## 2. メンテナンスリリースの概要

ユーザーの要望に応えるため、JBoss EAP 6 のマイクロリリースは 6.2 リリースで廃止され、代わりに 6 週間ごとをめぐりに定期的にリリースされるパッチ更新が導入されました。パッチ更新は、JBoss EAP の ZIP および RPM ベースのインストールに対して配信されます。

- ZIP ファイルによるインストールには、パッチの適用、削除、およびパッチ状態の表示を行う EAP の新機能によって管理される累積パッチが配信されます。
- RPM によるインストールには、累積パッチと同じ修正が含まれる更新 RPM が、同じ周期で Red Hat Network より配信されます。

新しいパッチ更新には、ユーザーが報告した問題に対するバグ修正が含まれ、場合によってはセキュリティに関する修正も含まれます。パッチ更新によって、作成される個別パッチの数が大幅に減少することが予想され、ユーザーはインストールを最新の状態に保つ必要があります。

詳細は、Red Hat ナレッジベースのアーティクル、『EAP 6.2+ におけるメンテナンスリリースモデルの変更』 (<https://access.redhat.com/site/ja/articles/643163>) および 『EAP 6.2 で変更になったパッチ管理』 (<https://access.redhat.com/site/ja/articles/653723>) を参照してください。

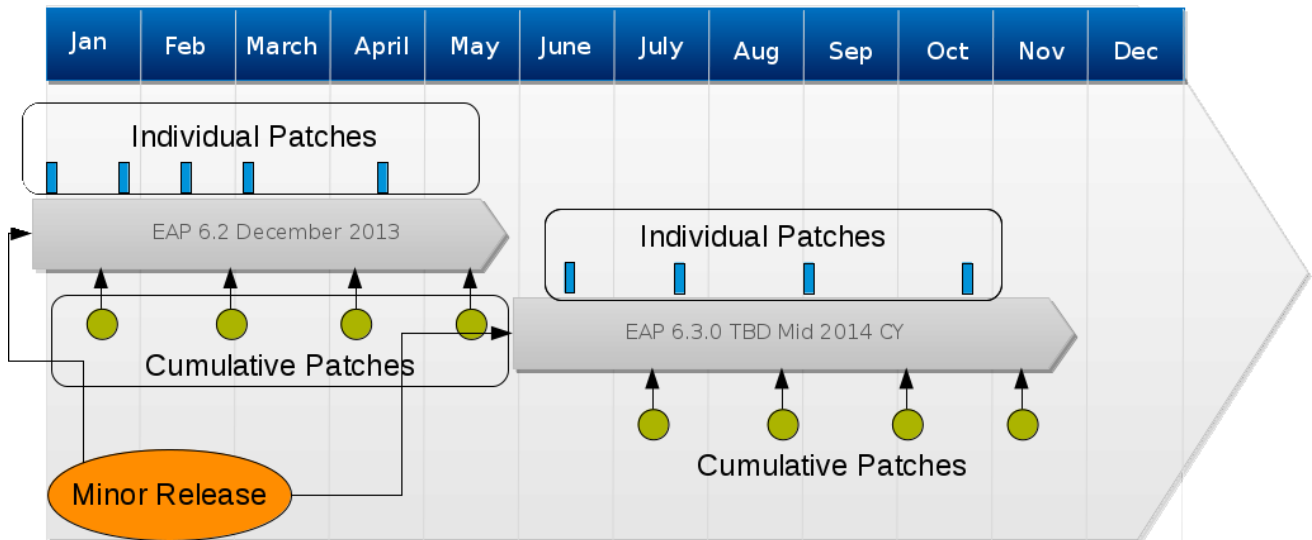


図1 JBoss EAP のパッチスケジュール

[バグを報告する](#)

### 3. パッチ更新の適用

パッチ更新の適用方法は、JBoss EAP のインストールに使用されたインストール方法によって異なります。詳細は、『インストールガイド』の「JBoss EAP のパッチとアップグレード」を参照してください。

