

ビルド欠陥の排除と速度低下の防止

ソフトウェア開発工程のビルドシステムで、深刻な不具合が発生することがよくあります。ビルドに時間がかかったり、またはエラーの発生により、ソフトウェアアプリケーションの市場投入が遅れ、莫大なコストがかかる可能性があります。現在の典型的な開発環境では特にその傾向が強く、整合性の高いソフトウェア製品を迅速に開発するには、信頼できる効率的なビルドプロセスが必要になります。

これまで、企業の開発チームには、アセンブリされたソフトウェア コンポーネントの品質およびセキュリティと、ビルドに失敗した原因を判断するための、深い理解が不足していました。

Coverity のビルド解析機能は、Coverity Integrity Center の主要部分であり、この機能を採用することにより、開発チームは、ソフトウェアのすべてのコンポーネントをビジネス要件と整合性基準に適合させることができます。ソフトウェアのビルドプロセスを深く理解することで、ソフトウェア開発、市場への出荷を遅らせる原因となる問題を特定し、時間のかかる、または失敗したビルドの的確な原因を突き止めることができます。

ビルド解析の利点

- 開発者の生産性を向上
- ビルドプロセスの変更を効率的に管理
- セキュリティとコンプライアンスを確保
- ソフトウェアの整合性を改善

生産性

この機能によって、開発者チームは、ビルドの全体的な整合性を正しく評価できるようになり、どのコンポーネントがアセンブルされているのか、それらは正しいコンポーネントなのか、どのプロセスがどれくらいの頻度で実行されたのか、といった領域を正確に把握できるようになりました。また、ビルド解析を使用してルールを設定し、正しいコンパイルフラグを確保して、ビルドに存在してはならないディレクトリやライブラリの追加または削除ができます。

変更

ビルド解析を使用すると、エンジニアは、ビルド時に実行する全てのプロセスのマップを作成できるようになるため、変更の影響を把握できます。このマップには、プロセスごとに、そのプロセスに対する全ての入力ファイルと、そのプロセスで生成される全ての出力ファイルの完全なセットが含まれています。開発者は、ファイルおよびプロセス粒度を持つ、ソフトウェア生産プロセス全体の依存ツリーを利用できるようになりました。

コンプライアンス

ソフトウェア コンポーネントの包括的な一覧を作成する機能で、ビルド解析によって生成された、重要なコンプライアンスとレポート データが提供されます。これには、チェックアウトされたファイルが実際に特定のビルドに含まれているファイルであることや、ソースコード管理 (SCM) システムで生成された一覧が特定のビルドに対して正確であることが示されます。

ソフトウェアの整合性

開発者チームは、ビルドされたアプリケーションを開発サイクルの開発段階に移行できるか、ソフトウェアの整合性を改善するために、別のステップを実行すべきかを正確に判断できるようになりました。Coverity のビルド解析を使用すると、ソフトウェアのビルド時間が短縮されるため、製品の市場投入も短縮できます。

Coverity のビルド解析で下記が実現します

- 不正なオブジェクトファイルの不適切または偶発的な使用による欠陥を排除

- ファイルへのアクセス違反やリークされたファイルを検出

- プロセス起動の不要な重複を発見

- 時間のかかるビルドの根本原因を特定して排除

- ビルド時に実行されるすべてのプロセスをマッピング

- 依存関係を包括的に把握する

- 変更が実際に実行されていることを確認

- 包括的なコンポーネント一覧を作成して、適切なコンポーネントの包含を確認

- 内部およびオープンソースコードの使用を確認して検証

- コンプライアンス要件に合わせた基本的なデータを生成

- 生産を加速化して、コンポーネントの不要な再テストを削減