



# イメージスキャン



本文の内容は、Sysdig Secure キャプチャーのドキュメント (<https://docs.sysdig.com/en/captures-122718.html>) を元に日本語に翻訳・再構成した内容となっております。

<b>イメージスキャン</b>	<b>9</b>
前提条件	9
Sysdigイメージスキャンの仕組み	9
イメージコンテンツレポート	10
使用される脆弱性データベース	10
ユースケース	11
コンテナ開発中のスキャン (DevOps)	11
実行中のコンテナのスキャン (セキュリティ担当者)	12
<b>イメージスキャンを開発パイプラインに統合する</b>	<b>13</b>
インラインスキャン	13
前提条件	13
インラインスキャンを実装する	13
パイプライン統合の例	14
Jenkinsと統合する	14
Jenkinsプラグインのインストールと構成	14
Jenkinsでスキャン結果を取得する	15
<b>レジストリ資格情報を管理する</b>	<b>16</b>
新しいレジストリを追加する	16



レジストリを編集する	18
レジストリを削除する	19
次のステップ	19
<b>スキャンイメージ</b>	<b>19</b>
イメージを手動でスキャンする	20
ランタイムタブから	20
リポジトリタブから	21
イメージを自動的にスキャンする	21
<b>スキャンポリシーを管理する</b>	<b>22</b>
事前設定されたポリシー	23
デフォルトポリシー	24
事前設定されたコンプライアンスポリシー	24
ポリシー設定-Dockerfileのベストプラクティス	25
監査ポリシー-NIST 800-190	25
監査ポリシー-PCI	25
カスタマイズされたポリシー	26
ポリシーを作成する	26
ポリシーを編集する	27
ポリシーを削除する	27
ホワイトリスト/ブラックリスト	28
ポリシーの割り当てを管理する	28



ポリシーを割り当てる	28
優先順位を使用する	30
<b>ホワイトリスト ブラックリストCVEとイメージ</b>	<b>31</b>
ホワイトリスト化/ブラックリスト化されたCVEとイメージを確認する	31
CVEをホワイトリストに登録する	31
リポジトリタブから	32
イメージのホワイトリスト/ブラックリスト	34
ホワイトリスト/ブラックリストからCVEまたはイメージを削除する	35
<b>Scanning ポリシーゲート、ルール、及び、トリガー</b>	<b>37</b>
Always	37
always	37
Dockerfile	37
effective_user	38
exposed_ports	39
instruction	40
no_dockerfile_provided	41
Files	41
content_regex_match	41
name_match	42
suid_or_guid_set	43
Licenses	43

blacklist_exact_match	43
blacklist_partial_match	44
Metadata	44
attribute	45
NPMs	45
blacklisted_name_version	45
feed_data_unavailable	46
newer_version_in_feed	46
unknown_in_feeds	47
version_not_in_feeds	47
Packages	47
blacklist	48
required_package	48
verify	49
Passwd File	50
blacklist_full_entry	50
blacklist_groupids	51
blacklist_shells	51
blacklist_userids	52
blacklist_usernames	52
content_not_available	53



Ruby Gems	53
blacklist	53
feed_data_unavailable	54
newer_version_found_in_feed	54
not_found_in_feed	55
version_not_found_in_feed	55
Secret Scans	55
content_regex_checks	56
Vulnerabilities	57
package	57
stale_feed_data	58
vulnerability_data_unavailable	59
<b>スキャンアラートの管理</b>	<b>60</b>
スキャンアラートリストの管理	60
アラートを追加する	60
ランタイムアラートを作成する	61
基本パラメータ	61
Scope	62
トリガー	62
スキャンされていないイメージ	63
スキャン結果の変更	63



CVEの更新	64
通知チャンネル	64
リポジトリアラートを作成する	64
基本パラメータ	64
Registry/Repo/Tag	65
トリガー	65
分析された新しいイメージ	65
スキャン結果の変更	65
CVEの更新	66
通知チャンネル	66
アラートを編集する	67
アラートを複製する	67
アラートを削除する	67
<b>スキャン結果を確認する</b>	<b>67</b>
ランタイムビュー	68
スキャンされていないイメージ	69
スキャンイメージ	69
スキャン結果ビュー	70
<b>スキャン結果の詳細</b>	<b>70</b>
ポリシー結果ビュー	71
概要	71



過去のスキヤンの日付を選択	72
スキヤンポリシーの詳細を確認する	72
脆弱性の概要を確認する	73
脆弱性の比較	73
コンテンツの詳細を確認する	76
<b>データ保持制限を設定する</b>	<b>76</b>
<b>レポート</b>	<b>78</b>
イメージスキヤンレポート	79
概要	79
レポートを実行する	79
Query by Vulnerability	81
Query by Package	81
Query by Policy	82

# イメージスキャン

イメージスキャンを使用すると、コンテナイメージの脆弱性、シークレット、ライセンス違反などをスキャンできます。開発ビルドプロセスの一部として使用でき、コンテナレジストリに追加されたイメージを検証でき、インフラストラクチャでコンテナを実行することで使用されるイメージをスキャンできます。

## 前提条件

- ネットワークとポートの要件  
イメージスキャンには、外部の脆弱性フィードへのアクセスが必要です。最新の定義に適切にアクセスするには、[ネットワークとポートの要件](#)を参照してください。
- イメージスキャンリクエストのホワイトリストに登録されたIP  
イメージスキャンリクエストとSplunkイベント転送は両方とも18.209.200.129から発生します。Sysdigがプライベートリポジトリをスキャンできるようにするには、ファイアウォールでこのIPアドレスからのインバウンドリクエストを許可する必要があります。

## Sysdigイメージスキャンの仕組み

イメージスキャンの基本設定は簡単です。イメージが保存されているレジストリ情報を提供し、スキャンをトリガーして、結果を確認します。

動作のステップ：

- イメージの内容が分析されます。
- コンテンツレポートは、複数の脆弱性データベースに対して評価されます。
- 次に、デフォルトまたはユーザー定義のポリシーと比較されます。

- 結果は、Sysdig Secureと（該当する場合）開発者の外部CIツールの両方で報告されます。

## イメージコンテンツレポート

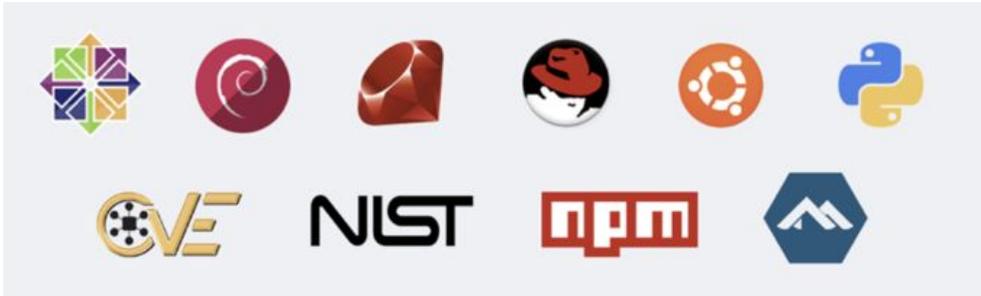
分析により、以下を含むイメージコンテンツの詳細レポートが生成されます。

- 公式OSパッケージ
- 非公式OSパッケージ
- 構成ファイル
- 資格情報ファイル
- ローカリゼーションモジュールとソフトウェア固有のインストーラー：
  - NPMを使用したJavascript
  - Python PiP
  - GEMを使用したRuby
  - .jarアーカイブを使用したJava/JVM
- イメージのメタデータと構成属性