ローカル Maven リポジトリへ累積パッチを適応する方法については、JBoss EAP 『開発ガイド』の「ローカル Maven リポジトリへのパッチ適用」を参照してください。

[バグを報告する](#)

### 4. セキュリティーアドバイザー

#### Zip

<https://rhn.redhat.com/errata/RHSA-2014-0172.html>

#### RPM/Red Hat Enterprise Linux

Red Hat JBoss Enterprise Application Platform 6 の EL5 セキュリティーアドバイザー

<https://rhn.redhat.com/errata/RHSA-2014-0170.html>

Red Hat JBoss Enterprise Application Platform 6 の EL6 セキュリティーアドバイザー

<https://rhn.redhat.com/errata/RHSA-2014-0171.html>

[バグを報告する](#)

### 5. バグ修正

#### CDI/Weld

### 1034776 EJB のビジネスインターフェースの処理が不適切 (ビジネスメソッドの定義クラスが EJB のビジネスインターフェースでない場合)

セッション Bean オブザーバーメソッドの検証が、スーパーインターフェースを持つローカルインターフェース上で宣言されたメソッドを適切に検証しませんでした。

その結果、「WELD-000088, Observer method must be static or local business method」によってデプロイメントに失敗することがありました。これは、オブザーバーメソッドを持つセッション Bean クラスが、スーパーインターフェースによって定義されたメソッド署名を持たないインターフェースより継承され、同時にセッション Bean クラスがメソッド実装を持つクラスを拡張する場合に発生しました。

checkObserverMethods メソッドに継承されたメソッドが含まれるよう、Weld の内部セッション Bean 実装が修正されました。そのため、前述の実装のデプロイメントが正常にデプロイされるようになりました。

### 1051205 EAP 6 クラスタで CDI Bean をレプリケートすると NPE が発生する

Bean デプロイメントアーカイブ (BDA) の ID を作成するメソッドに問題があり、クラスタード間で BDA が同一であることをコンテナが認識しなかったため、セッションをレプリケートできませんでした。このエラーは修正され、BDA がクラスタード間で同じであると認識されるようになりました。

### 1052277 「org.jboss.weld.exceptions.DeploymentException: WELD-001414 Bean name is ambiguous」の永久修正

複数のサブデプロイメントを持つデプロイメントでは、Bean 名の曖昧さの検証が分離されなかったため、場合によっては「DeploymentException」が発生する原因となりました。本リリースの Red Hat JBoss EAP 6 では Weld コンポーネントがアップグレードされ、この問題は修正されました。

## クラスローディング

### 1051335 モジュール 「org.jboss.log4j.logmanager」は「javax.mail.api」モジュールに依存する必要がある

log4j.logmanager モジュールは javax.mail.api モジュールと暗黙的な依存関係を持っていましたが、module.xml ファイルには明示的に記述されていませんでした。これにより、java メールが必要なアペンダーを使用するアプリケーションがデプロイされると、ClassNotFoundException がスローされる原因となりました。オプションの依存関係がモジュールに指定されたため、この例外は発生しないようになりました。

## ドメイン管理

### 1051189 管理コンソールのログアウト手順が LDAP 接続で失敗する

以前のバージョンの JBoss EAP 6 では、管理コンソールのログアウト手順が LDAP 接続で失敗する問題がありましたが、この問題は本リリースで修正されました。

この問題は、ログアウト論理が LDAP では使用されない DIGEST 認証を想定したため発生しました。

本リリースでは、BASIC 認証が使用されているかを検出し、使用されている場合は BASIC 認証のチャレンジを使用して、キャッシュされたクレデンシャルを無視するよう Web ブラウザーに働きかけます。



これにより、ユーザーは管理コンソールの LDAP 接続から正常にログアウトできるようになりました。

#### 1050068 タイプ別にディレクトリーをグループ化すると、誤った logging.properties ファイルが使用される

directory-grouping="by-type" でサーバーを定義すると、domain/configuration/logging.properties ファイルが常に使用されましたが、これは不適切な設定でした。その結果、誤ったロギング設定が使用されました。これは、不適切なロジックによって誤ったプロパティファイルが選択されたことが原因でした。この問題を解決するため、プロパティファイルが選択されるロジックが改良され、directory-grouping="by-type" が使用されると正しい logging.properties ファイルが使用されるようになりました。

