加速しよう、未来を。

VERISERVE

ソースコードファジングツール Mayhem

ソースコードの潜在的なバグや脆弱性を検出し、ソフトウェアのセキュリティや堅牢性を向上させます

① ソースコードの動的解析（ファジング）を実行

- Mayhem プラットフォーム上でファズの生成、アプリケーションの実行、ファズの送信を繰り返すことで、動的にアプリケーションを解析します。
- 解析結果を GUI 上でリアルタイムに確認できます。
- あらゆるテストケースを再現してテストするため、偽陽性はありません。

② シンボリック実行技術※をファズ生成に応用

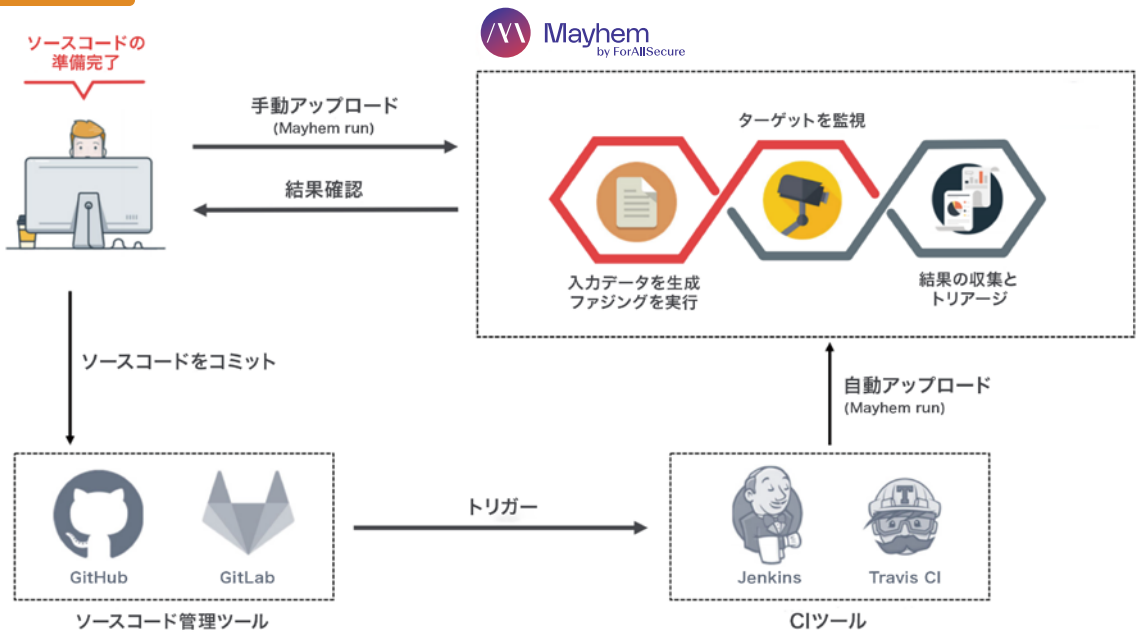
- より深いパスまで解析することができます。

※シンボリック実行（Symbolic execution）によるファズ生成エンジンは、カーネギーメロン大学の研究による特許技術を元に開発されたものです。

③ 開発の初期段階から ソフトウェアセキュリティの実装が可能

- 動的テストのシフトレフトにより、開発の初期段階からバグや脆弱性を発見することで修正コストを削減できます。
- DevSecOps を念頭に置いて開発されており、さまざまな CI/CD ツールと連携できます。

ご利用イメージ

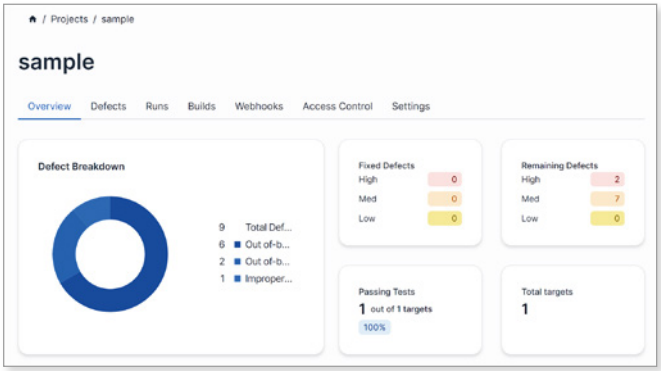


APIファジングを実行するオプション機能もございます。

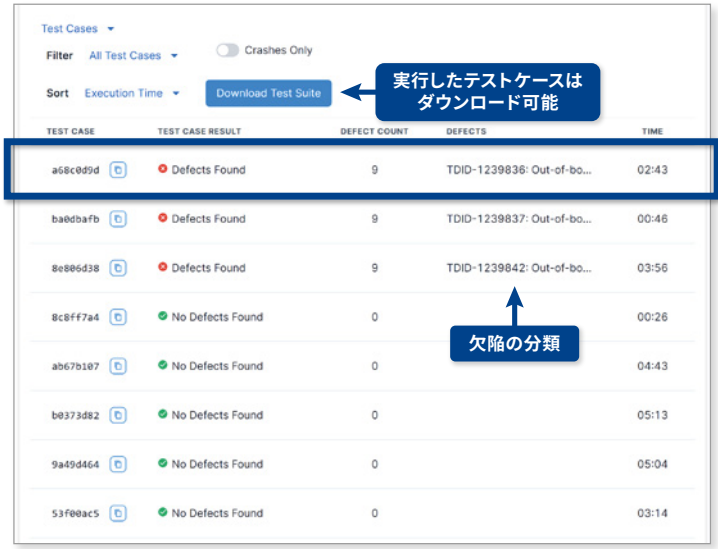
主な機能

ファジングの進行状況と検出された欠陥をリアルタイムで確認できます。

見つかった欠陥情報のサマリーをダッシュボードで確認できます。



実行したテストケースを一覧で確認することができます。



各テストケースの詳細を確認することができます。欠陥を再現するコマンドが提供されるため、再現確認を容易に実施することができます。また、テストケースの実行時の標準出力やバックトレース等の情報が提供され、欠陥の原因の特定に役立てることができます。

ファジングのカバレッジ情報が提供されます。
他ツールと組み合わせることでカバレッジをさらに可視化することもできます。

ソースコードファジングサービスの実施手順（Mayhemを使用）

お見積り
- 要件をヒアリングの上、お見積りいたします。

※ツールの選定なども含みます。

キックオフ
- スコープを確定します。
- テストの実施方法や、監視方法、終了要件を設定します。

ファジング計画
- 決定したスコープや要件に基づき、テスト計画を作成いたします。
- 要件を網羅するのに十分なファズデータを準備します。

ファジングの実行
- ファジングを実行します。
- 実行結果やパフォーマンスに応じてチューニングを行います。

結果の確認＆レポート
- テスト結果の確認を行い、脆弱性が検出された箇所をレポートします。
- 再利用可能なテストケースは次回の実行に活用できるようDB化します。

テスト対象はLinux上で実行可能なバイナリファイルに制限させていただいています。

APIファジングサービスも同様の手順でご提供できます。

加速しよう、未来を。
VERISERVE
https://www.veriserve.co.jp

東京都千代田区神田三崎町 3-1-16 神保町北東急ビル 9 階
サイバーセキュリティ事業部
mayhem@veriserve.co.jp

Mayhem
詳細ページはコチラ