## 使用される脆弱性データベース

Sysdig Secureは、さまざまな脆弱性データベースに対して継続的にチェックし、新しく検出されたCVEでランタイムスキャン結果を更新します。

現在のデータベースリストには次のものが含まれます。



[Centos](#) [Debian](#) [Ruby](#) [Red Hat](#) [Ubuntu](#) [Python](#)

[CVE](#) [NIST](#) [NPM](#) [Alpine](#) [NVD](#)

## ユースケース

組織として、環境内で実行される許容可能な、安全で信頼できるイメージを定義します。開発パイプラインのイメージスキャンは、セキュリティ担当者とは多少異なるフローに従います。

### コンテナ開発中のスキャン (DevOps)

開発パイプラインの一部としてイメージスキャンを使用して、ベストプラクティス、脆弱性、およびシークレットコンテンツをチェックします。

始めに：

- **レジストリの追加**：イメージにアクセスするために必要な資格情報とともに、イメージが保存されているレジストリを追加します。
- **CIツールの統合**：Jenkinsプラグインを使用するか、SysdigLabsソリューションから独自の統合を構築して、イメージスキャンを外部CIツールと統合します。
- **イメージのスキャン**：プラグインまたはCLI統合により、イメージスキャンプロセスがトリガーされます。失敗したビルドは、構成されている場合は停止されます。
- **結果の確認 (CIツール)**：開発者は、統合CIツール (Jenkins) で結果を分析できます。

(オプション：ポリシーを追加するか、ニーズに合わせてデフォルトポリシーを調整し、特定のイメージまたはタグにポリシーを割り当て、アラートと通知を構成します。)

## 実行中のコンテナのスキャン (セキュリティ担当者)

セキュリティ担当者は、イメージスキャンを使用して、実行中のコンテナ、スキャンステータス、およびイメージに新しい脆弱性が存在するかどうかを監視します。

始めに：

- **レジストリの追加**：イメージにアクセスするために必要な資格情報とともに、イメージが保存されているレジストリを追加します。
- **イメージのスキャン**：イメージスキャンをトリガーします（手動で、または自動スキャンするようにアラートを構成します）。
- **レビュー結果 (Sysdig Secure)**：セキュリティ担当者は、Sysdig SecureイメージスキャンUIでスキャン結果を分析できます。

(オプション：ポリシーを追加するか、ニーズに合わせてデフォルトポリシーを調整し、特定のイメージまたはタグにポリシーを割り当て、アラートと通知を構成します。)

### 注意

イメージスキャンには、外部の脆弱性フィードへのアクセスが必要です。最新の定義に適切にアクセスするには、ネットワークとポートの要件を参照してください。

# イメージスキャンを開発パイプラインに統合する

開発パイプラインの一部としてイメージスキャンを使用して、ベストプラクティス、脆弱性、およびシークレットコンテンツをチェックするオプションがあります。

## インラインスキャン

バージョン2.5.0以降、Sysdig Secureユーザーは、ローカルでイメージをスキャンおよび分析し、レジストリへのアクセスを提供せずにインフラストラクチャメタデータをSysdigプラットフォームに送り返すことができます。この機能は、さまざまな場合に必要になることがあります。

- イメージは自分の環境を離れません
- SaaSユーザーは、SysdigのSaaSサービスにイメージと独自コードを送信しません
- レジストリを公開する必要はありません
- イメージをより簡単に並行してスキャンできます
- レジストリに到達する前にイメージをスキャンできます。
  - レジストリのコストを削減
  - ビルドパイプラインを簡素化する

## 前提条件

- Sysdig SecureおよびSysdigインストールに接続する機能
- Dockerエンジン
- DockerHubへのアクセス
- Bash

## インラインスキャンを実装する

- スクリプトにアクセスする
- ここ([https://raw.githubusercontent.com/sysdiglabs/secure-inline-scan/master/inline\\_scan.sh](https://raw.githubusercontent.com/sysdiglabs/secure-inline-scan/master/inline_scan.sh))からinline\_scan.shスクリプトをダウンロードします。

- パラメーターと例を確認する
- GitHubのReadMeファイルには、スクリプトパラメーターとその使用方法が説明されており、詳細な例が示されています。
- 期待される出力

スキャンがトリガーされた後、コマンドラインは成功または失敗の結果メッセージを送信します。

詳細な結果分析を表示するには、Sysdig Secureダッシュボードにログインして、[Scan Result] ページを確認します。

## パイプライン統合の例

さまざまなパイプラインについて、詳細に文書化された例があります。

- Gitlab: <https://sysdig.com/blog/gitlab-ci-cd-image-scanning/>
- Githubアクション: <https://sysdig.com/blog/image-scanning-github-actions/>
- AWS Codepipeline: <https://sysdig.com/blog/image-scanning-aws-codepipeline-codebuild/>
- Azureパイプライン: <https://sysdig.com/blog/image-scanning-azure-pipelines/>
- CircleCI: <https://sysdig.com/blog/image-scanning-circleci/>

## Jenkinsと統合する

Sysdigには、SysdigイメージスキャンをJenkinsベースのビルドプロセスに統合するプラグインがあります。

### Jenkinsプラグインのインストールと構成

Sysdig Secure Jenkins Plugin(<https://plugins.jenkins.io/sysdig-secure/>)のドキュメント (jenkins.ioにあります) には次のことが記載されています。

- 前提条件
- プラグインの入手
- Jenkins UIで必要なシステム設定手順

- ビルドステップとしてのSysdig Secure Image Scanningの追加（Jenkins UIで）
- スキャンされたビルドで実行するアクションの設定（ビルドの失敗や警告の発行のタイミングなど）

## Jenkinsでスキャン結果を取得する

Sysdigプラグインは、Jenkinsビルドリストにリストされているスキャンレポートを生成します。

Sysdig Scanning Reportをクリックして、概要情報とポリシーチェックと結果のリストを表示します。



# レジストリ資格情報を管理する

Sysdig Secureがイメージをプルおよび分析するには、レジストリ資格情報が必要です。各レジストリタイプには、必要な資格情報の一意の入力フィールドがあります（例：docker.ioのユーザー名/パスワード、Google Container RegistryのJSONキー）。

## 新しいレジストリを追加する

1. **Image Scanning** モジュールから、**Registry Credentials** を選択します。
2. **Add Registry** をクリックします。

[New Registry]ページが表示されます。

Image Scan Runtime Alerts Repositories Scanning Policies Registry Credentials

### New Registry

Path: Enter registry path (e.g. docker.io)

Type: Docker V2 (dropdown menu)

Username: Docker V2 (dropdown menu)

Password: AWS ECR (dropdown menu)

Internal Registry Address: Optional: e.g. docker.registry.svc:5000, docker-registry.default.svc.cluster.local:5000

Allow Self Signed:

Use Image to Test Credentials:

3. レジストリへのパスを入力します。例えば、docker.io
4. ドロップダウンメニューからレジストリ **Type** を選択します。
5. レジストリ固有の **credentials** を選択します（選択した **Type** に基づいて）

- a. Docker V2には多くのDocker V2レジストリがあり、資格要件は異なる場合があります。

たとえば、Azure Container Registryの場合 :

- i. Admin Account

- `Username:` in the `'az acr credentials show --name <registry name>'` command result

- `Password:` The password or password2 value from the `'az acr credentials show'` command result

- ii. Service Principal

- `Username:` The service principal app id

- `Password:` The service principal password

- b. AWS ECR:

- i. AWS access key

- ii. AWS secret key

- c. Google Container Registry:

- i. JSON Key

- 6. (主にOpenShiftクラスターの場合) : 内部レジストリアドレスを追加します。

OpenShiftクラスターのイメージレジストリーを実行する推奨方法は、ローカルで実行することです。Sysdigエージェントは内部レジストリ名を検出しますが、Anchoreエンジンがイメージをプルおよびスキャンするには、内部レジストリ自体にアクセスする必要があります。