#### 1052365 syslog 上で監査ログの FACILITY を設定する方法がない

以前のリリースの JBoss EAP 6 では、syslog 上で監査ログの **FACILITY** を設定する方法がありませんでした。本リリースでは、この設定オプションが有効になり、次のように使用できます。

```
--- -Dorg.jboss.TEMP.audit.log.facility=N ---
```

RFC-5424 の 6.2.1 項にあるとおり、**N** は FACILITY になります。

#### 1047536 管理監査設定: syslog プロトコルは小文字だが大文字が想定されている

syslog ハンドラーがドメインコントローラーに追加されると管理監査設定のバグが原因でサーバーインスタンスが失敗しました。

org.jboss.as.domain.management.audit は UDP プロトコルを小文字 (**udp**) で定義し、列挙は大文字 (**UPD**) を使用することが判明しました。

org.jboss.as.domain.management.audit のプロトコルを大文字で定義することで、この問題は修正されました。

#### 1051240 LDAP グループロード - 存在しないユーザーは失敗すべきでない

以前のリリースの JBoss EAP 6 では、ドメインモードサーバーインスタンスが起動しない問題がありました。

LDAP よりグループをロードするようセキュリティレームが設定されているドメインモードでは、サーバーインスタンスがサーバーユーザーのグループ情報をロードしようとして失敗しました。

リモートユーザーが認証されたサーバーの場合はグループ情報をロードしないようコードを変更したため、この問題は修正されました。

LDAP ディレクトリーにユーザーがなくても、サーバーインスタンスは失敗しないようになりました。

#### 1044394 - システムプロパティの誤処理

本リリースの JBoss EAP では、システムプロパティの誤処理に関する複数の問題が修正されました。詳細は、リンク先の Bugzilla チケットを参照してください。

#### 1007696 ランタイム名がデプロイメント名と異なると CLI がアプリケーション状態を表示しない

アプリケーションのランタイム名が EAR 名と異なると、管理 CLI がアプリケーションの状態を表示せず、代わりに「No metrics available.」というメッセージを返しました。この問題は、管理 CLI が EAR 名でアプリケーションをクエリしたことが原因で、アプリケーション名が EAR 名と異なると、アプリケーションが一致しませんでした。この問題を解決するため、アプリケーションのランタイムで検索が実行されるようになりました。その結果、ランタイム名が EAR 名と一致しなくても CLI 操作が正常に実行されるようになりました。

### 1039699 ドメインで同じランタイム名のデプロイメントを使用できない

domain.xml ファイルの解析に使用されるコードによって、ランタイム名の設定値はドメインのすべてのデプロイメントで一意であると誤って検証されました。この設定値は、サーバーグループへマップされたデプロイメントのセット内でのみ一意である必要があります。

ドメインの同じランタイム名を持つ 2 つのデプロイメントを使用した場合 (同じデプロイメントの 2 つのバージョンなど)、ドメインコントローラーが再起動されると致命的な起動エラーが発生し、ホストコントローラーのログに以下のような出力が記録されました。

```
JBAS010932: Caught exception during boot:
org.jboss.as.controller.persistence.ConfigurationPersistenceException:
JBAS014676: Failed to parse configuration
...
Caused by: javax.xml.stream.XMLStreamException: ParseError at [row,col]:
[1348,9]
...
Message: JBAS014664: An element of this type named foo has already been
declared
```

ランタイム名の一意性に対する XML パーサー検証の範囲が個別のサーバーグループに限定されました。また、管理設定モデルの解析後の検証が改良され、無効とマーク付けされたデプロイメントが除外されるようになりました。同じランタイム名を持つ 2 つのデプロイメントが設定に含まれるドメインコントローラーを正しく再起動できるようになりました。