例 :

外部名 : mytestregistry.example.com

内部名 : docker-registry.default.svc : 5000

### 注意

Sysdigは内部レジストリ名を外部レジストリ名にマップするため、ランタイムリストとリポジトリリストには外部名のみが表示されます。

7. オプション：スイッチをトグルして、`Allow Self-Signed`証明書を許可します。

デフォルトでは、UIはTLS/SSL対応のレジストリからのみイメージをプルします。

（自己署名証明書または不明な認証局からの証明書でレジストリが保護されている場合）証明書を検証しないようUIに指示するには、`[Allow Self-Signed]`を切り替えます。

8. オプション：「`Test Credentials`」スイッチを切り替えて、エントリーを検証します。

有効にすると、Sysdigは入力された資格情報を使用してイメージをプルしようとします。成功すると、レジストリが保存されます。失敗した場合、エラーが表示され、資格情報またはイメージの詳細を修正できます。

有効にした場合は、`test registry path`を次の形式で入力します。

```
registry/repo:tag
```

9. 例えば、`quay.io/sysdig/agent:0.89`

10. Saveをクリックします。

## レジストリを編集する

1. `Image Scanning`モジュールから、`Registry Credentials`を選択します。
2. 既存のレジストリを選択して、`Edit`ウィンドウを開きます。

3. 必要に応じてパラメーターを更新し、[Save]をクリックします。

#### 注意

レジストリタイプは編集できません。

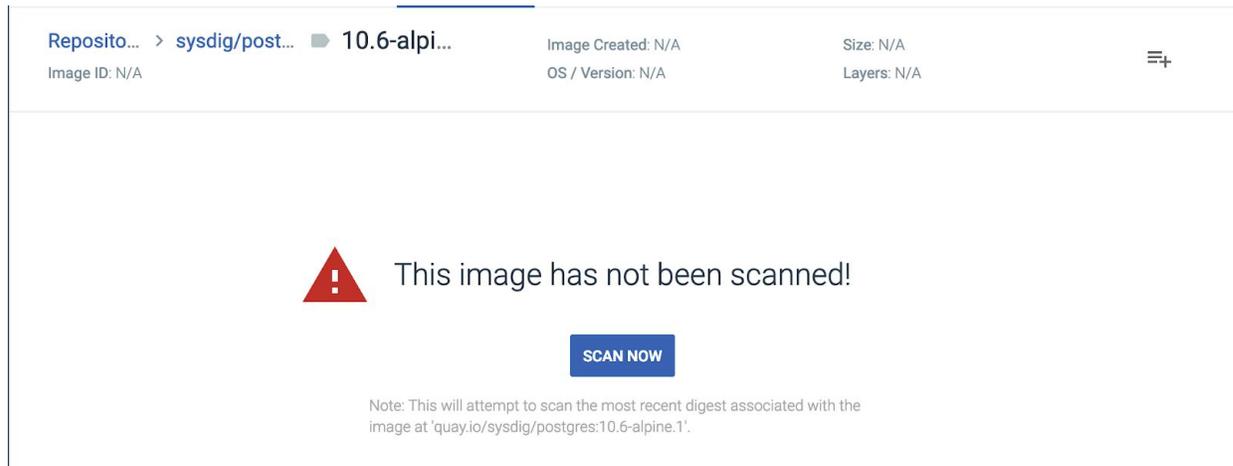
## レジストリを削除する

1. **Image Scanning** モジュールから、**Registry Credentials** を選択します。
2. 既存のレジストリを選択して、編集ウィンドウを開きます。
3. **Delete Registry** をクリックし、**Yes** をクリックして変更を確認します。

## 次のステップ

少なくとも1つのレジストリが正常に追加されると、提供されている**Default**のスキャンポリシーを利用して、イメージをスキャンし、スキャン結果を確認することができます。

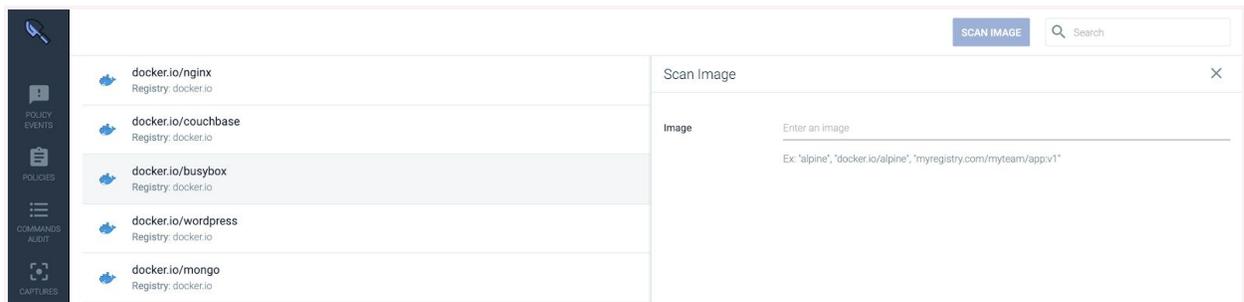




3. **Scan Now** をクリックします。

## リポジトリタブから

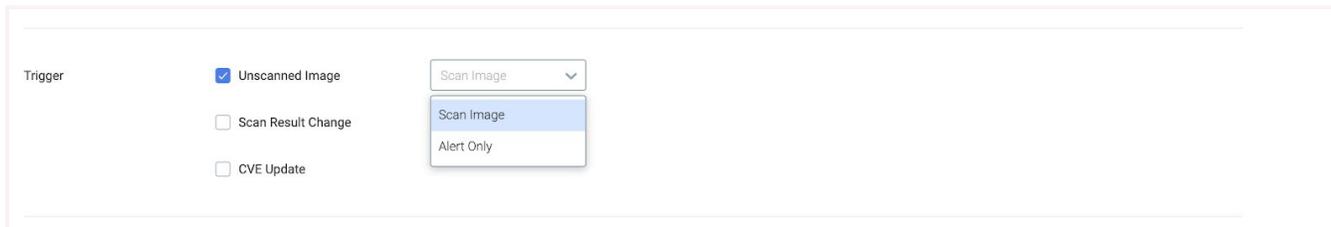
1. **Image Scanning** モジュールから、**[Repositories]** タブを選択します。
2. **[Scan Image]** をクリックします



3. イメージへのパスを定義し、**[Scan]** をクリックします。

## イメージを自動的にスキャンする

[Unscanned Image]のトリガードロップダウンメニューを[Scan Image]に設定して、スキャンされていないイメージが見つかったときに自動的にイメージスキャンをトリガーするスキャンアラートを構成します。



Trigger

- Unscanned Image
- Scan Result Change
- CVE Update

Scan Image

- Scan Image
- Alert Only

スキャンアラートの管理も参照してください。

# スキャンポリシーを管理する

イメージスキャンポリシーは、次のようないくつかのシナリオを定義します。

- ビルドプロセスが停止している可能性があります。
- 管理者は、コンテナイメージ内の潜在的なリスクについて警告を受けることがあります。

各スキャンポリシーは、ゲートとトリガーで構成されるルールで構成されます。Sysdigには、レジストリ資格情報が設定されるとすぐにスキャンを実行するために使用できるデフォルトポリシーが含まれています。

ユーザーは、利用可能なスキャンポリシーゲートおよびトリガーから追加のルールまたはポリシーを作成できます。

## 事前設定されたポリシー

Sysdigには、現状のまま、または構築するテンプレートとして使用できる4つのベースラインポリシーが用意されています。

IMAGE SCAN		Policies		<a href="#">Add Policy</a>
	<b>DefaultPolicy</b> System default policy	Policy ID: default	Rules: 8	
	<b>Default Configuration Policy - Dockerfile Best Practices</b> This policy provides out of the box rules around Dockerfile best practices. We frequently update these policies and if ...	Policy ID: dockerfile_best_practices	Rules: 10	
	<b>Default Audit Policy - NIST 800-190</b> This policy interprets NIST 800-190 controls and provides out of the box rules to detect image misconfiguration. We ...	Policy ID: nist_800-190	Rules: 14	
	<b>Default Audit Policy - PCI</b> This policy interprets PCI controls and provides out of the box rules to detect image misconfiguration. We frequently ...	Policy ID: pci	Rules: 7	

## デフォルトポリシー

このポリシーは、次のような最も一般的なイメージスキャンのケースを対象としています。

- 中および高脆弱性のチェック
- 設定アイテムのチェック（例：イメージのヘルスチェックの確認、公開ポートの禁止）
- 脆弱性フィードデータが最新であることを検証します。

このポリシーは、削除できない基本的な包括的なポリシーです。他のポリシーの割り当てが行われな  
ない場合、デフォルトのポリシーが自動的に使用されます。

### ヒント

デフォルトポリシーを編集でき、Sysdig Secureをアップグレードしても編集内容は保持されます

## 事前設定されたコンプライアンスポリシー

構成済みの他の3つのポリシーは、コンプライアンスルールを扱います。それらを使用するには、ポリシー割り当てリストに追加する必要があります。

#### 警告

事前設定されたコンプライアンスポリシーを編集する場合は、一致するルールを含む新しいポリシーを作成して編集します。

そうしないと、Sysdig Secureのアップグレード中にカスタマイズが上書きされて失われる可能性があります。

## ポリシー設定-Dockerfileのベストプラクティス

このポリシーは、許可しないなど、Dockerfileのベストプラクティスに関するすぐに使用できるルールを提供します。

- 環境変数として焼き付けられた秘密
- ルートユーザー設定
- 露出ポート
- .yumアップグレードを含む指示を実行します。

## 監査ポリシー-NIST 800-190

このポリシーは、NIST 800-190コントロールを許可しないなどのSysdig Secureスキャンニングポリシーにマップします。

- 非公式ノードまたはRubyパッケージ
- Dockerファイルに指示を追加する
- 予期された値以外のベースディストリビューションの使用

## 監査ポリシー-PCI

このポリシーは、PCI（Payment Card Industry）コントロールをSysdig Secureスキャンポリシーにマップします（脆弱性や資格情報をイメージに含めないようにするなど）。

## カスタマイズされたポリシー

事前設定されたコンプライアンススキャンポリシーを直接編集しないでください。一致するポリシーを作成し、編集します。

## ポリシーを作成する

1. **Image Scanning** モジュールから、**Scanning Policies** を選択し、**Add Policy(+)** をクリックします。

[New Policy]ページが表示されます

The screenshot shows the 'New Policy' page in the Sysdig Secure interface. The page is titled 'Image Scan' and has tabs for 'Runtime', 'Alerts', 'Repositories', 'Scanning Policies', and 'Registry Credentials'. The 'Scanning Policies' tab is active. The page contains the following fields:

- Name:** A text input field containing 'Scanning Policy'.
- Description:** A text area containing 'Description of policy'.
- Rules:** A dropdown menu with 'Select gate...' selected. The dropdown is open, showing a list of options: Always, Dockerfile, Files, Licenses, Metadata, Npms, Packages, Passwd file, and Ruby gems.

At the bottom right of the page, there are two buttons: 'Cancel' and 'Save'.

2. 新しいポリシーのNameとオプションのDescriptionを定義します。
3. Ruleを追加します。
  - a. ドロップダウンメニューからGateを選択し、Triggerを選択します。
  - b. 関連するパラメーターを設定します。（一部のトリガーでは、パラメーターを設定する必要はありません。）

各オプションの詳細については、ポリシーゲートおよびトリガーのスクリーンを参照してください。

以下の例では、packageトリガーでvulnerabilitiesゲートを使用しています。

New Policy

Name: Test Policy

Description: Description of policy

Rules: Vulnerabilities Package Package type: all; Severity: medium; Severity comparison: >= Warn

Fix available (optional): Leave blank

Package type: all

Severity: medium

Severity comparison: >=

Vendor only (optional): Leave blank

Select gate...

4. オプション：ステップ5を繰り返して、必要に応じてルールを追加します。
5. Saveをクリックします。

## ポリシーを編集する

1. Image Scanningモジュールから、Scanning Policiesを選択します。
2. リストから目的のポリシーを選択します。
3. 必要に応じてポリシールールを編集し、[Save Policy]をクリックします。

## ポリシーを削除する

1. `Image Scanning` モジュールから、`Scanning Policies` を選択します。。
2. リストから目的のポリシーを選択します。
3. `[Delete]` (ゴミ箱) アイコンをクリックし、`[Yes]` を選択して変更を確認します。

## ホワイトリスト/ブラックリスト

必要に応じて、特定のイメージまたはCVEをグローバルにホワイトリストまたはブラックリストに登録できます。ホワイトリスト|ブラックリストCVEとイメージを参照ください。これは、ポリシーの評価順序には影響しません。

## ポリシーの割り当てを管理する

スキャンに対して非常に単純な単一ポリシーのアプローチを使用しない限り、特定のポリシーを特定のレジストリ、リポジトリ、またはタグに割り当てる可能性があります。

これを行うには、`[Policy Assignments]` ページを使用します。

例えば：

- サンプルProdイメージポリシーで「Prod」タグを持つすべてのイメージを評価するには、次の割り当て (`registry/repo/tag`) を使用します：`*/*/Prod`
- サンプルGoogleポリシーを使用してgcr.ioからすべてのイメージを評価するには、割り当て (`registry/repo/tag`) を使用します：`gcr.io/*/*`

## ポリシーを割り当てる

1. `Image Scanning` モジュールから、`[Scanning Policies]` を選択し、`[+Policy Assignments]` を選択します。

以前に定義された割り当てが優先度順にリストされます。

Policy Assignments

Entries are evaluated in priority order - drag an assignment to change the priority.

+ Add Policy Assignment

Priority	Registry	Repository	Tag	Assigned Policy	Audits
1	*	redis	*	DefaultPolicy	nist_800-190 x dockerfile_best_practices x pci x X
2	docker.io	node	*	DefaultPolicy	nist_800-190 x pci x dockerfile_best_practices x X
3	*	sysdig/agent	*	DefaultPolicy	pci x nist_800-190 x X
4	*	*	*	DefaultPolicy	dockerfile_best_practices x X

2. [+Add Policy Assignment]をクリックします。

[Assignment]ページの上部に新しいエントリ行が表示されます。目的の割り当ての詳細を入力します。

- Priority** : 優先度は、割り当てられたポリシーに対する評価の順序です。新しい割り当てはそれぞれ優先度1に自動的に配置されます。ポリシー割り当てを作成して保存したら、リストの新しい位置にドラッグして優先度の順序を変更できます。優先順位の使用も参照してください。
- Registry** : 任意のレジストリドメイン（例：quay.io）。ワイルドカードがサポートされています。アスタリスク\*はレジストリを指定します。
- Repository** : 任意のリポジトリ（通常=イメージの名前）。ワイルドカードがサポートされています。アスタリスク\*は、任意のリポジトリを指定します。
- Tag** : 任意のタグ。ワイルドカードがサポートされています。アスタリスク\*は任意のタグを指定します。
- Assigned Policy** : 評価に使用するポリシーの名前。ドロップダウンメニューから選択します。