### 1039875 AbstractBoottimeAddStepHandler にないコンストラクターの追加

アップストリームにて org.jboss.as.controller.AbstractBoottimeAddStepHandler のスーパークラスに新しいコンストラクターが追加されましたが、ハンドラー自体へ追加されなかったため、クラスが持つコンストラクターが異なりました。そのため、これらのコンストラクターが必要に応じて追加されました。

### 1041330 DelegatingServerInventory の NPE

ホストコントローラーの起動直後に管理リクエストがホストコントローラーへ送信されると、**NullPointerException** タイプの例外によって受信に失敗することがあります。これは、ホストコントローラーがリクエストを受信できるようになる時点と、サーバー管理システムが完全起動する時点に差があるからです。通常、この差異は 1 秒未満ですが、この間に送信されたリクエストは受信されません。

### 1053382 syslog 上で監査ログの appName を設定する方法がない

以前のバージョンの JBoss EAP 6 では、システムプロパティから appName を設定できませんでした。この機能は RFC-5424 仕様では必須となっていました。この機能が JBoss EAP 6.2.1 に追加され、以下のコマンド構文で使用できるようになりました。

```
-Dorg.jboss.TEMP.audit.log.appName=APP_NAME
```

## 1040480 SecurityManager が有効な状態で実行すると問題が発生する

本リリースの JBoss EAP 6 では、Java Security Manager の使用によって発生した複数の問題が修正されました。詳細は、リンク先の Bugzilla チケットを参照してください。

### HornetQ

#### HORNETQ-1280 アドレス設定 <expiry-delay/> が絶対的に使用される

<address-setting/> の <expiry-delay/> に対して指定された値が、誤ってエポック時刻からの絶対時間 (ミリ秒) として解釈されました。たとえば、**1000** を値として指定した場合、1000 ミリ秒の *expiry-delay* ではなく、**1970-01-01 00:00:01 GMT** として計算されました。このエラーにより、*expiry-delay* が想定どおり挙動しませんでした。この問題を解決するため、現在の時間が計算に追加されたため、パラメーターの値が適切に算出されるようになりました。

#### HORNETQ-1281 すべてのレベルでパスワードをログに記録しない

状況によって、SSL キーストア/トラストストアのパスワードが INFO レベルでログに記録され、MDB のアクティベーション設定プロパティのパスワードセットが TRACE レベルでログに記録されることがありました。

コードのリファクタリングが行われ、パスワードがどのレベルでもログに記録されないようになりました。

#### HORNETQ-1284 コンシューマー最大レートの挙動が直感的でない

consumer-max-rate ロジックが想定どおり挙動しませんでした。consumer-max-rate が指定された場合、このレートはコンシューマーの生存期間中の平均として算出されました。たとえば、consumer-max-rate が **10** の場合、コンシューマーが生存している間は 1 秒ごとに 10 個のメッセージを消費できました。よって、消費されるメッセージがキューに届くまで、コンシューマーが 60 秒間生存した場合は、1 秒ごとに 600 個のメッセージを消費することが可能でした。

consumer-max-rate が秒単位で算出されるようになったため、コンシューマーは consumer-max-rate を越える数のメッセージを 1 秒ごとに消費できなくなり、コンシューマーの生存期間中の平均として算出されないようになりました。

#### HORNETQ-1290 xaStart がタイムアウトを送信すると、トランザクションのサーバーがダーティー状態になる

クライアントから XA トランザクションを起動する基盤の操作に失敗した場合、クライアントのセッションに対して維持されたサーバー側の状態が「ダーティー」になり、その後の操作に問題が発生しました。

サーバー側のセッションが適切にこのような操作を処理するようになりました。

#### HORNETQ-1289 XmlDataExporter & XmlDataImporter への JMS データのサポート追加

ジャーナルと動作する `XmlDataExporter` および `XmlDataImporter` ツールが、JMS ジャーナルファイルのデータ (JMS 宛先および接続ファクトリーバインディングなど) を処理しませんでした。

このデータのエクスポートおよびインポートのサポートが追加されました。

#### HORNETQ-1292 メッセージがドロップしたときにディスクから大型メッセージを削除

address-full-policy が DROP だった場合、宛先が満杯でも大型のメッセージがディスクから削除されませんでした。

アドレスが満杯で `address-full-policy` が `DROP` の場合に、大型のメッセージがディスクから削除されるようになりました。

#### **HORNETQ-1231** `listDeliveringMessagesAsJSON` が消費されたメッセージを無視する

`listDeliveringMessagesAsJSON` が想定されるすべてのメッセージを列挙しませんでした。

`listDeliveringMessagesAsJSON` が適切なメッセージをすべてリストするようになりました。

#### **HORNETQ-1296** `DeliveringCount` および `MessageCount` が負の値になる

状況によっては、サーバーが `kill` されたときに `DeliveringCount` および `MessageCount` プロパティが負の値になることがありました。

これらのプロパティが負の値にならないようになりました。

#### **HORNETQ-1299** XML インポーターにフラグを追加して `AS7/Wildfly` の追加 `JNDI` エントリーを作成

JBoss EAP 6 インスタンスには、リモートクライアント向けの特別な `JNDI` エントリーが必要です。スタンドアロン HornetQ または以前のバージョンの EAP よりデータをインポートする場合、リモートクライアントに必要な `JNDI` エントリーがありませんでした。特別なエントリーを自動的に作成するようインポーターに指示するフラグを追加する必要がありました。

必要な `JNDI` エントリーを作成するため、新しいフラグが XML インポーターに追加されました。

#### **HORNETQ-1303** XML エクスポーターが大型のページ化メッセージに `NPE` をスローする

大型のページ化メッセージがエクスポートされると、XML エクスポーターによって `NullPointerException` がスローされました。

XML エクスポーターが大型のページ化メッセージを適切に処理するようになりました。

#### **HORNETQ-1304** XML エクスポーターがバインドिंगのないページ化メッセージに `NPE` をスローする

エクスポート中に、ページ化メッセージによってジャーナルにないバインドिंगが参照されると、XML エクスポーターによって `NullPointerException` がスローされました。

XML エクスポーターが、ジャーナルに記録されていないバインドिंगを参照するページ化メッセージを適切に処理するようになりました。

## IIOP

#### **1036819** Backport `JacORB #904` `CDRInputStream.read_string` は 0 文字列サイズを正常に処理する必要がある

関数 `CDRInputStream.read_string` が空の文字列を誤処理し、長さをゼロ (0) として算出したため、マーシャル例外が発生しました。空の文字列の処理が修正され、`CDRInputStream.read_string` が空の文字列を返すようになったため、以前のエラーが発生しないようになりました。

## JBossWeb

#### **JBWEB-281** 一貫しない `genStringAsCharArray` の使用

JSP 設定プロパティ `genStringAsCharArray` の名前が JBoss Web で一貫していませんでした。そのため、生成されたサーブレットコードに文字列ではなく `char` 型配列を使用する機能が有効にできませんでした。

この問題は本リリースの JBoss EAP 6 で修正され、この機能が適切に有効化できるようになりました。

### JBWEB-282 org.apache.catalina.security.SecurityUtil が同期化されない

`org.apache.catalina.security.SecurityUtil` の `doAsPrivilege` メソッドが `HashMap` へのアクセスを同期化しませんでした。これにより、複数のアプリケーションが同時にデプロイされるとマップへの同時アクセスが可能であったため、`HashMap` アクセスが CPU の使用率が高くなるループになりました。

この問題は、本リリースの JBoss EAP 6 で修正されました。この問題に対処するため、`SecurityUtil` の `HashMap` がスレッドセーフの `ConcurrentHashMap` に置き換えられました。

## JCA

### 1047961 サーバー起動中の競合 - RA がデプロイされない

競合によって、一部のインスタンスでカスタムリソースアダプターがデプロイされない問題がありました。

競合が発生したのは、`org.jboss.as.connector.subsystems.resourceadapters.ResourceAdapterService` の起動が RA のデプロイに間に合わなかったためです。この問題が発生しないようにするため、コードが追加されました。