3. 保存をクリックします。

4. オプション：ドラッグハンドル（行の左側にある4つのドット）をクリックし、割り当てをリスト上の別の場所にドラッグして、優先順位を再編成します。

## 優先順位を使用する

複数のスキャンポリシーを使用する場合、Anchoreエンジンは、ポリシー割り当てリストの優先度1から開始して、トップダウンの順序でそれら进行评估します。入カイメージに一致する最初のポリシー割り当てルールが評価され、以降のすべてのルールは無視されます。したがって、優先順位は重要です。

### ヒント

たとえば、2つのポリシー定義が定義されたリストを想像してください。

```
Priority 1 Registry = quay.io Repository = sysdig/*
```

```
Priority 2 Registry = quay.io Repository = sysdig/myrepo
```

最初のルールはワイルドカードを使用するため、評価はsysdig/で始まるすべてのリポジトリに適用され、sysdig/myrepoを評価する前に停止します。

優先順位を逆にして、目的の動作を取得します。

ポリシー割り当てリストの下部には、削除できないキャッチオールエントリがあります。形式は次のとおりです。

```
registry = * repository = * tag = * assigned policy = default
```

(`assigned policy`は変更できますが、他のフィールドは編集できません。)



この行の目的は、別のポリシー評価に該当しないレジストリが、少なくともシステム設定のDefaultポリシーに対して評価されるようにすることです。

# ホワイトリスト|ブラックリストCVEとイメージ

Sysdig Secureを使用すると、ユーザーはCVEとイメージをグローバルに信頼されたまたはブラックリストに登録されたものとして定義できます。たとえば、低リスクのCVEをグローバルに承認して、より重要な修正を含むビルドに影響を与えないようにすることができます。または、特定のイメージをグローバルに承認済みまたは未承認としてマークして、常に/決してスキャンに合格しないようにすることができます。

## ヒント

ユーザー、ポート、パッケージなど、他のエンティティのブラックリストオプションは、ポリシーゲートとトリガーのスキャンにリストされています。

## ホワイトリスト化/ブラックリスト化されたCVEとイメージを確認する

ホワイトリスト/ブラックリストに登録されたCVEとイメージの現在のリストを確認するには：

1. `Image Scanning`モジュールから、`Scanning Policies`を選択します。
2. `[Whitelists and Blacklist]`ボタンをクリックします。
3. 関連するタブ（`CVE Whitelist`、`Global Trusted Images`、または`Global Blacklisted Images`）を選択します。

## CVEをホワイトリストに登録する

CVEをホワイトリストに登録するには、`[Scanning Policies]`タブと`[Repositories]`タブの2つの方法があります。

[Scanning Policies]タブから：

1. **Image Scanning**モジュールから、**Scanning Policies**を選択します。

Policy Name	Description	Policy ID	Rules
DefaultPolicy	System default policy2	default	6
old policy	No description	policy_1EESzCIA17os7o59JNOxrSBMTKq	5
qa policy	No description	policy_1EVEkts9UnOsgNwBohVGpuCSWZ5	2
new one	it does things	policy_1FXHrOT43VizwFpVeGpp85y4uHa	2
Default Configuration Policy - Dockerfile Best Practices	This policy provides out of the box rules around Dockerfile best practices. We frequently update th...	dockerfile_best_practices	10
Default Audit Policy - NIST 800-190	This policy interprets NIST 800-190 controls and provides out of the box rules to detect image mis...	nist_800-190	14
Default Audit Policy - PCI	This policy interprets PCI controls and provides out of the box rules to detect image misconfigurati...	pci	7

2. **[Whitelists and Blacklists]**をクリックします。
3. **[Add CVE]**をクリックし、各CVEをコンマ区切りリストに追加し、**[Ok]**をクリックして保存します。

リスト内の各項目は、CVE命名形式 (**CVE-YEAR-ID**) に従う必要があります。

## Whitelist CVEs

CVE-2019-155, CVE-2019-166, CVE-2019-001

Cancel

Ok

## リポジトリタブから

1. **Image Scanning** モジュールから、**[Repositories]** を選択し、リストされているリポジトリのいずれかを選択します。  
脆弱性に関連するポリシー結果がある場合、そのCVEをホワイトリストに登録できます。
2. 関連するCVEの横にある**[More Options]** (+) アイコンをクリックします。

Overview Policy Vulnerabilities Content

**!** DefaultPolicy evaluation **Failed** with **25 Stop** actions and **29 Warn** actions on April 25 2019, 2:02 pm

Evaluation Breakdown

<b>!</b> PACKAGE vulnerabilities	HIGH Vulnerability found in os package type (dpkg) - apt (fixed in: 1.4.9) - (CVE-2019-3462 - https://security-tracker.debian.org/tracker/CVE-2019-3462)	
<b>!</b> PACKAGE vulnerabilities	HIGH Vulnerability found in os package type (dpkg) - libapt-pkg5.0 (fixed in: 1.4.9) - (CVE-2019-3462 - https://security-tracker.debian.org/tracker/C	Add CVE to Global Whitelist
<b>!</b> PACKAGE vulnerabilities	HIGH Vulnerability found in os package type (dpkg) - libc6 (fixed in: 2.24-11+deb9u4) - (CVE-2017-1000408 - https://security-tracker.debian.org/tracker/CVE-2017-1000408)	
<b>!</b> PACKAGE vulnerabilities	HIGH Vulnerability found in os package type (dpkg) - libc6 (fixed in: 2.24-11+deb9u4) - (CVE-2017-15670 - https://security-tracker.debian.org/tracker/CVE-2017-15670)	
<b>!</b> PACKAGE vulnerabilities	HIGH Vulnerability found in os package type (dpkg) - libc6 (fixed in: 2.24-11+deb9u4) - (CVE-2017-15804 - https://security-tracker.debian.org/tracker/CVE-2017-15804)	
<b>!</b> PACKAGE vulnerabilities	HIGH Vulnerability found in os package type (dpkg) - libc6 (fixed in: 2.24-11+deb9u4) - (CVE-2017-16997 - https://security-tracker.debian.org/tracker/CVE-2017-16997)	

3. [Add CVE to Global Whitelist]を選択します。

これで、CVEが[[CVE Whitelist](#)]タブにリストされます。

## イメージのホワイトリスト/ブラックリスト

### 注意

イメージが「信頼できるイメージ」リストと「ブラックリストに登録されたイメージ」リストの両方に追加された場合、ブラックリストに登録されたものが優先されます。

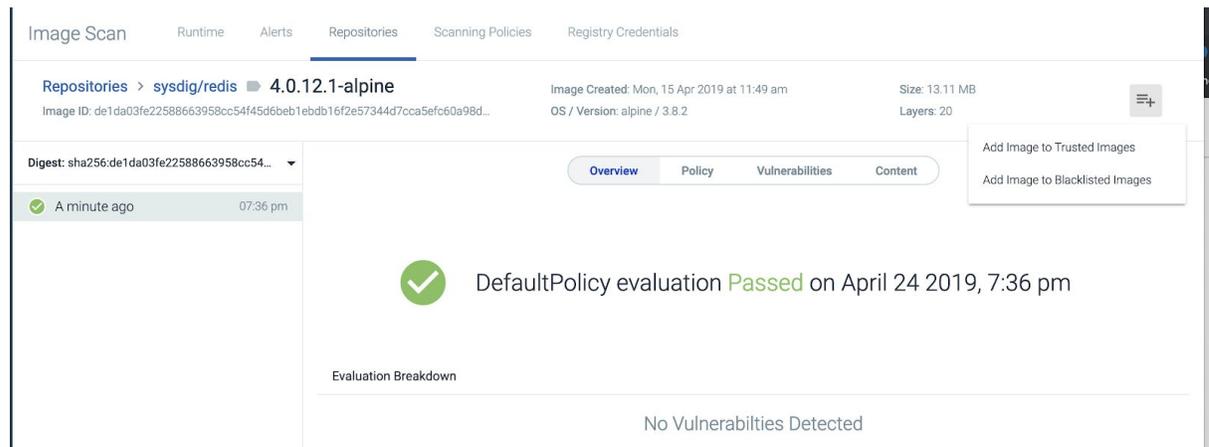
イメージをホワイトリストまたはブラックリストに登録するには、[[Scanning Policies](#)]タブと [[Repositories](#)]タブの2つの方法があります。