## JPA

### 1037636 JBoss AS/Hibernate JPA 統合でのメモリーリーク

クエリキャッシュを統計をチェックする時に発生したメモリーリークは、本リリースの JBoss EAP 6 で修正されました。

メモリーリークは、クエリキャッシュをチェックするために `jboss-cli` が呼び出されると発生しました。たとえば、以下コマンドによってリークが発生します。

```
./jboss-cli.sh --connect --  
command='/deployment=DeploymentName.ear/subdeployment=MyEjb.jar/subsystem=jpa/hibernate-persistence-unit=DeploymentName.ear\MyEjb.jar#MyPersistenceUnit:read-children-resources(child-type=query-cache)'
```

クエリキャッシュの統計をチェックする時に無効なクエリ名が使用されたため、無効な結果が返され、さらに無効なクエリ名が Hibernate の統計に追加されました。本リリースでは、クエリキャッシュの統計をチェックする時に無効なクエリ名が使用されないようになりました。

## ロギング

### 1038862 SEVERE ロギングレベルが CLI で使用できない

以前のバージョンの JBoss EAP では、CLI を使用すると SEVERE ロギングレベルが使用できませんでした。

この問題は本リリースで対処され、SEVERE をロギングレベルとして設定できるようになりました。

### 1031221 strong=true の場合、LogContext のルートロガーが強い子ハッシュマップで作成されない

以前のバージョンの JBoss EAP 6 では、ロガーのガベージコレクションが早期に行われ、ロガーと設定が失われることがありました。たとえば、ロガーにアベンダーが割り当てられた場合、ロガーのガベージコレクションが行われると、アタッチされたアベンダーがない状態で新しいロガーが作成されました。本リリースでは、弱参照ではなく強参照を使用するようロガーのマップが変更されました。これにより、ロガーのガベージコレクションが実行されなくなり、適切に動作するようになりました。その結果、カスタムロガーが想定どおり処理されるようになりました。

### 1049074 デプロイメントユニットプロセッサ (DUP) のロギングがクラスローダーをリークする

以前のバージョンの EAP 6 では、デプロイメントごとのロギングまたはロギングプロファイルがデプロイメントで使用されると、アンデプロイメントによってクラスローダーが LogContextSelector から削除されませんでした。これは、ClassLoaderLogContextSelector は呼び出しスタックを使用して LogContext を判断したためです。

本リリースでは、正しい LogContext が DeploymentUnit にアタッチされました。アンデプロイ時に、デプロイメントのクラスローダーに依存せずに、デプロイメントにアタッチされた LogContext を使用して、セクターから LogContext が削除されるようになりました。アンデプロイ時にクラスローダーが想定どおり削除されるようになりました。

## 命名

### 1050184 LDAP: InitialDirContext.search() メソッドが URL が含まれる name パラメーターを許可しない

本リリースの JBoss EAP には、最初の **name** フィールドに URL およびポート番号が含まれていると LDAP サーバーの検索に失敗し、例外がスローされる問題の修正が含まれています。

この形式がサポートされるようになり、例外がスローされないようになりました。

### 1061631 InitialContext によって元の例外の原因が表示されない

InitialContext のインスタンス化に失敗し、以下のメッセージが表示されることがありました。

```
javax.naming.NamingException: JBAS011843: Failed instantiate
InitialContextFactory com.sun.jndi.ldap.LdapCtxFactory from classloader
ModuleClassLoader for Module
"deployment.externalContextBindingTest.jar:main" from Service Module
Loader
```

このメッセージには原因が表示されず、トラブルシューティングの実行が不可能でした。この問題を解決するため、原因が表示されるようになりました。InitialContext のインスタンス化に失敗すると、エラーメッセージに根本的な原因が表示されるようになりました。

```
javax.naming.NamingException: JBAS011843: Failed instantiate
InitialContextFactory com.sun.jndi.ldap.LdapCtxFactory from classloader
ModuleClassLoader for Module
"deployment.externalContextBindingTest.jar:main" from Service Module
```

```
Loader [Root exception is javax.naming.CommunicationException:
127.0.0.1:10389 [Root exception is java.net.ConnectException: Connection
refused]]
```

## リモート

### 1032312 異なるシリアライズ不可能なスーパークラスを持つシリアル可能クラスとの間でプロトコルの互換性がない

以前のバージョンの JBoss EAP 6 では、マーシャリング層によって、シリアライゼーションストリームへのシリアライズ不可能なクラスに関連する余分なデータが発生しました。これにより、パフォーマンスの劣化、ClassNotFounds、異なる JVM 間でのその他のエラーなど、不適切な挙動が多岐にわたって発生しました。

本リリースでは、余分なデータが省略されました。プロトコルのパフォーマンスが向上され、同じクラスの異なるクラス階層を持つ JVM の間で互換性が確保されました。

### 1030104 IBM-I 上で IBM JDK を実行する場合のセグメンテーション違反およびその他の xnio 問題

本リリースの JBoss EAP 6 では、セグメンテーション違反によって IBM システム上のアプリケーションが実行に失敗する問題が修正されました。

この不具合は、IBM のオペレーティングシステムで使用するために最適化された IBM JDK の NIO 実装が原因でした。JBoss EAP の I/O 層はこの実装を検出し、使用を試みますが、一部のオペレーティングシステム (IBM-I など) によってはこの実装によってセグメンテーション違反が発生しません。

本リリースでは、これらのオペレーティングシステムが検出され、安全なフォールバックが使用されるようになりました。これにより、セグメンテーション違反の問題が修正され、アプリケーションサーバーが突然クラッシュしなくなりました。

## セキュリティ

### 1060972 PicketBox コンポーネントに追加の Java Security Manager チェックを追加

追加の Java Security Manager チェックが PicketBox コンポーネントに追加されました。

## サーバー

### 1050003 「Class-Path: .」によって「BAS011046: A component named 'TestBean' is already defined in this module」が発生

**Class-Path** を . として設定する JAR マニフェストが含まれる WAR をデプロイすると、JAR ファイルのアノテーションが 2 回処理されることがありました。これにより、デプロイメントに失敗し、**JBAS011046: A component named 'TestBean' is already defined in this module** に似たエラーが生成されました。

この問題は、本リリースの JBoss EAP 6 で修正されました。JAR マニフェストの処理時に、**Class-Path: .** が含まれる JAR マニフェストはチェックされ無視されるようになりました。

## Web

### 1050033 WAIT\_FOR\_BEFORE\_START は / コンテンツアプリケーションで動作しない

Web コネクタサービスのバグにより、/ コンテンツアプリケーションのデプロイ時に **WAIT\_FOR\_BEFORE\_START** が動作しませんでした。**WAIT\_FOR\_BEFORE\_START** が / に設定され、/ アプリケーションがデプロイされると、コネクタが起動しませんでした。

この問題は本リリースの JBoss EAP 6 で修正されました。**WAIT\_FOR\_BEFORE\_START** が / に設定され、/ コンテンツアプリケーションをデプロイすると、コネクタが正常に起動するようになりました。

## Web コンソール

### 1051171 セキュア化 (ssl) 管理コンソールの設定からログアウトすると http アドレスにリダイレクトされる

ログアウトのリダイレクトの処理にバグがあり、SSL でセキュア化された (https) Web 管理コンソールからログアウトすると、誤ってセキュア化されていない (http) ログアウトページにリダイレクトされました。その結果、**Page Not Found (404)** エラーが表示されました。

この問題は、本リリースの JBoss EAP 6 で修正されました。ログアウトのリダイレクトが適切に Web 管理コンソールのプロトコルとポートを保持するようになりました。セキュア化された管理コンソールからログアウトすると、セキュア化されたログアウトページが表示されるようになりました。

## [バグを報告する](#)



## A. 改訂履歷

改訂 6.2.1-15

Thursday Feb 13 2014

Russell Dickenson

Red Hat JBoss Enterprise Application Platform 6.2.1 GA