[[Scanning Policies](#)]タブから：

1. [Image Scanning](#)モジュールから、[Scanning Policies](#)を選択します。
2. [[Whitelists and Blacklists](#)]をクリックします。
3. 関連するタブ（[Global Trusted Images](#)、または[Global Blacklisted Images](#)）を選択し、[Add Image](#)ボタンをクリックします。
4. 各イメージをコンマ区切りリストに追加し、[[Ok](#)]をクリックします。  
タグ名は有効なASCIIでなければならず、小文字と大文字、数字、アンダースコア、ピリオド、ダッシュを含めることができます。  
タグ名はピリオドまたはダッシュで始まってはならず、最大128文字を含めることができます。

[[Repositories](#)]タブから：

1. [Image Scanning](#)モジュールから、[Repositories](#)を選択します。
2. リストから関連するリポジトリを選択し、関連するイメージを開きます。
3. ページ上部の[[More Options](#)] (+) アイコンをクリックします



1. 必要に応じて、[Add Image to Trusted Images]または[Add Image to Blacklisted Images]を選択します。
2. これで、必要に応じて、CVEが[Global Trusted Images]タブまたは[Global Blacklisted Images]タブにリストされます。

## ホワイトリスト/ブラックリストからCVEまたはイメージを削除する

さまざまなリストから1つ以上のCVEまたはイメージを削除するには：

1. Image Scanning モジュールから、Scanning Policies を選択します。
2. [Whitelists and Blacklists] をクリックします。
3. 関連するタブ (CVE Whitelist、Global Trusted Images、または Global Blacklisted Images) に移動します。
4. 関連するCVE/イメージの横にある Delete (X) アイコンをクリックします

The screenshot shows the Sysdig Image Scan interface. The top navigation bar includes 'Image Scan', 'Runtime', 'Alerts', 'Repositories', 'Scanning Policies', and 'Registry Credentials'. The 'Scanning Policies' tab is active. On the left, a sidebar lists 'Lists' with 'CVE Whitelist' selected. The main content area is titled 'CVE Whitelist' and features a search bar and an 'Add CVE' button. Below, a table lists CVEs with 'x' icons for removal.

CVE Whitelist	
Search	
<a href="#">Add CVE</a>	
CVE	
CVE-2019-155	x
CVE-2019-166	x
CVE-2019-001	x

5. **Save** をクリックします。

# Scanning ポリシーゲート、ルール、及び、トリガー

この文書はSysdig Secureポリシーバンドル内でサポートされているゲート（およびそれらのそれぞれのトリガー/パラメーター）の包括的なリストを提供します。これらのポリシーゲート、トリガー、およびパラメータを使用して、部分ファイル名のホワイトリスト/ブラックリストから、どのログインシェルが承認されるかを定義するまで、詳細なスキャンポリシーを構築できます。

この情報はまたCLIを使用して得ることができます：

```
user@host:~$ anchore-cli policy describe (--gate <gatename> (--trigger <triggername>))
```

スキャンポリシーの構築の詳細については、[Image Scanning](#)のドキュメントを参照してください。

## Always

このゲートは無条件に起動されるため、有用なテストリソースをユーザーに提供します。

### always

ポリシーに存在する場合、alwaysトリガー/ゲートは作動します。

Alwaysゲートは、イメージブラックリスト/ホワイトリストが期待どおりに機能しているかどうかをテストするのに役立ちます。



New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Always Always No parameters required Warn x

Select gate

## Dockerfile

dockerfileゲートは、ベスト・プラクティスに従わない公開ポートおよび命令について、dockerfileの内容、または提供されていない場合はdockerfileの想定内容を確認します。

ゲートは、コンテンツがdocker layer historyに基づいていることを前提としています。

### effective\_user

このトリガーは、有効なユーザーが提供されたユーザーと一致するかどうかを確認し、構成されたタイプに基づいて起動します。

パラメーター	ディスクリプション	例
type	ユーザーをホワイトリストに登録するかブラックリストに登録するかを決定します。	N/A
user	ユーザーの名前	root,docker



## New Policy

Name

Description

Rules     X

Type

Users

Select gate...

## exposed\_ports

このトリガーは、公開されているポートのセットを評価して、ホワイトリストに登録するかブラックリストに登録するかを決定します。

パラメーター	ディスクリプション	例
actual_dockerfile_only	評価が推測または推測されたdockerfileをスキップするかどうかを定義し、ユーザー提供のdockerfilesのみを評価します。デフォルト値はfalseです。	true
ports	ポート番号のカンマ区切りリスト	80,8080,8088
type	ポートをホワイトリストに登録するかブラックリストに登録するかを定義します	N/A



## New Policy

Name

Description

Rules

Actual dockerfile only (optional)

Ports

Type

Select gate...

## instruction

このトリガは、リスト内のdirectives/instructionsがdockerfile内の条件と一致するかどうかを評価します。

パラメーター	ディスクリプション	例
actual_dockerfile_only	評価が推測または推測されたdockerfileをスキップするかどうかを定義し、ユーザー提供のdockerfilesのみを評価します。デフォルト値はfalseです。	true
check	実行するチェックのタイプ	=
instruction	dockerfile instruction の確認	FROM
value	dockerfile instruction をチェックする値	scratch



New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Dockerfile, Instruction, Actual dockerfile only: true, Check: \*, Instruction: FROM, Value: scratch, Warn

Actual dockerfile only (optional): true

Check: \*

Instruction: FROM

Value (optional): scratch

Select gate...

## no\_dockerfile\_provided

イメージにdockerfileが提供されていない場合、このトリガーは作動します。このトリガーにはパラメーターは必要ありません。

New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Dockerfile, No dockerfile provided, No parameters required, Warn

Select gate...

## Files

ファイルゲートは、分析されたイメージ内のファイルを確認します。この評価はファイルの内容、名前、そしてファイルシステムの属性をカバーします。

## content\_regex\_match

このトリガーは、analyzer\_config.yamlのcontent\_searchセクションで構成された正規表現を使用して一致が見つかったファイルごとに発生します。

正規表現の値に関する詳細は、analyzer\_config.yamlファイルを参照してください。

パラメーター	ディスクリプション	例
regex_name	FILECHECK_CONTENTMATCHアナライザー・パラメーターに表示される正規表現文字列	.*password.*

New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Files | Content regex match | Regex name: .\*password.\* | Warn | x

Regex name (optional): .\*password.\*

Select gate...

## name\_match

コンテナ内のファイルの名前が提供された正規表現と一致する場合、このトリガは作動します。

このトリガーは、ポリシー評価にパフォーマンス上の影響を与えます。

パラメーター	ディスクリプション	例
regex	検索する正規表現	.*\.pem



### New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Files | Name match | **Regex: \*.pem** | Warn X

Regex: \*.pem

Select gate...

## suid\_or\_guid\_set

このトリガーは、セットユーザーID（SUID）またはセットグループID（SGID）が構成されているファイルごとに発生します。パラメータは必要ありません。

### New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Files | **Suid or guid set** | No parameters required | Warn X

Select gate...

## Licenses

このゲートは、たとえば社内のポリシーに違反しているパッケージが使用されていないことを確認するために、コンテナイメージにあるソフトウェアライセンスを確認するために使用されます。

## blacklist\_exact\_match

指定された正確なライセンスの下で配布されたパッケージがイメージに含まれている場合、このトリガーは作動します。

パラメーター	ディスクリプション	例
--------	-----------	---



licenses	ブラックリストに登録するライセンス名のコンマ区切りリスト	GPLv2+,GPL-3+,BSD-2-clause
----------	------------------------------	----------------------------

### New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Licenses: Blacklist exact match Licenses: GPLv2+GPL-3+BSD-2-clause Warn

Select gate: [dropdown]

## blacklist\_partial\_match

提供された部分文字列を含むライセンスに基づいて配布されたパッケージがイメージに含まれている場合、このトリガーは作動します。

パラメーター	ディスクリプション	例
licenses	ライセンスをブラックリストに入れる文字列のコンマ区切りリスト	LGPL,BSD

### New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Licenses: Blacklist partial match Licenses: LGPL,BSD Warn

Select gate: [dropdown]

## Metadata

このゲートは、サイズ、オペレーティングシステム、アーキテクチャなどのイメージメタデータを確認します。

## attribute

名前付きイメージメタデータ値が指定の条件に一致すると、属性トリガーが発生します。

パラメーター	ディスクリプション	例
attribute	属性名の確認	size
check	評価のために実行する操作	>
value	評価に使用する値	1073741824

### New Policy

Name:

Description:

Rules:

Attribute:

Check:

Value (optional):

Select gate:

## NPMs

NPMゲートは、NPMパッケージがインストールされているすべてのイメージを確認します。

### blacklisted\_name\_version

評価されたイメージに、ブラックリストに記載されているNPMパッケージがインストールされている場合は、名前順、またはオプションで名前とバージョン順でトリガされます。



パラメーター	ディスクリプション	例
name	ブラックリストに載っているNPMパッケージの名前	time_diff
version	ブラックリストに登録するNPMパッケージの特定のバージョン	0.2.9

#### New Policy

Name: My Scanning Policy

Description: Description of policy

Rules: Npmjs | Blacklisted name version | Name: time\_diff, Version: 0.2.9 | Warn

Select gate

### feed\_data\_unavailable

エンジンがNPMデータフィードにアクセスできない場合、このトリガーは作動します。パラメータは必要ありません。

#### New Policy

Name: My Scanning Policy

Description: Description of policy

Rules: Npmjs | Feed data unavailable | No parameters required | Stop

Select gate

### newer\_version\_in\_feed

NPMデータ・フィードにパッケージの新しいバージョンがリストされている場合、このトリガーは作動します。パラメータは必要ありません。



### New Policy

Name

Description

Rules     X

## unknown\_in\_feeds

インストールされているNPMが公式のNPMデータベースにない場合、このトリガーは作動します。パラメータは必要ありません。

New Policy

Name

Description

Rules     X

## version\_not\_in\_feeds

NPMバージョンが公式のNPMフィードに有効なバージョンとしてリストされていない場合、このトリガーは作動します。パラメータは必要ありません。

New Policy

Name

Description

Rules     X

## Packages

パッケージゲートは、イメージ内のすべてのパッケージをレビューし、名前、バージョン、およびホワイトリスト/ブラックリストに登録されたパッケージを確認します。

### blacklist

このトリガーは、名前、または名前とバージョンのいずれかによってブラックリストに登録されているパッケージがイメージに含まれている場合に発生します。

パラメーター	ディスクリプション	例
name	ブラックリストに載っているパッケージの名前	openssh-server
version	ブラックリストに載せるべきパッケージの正確なバージョン	1.0.1

#### New Policy

Name: My Scanning Policy

Description: Description of policy

Rules: Packages | Blacklist | Name: openssh-server, Version: 1.0.1 | Stop

Name: openssh-server  
Version (optional): 1.0.1

Select gate...

### required\_package

指定されたパッケージ/バージョンがイメージに見つからない場合は、required\_packageトリガーが発生します。

パラメーター	ディスクリプション	例
--------	-----------	---

name	必要なパッケージの名前	libssl
version	必要なパッケージのバージョン	1.10.3rc3
version_match_type	トリガーが正確なパッケージとバージョン（厳密）、または単にパッケージのバージョン（最小）のどちらを必要とするかを定義します。これはバージョンが定義されている場合にのみ関係します。	exact

### New Policy

Name:

Description:

Rules:

Name:

Version (optional):

Version match type (optional):

## verify

このトリガーは、イメージ内のパッケージデータベースに対するパッケージの整合性を確認し、指定されたディレクトリのすべてまたは定義済みリストのいずれかに含まれるコンテンツの変更または削除を試みます。

パラメーター	ディスクリプション	例
check	チェックが不足しているパッケージ、変更されたパッケージ、またはすべてに焦点を合わせるべきかどうかを定義します。	changed
only_directories	チェックが制限されるべきディレクトリのリストを定義します	/usr,/var/lib



only_packages	検証する必要があるパッケージのリストを定義します	libssl,openssl
---------------	--------------------------	----------------

**New Policy**

Name:

Description:

Rules:

## Passwd File

このゲートは、ブラックリストに載っているユーザー、グループ、およびシェルなどの/etc/passwdを確認します。

### blacklist\_full\_entry

パスワード全体が/etc/passwdファイルに見つかり、このトリガーは作動します。

パラメーター	ディスクリプション	例
entry	/etc/passwdに一致する完全なエントリ	ftp:x:14:50:FTP User:/var/ftp:/sbin/nologin



New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules:
 

- Passwd file
- Blacklist full entry
- Entry: ftp:x:14:50:FTP User:/var/ftp/sbin/nologin
- Stop

Entry: TP User:/var/ftp/sbin/tpologin

Select gate...

## blacklist\_groupids

指定されたグループIDが/etc/passwdファイルに見つかった場合、このトリガーは作動します。

パラメーター	ディスクリプション	例
group_ids	トリガーを作動させるグループIDのカンマ区切りの数値リスト	999,20

New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules:
 

- Passwd file
- Blacklist groupids
- Group ids: 999,20
- Warn

Group ids: 999,20

Select gate...

## blacklist\_shells

指定されたログインシェルが/etc/passwdファイル内の任意のユーザーの下に見つかった場合、このトリガーは作動します。

パラメーター	ディスクリプション	例
--------	-----------	---



shells	ブラックリストへのシェルコマンドのリスト	/bin/bash,/bin/zsh
--------	----------------------	--------------------

### New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules:
 

- Passwd file
- Blacklist shells
- Shells: /bin/bash,/bin/zsh (Warn)

Shells: /bin/bash,/bin/zsh

Select gate...

## blacklist\_userids

指定されたユーザーIDが/etc/passwdに存在する場合、このトリガーは作動します。

パラメーター	ディスクリプション	例
user_ids	ブラックリストへのユーザーIDの数値、カンマ区切りリスト	0,1

### New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules:
 

- Passwd file
- Blacklist userids
- User ids: 0,1 (Warn)

User ids: 0,1

Select gate...

## blacklist\_usernames

指定されたユーザー名が/etc/passwdファイルに見つかった場合、blacklist\_usernamesトリガーは作動します。



パラメーター	ディスクリプション	例
user_names	ブラックリストに入れるユーザー名のカンマ区切りリスト	daemon,ftp

### New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Password file | Blacklist usernames | User names: daemon,ftp | Warn

User names: daemon,ftp

Select gate...

## content\_not\_available

/etc/passwdファイルがイメージに存在しない場合、content\_not\_availableトリガーは作動します。パラメータは必要ありません。

### New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Password file | Content not available | No parameters required | Warn

Select gate...

## Ruby Gems

Ruby Gemsゲートは、開発者が正式なGEMデータベースからの正式なパッケージを使用していることを保証し、サポートされなくなったバージョンのパッケージを使用していないことを保証します。

## blacklist

設定された名前とバージョンに一致するGEMパッケージが評価されたイメージで見つかった場合、ブラックリストトリガーは作動します。

パラメーター	ディスクリプション	例
name	gemの名前	time_diff
version	ブラックリスト gemのバージョン	0.2.9

New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Ruby gems | Blacklist | Name: time\_diff, Version: 0.2.9 | Stop

Name: time\_diff  
Version (optional): 0.2.9

Select gate...

## feed\_data\_unavailable

SysdigがGEMデータフィードにアクセスできない場合、このトリガーは作動します。パラメータは必要ありません。

New Policy

Name: Test Scanning Policy

Description: Description of policy

Rules: Ruby gems | Feed data unavailable | No parameters required | Stop

Select gate...

## newer\_version\_found\_in\_feed

インストールされているGEMパッケージが最新バージョンではない場合、このトリガーは作動します。パラメータは必要ありません。

#### New Policy

Name	<input type="text" value="Test Scanning Policy"/>
Description	<input type="text" value="Description of policy"/>
Rules	<input type="text" value="Ruby gems"/> <input type="text" value="Newer version found in..."/> <input type="text" value="No parameters required"/> <input type="text" value="Stop"/> X
	<input type="text" value="Select gate..."/>

### not\_found\_in\_feed

インストールされたGEMが公式GEMデータベースにない場合、このトリガーは作動します。パラメータは必要ありません。

#### New Policy

Name	<input type="text" value="Test Scanning Policy"/>
Description	<input type="text" value="Description of policy"/>
Rules	<input type="text" value="Ruby gems"/> <input type="text" value="Not found in feed"/> <input type="text" value="No parameters required"/> <input type="text" value="Stop"/> X
	<input type="text" value="Select gate..."/>

### version\_not\_found\_in\_feed

GEMが正式なGEMフィードに有効/サポートされているバージョンとしてリストされていない場合、このトリガーは作動します。パラメータは必要ありません。



## New Policy

Name

Description

Rules

## Secret Scans

シークレットスキャンは、設定された正規表現に基づいて、イメージが侵害された場合に利用できる可能性のあるシークレットがイメージに焼き付けられているかどうかを判断します。

### content\_regex\_checks

content\_regex\_checksは、コンテンツ検索アナライザーが構成済みの名前付き正規表現との一致を検出した場合にトリガーを起動します。一致は、content\_regex\_nameとfilename\_regexのいずれかが設定されている場合は、それらによってフィルタリングされます。

content\_regex\_nameは、analyzer\_config.yamlのsecret\_searchセクション値にする必要があります。

パラメーター	ディスクリプション	例
content_regex_name	変数/コンテンツの名前。イメージで見つかった場合にトリガーを作動させます  デフォルトで使用可能な名前はAWS_ACCESS_KEY、AWS_SECRET_KEY、PRIV_KEY、DOCKER_AUTH、およびAPI_KEYです。	AWS_ACCESS_KEY
filename_regex	content_regex_nameの存在について分析する必要があるファイルをフィルタリングします	/etc/.*



**New Policy**

Name:

Description:

Rules: Secret scans Content regex checks Content regex name: AWS\_ACCESS\_KEY, Filename regex: /etc/\* Stop X

Content regex name (optional):

Filename regex (optional):

Select gate:

## Vulnerabilities

CVE /脆弱性チェックを使用して、含まれているパッケージに設定レベルを超える脆弱性がないこと、指定された期間より古いこと、またはデータが利用できない場合を確認できます。

### package

イメージ内の脆弱性が設定された比較基準に一致すると、パッケージトリガーが作動します。以下の表は、利用可能なパラメータと基準をまとめたものです。

パラメーター	ディスクリプション	例
fix_available	存在する場合、脆弱性レコードの修正プログラムの可用性はパラメータの値と一致する必要があります	true
package_type	特定のタイプのパッケージ	all
severity	脆弱性の深刻度	high
severity_comparison	セキュリティ評価のために実行する比較のタイプ	>



vendor_only	trueの場合、このCVEに対して利用可能な修正は、「Won't be addressed by the vendor」として明示的にマークされません	true
-------------	---	------

**New Policy**

Name:

Description:

Rules: Vulnerabilities | Package | Fix available: true; Package type: all; Severity: high; Severity comparison: >; Vendor only: true | Stop X

Fix available (optional):

Package type:

Severity:

Severity comparison:

Vendor only (optional):

Select gate:

## stale\_feed\_data

CVEデータが指定されたウィンドウよりも古い場合、stale\_feed\_dataトリガーが発生します。

パラメーター	ディスクリプション	例
max_days_since_sync	トリガーが作動するまでの同期データの日数を決定します	10

**New Policy**

Name:

Description:

Rules: Vulnerabilities | Stale feed data | Max days since sync: 10 | Stop X

Max days since sync:

Select gate:

## vulnerability\_data\_unavailable

利用可能な脆弱性データがない場合は、脆弱性データ有効化トリガーが作動します。このトリガーにはパラメーターは必要ありません。

The screenshot shows a 'New Policy' configuration form. It has three main sections: 'Name', 'Description', and 'Rules'.  
- 'Name': A text input field containing 'Test Scanning Policy'.  
- 'Description': A larger text input field containing 'Description of policy'.  
- 'Rules': A list of rules. The first rule is for 'Vulnerabilities'. It has a trigger dropdown set to 'Vulnerability data unav...', a parameter field with 'No parameters required', and a severity dropdown set to 'Stop'. There is a close button 'x' next to the severity dropdown. Below this rule, there is a 'Select gate' dropdown menu. To the right of the rule list, there is a separate dropdown menu with 'Stop' and 'Warn' options.

## スキャンアラートの管理

すべてのSysdigアラートと同様に、イメージスキャンアラートは、インフラストラクチャで問題が発生したときにユーザーに通知するように構成できます。スキャンアラートは、リポジトリ内の静的イメージまたは実行中（実行時）イメージに対して作成できます。スキャンアラートは、スキャンされていないイメージが環境に追加されたとき、イメージがポリシー評価に失敗したとき、スキャン結果が変更されたとき、またはCVEが更新されたときに焦点を合わせます。

ユーザーがアラートを実装する場合の例：

- 処理する3つの異なるイメージの新しいCVE更新があるかどうかを知りたい
- 組織全体で使用されているdocker hubからの共通イメージのいずれかが変更されたポリシーステータスを持っている場合、通知を受けたい

## スキャンアラートリストの管理

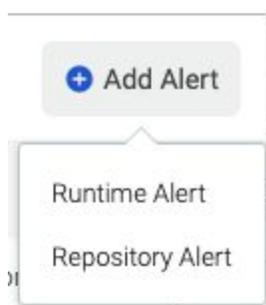
Image Scanning モジュールから、[Alerts] タブを選択します。スキャン警告リストが表示されます。

Alert Name	Description	Repository	Channel	Actions
Test	test description	reg/test/tag	test, Email Channel (test@sysdig.com)	Duplicate Alert, Delete Alert

ここから、既存のアラートを検索し、アラートを作成、複製、または削除できます。

## アラートを追加する

1. 新しいアラートを作成するには : `Image Scanning` モジュールから、`[Alerts]` タブを選択し、`[Add Alert]` をクリックします。
2. `Runtime` または `Repository` のアラートタイプを選択します。



3. 適切な `[New Alert]` ページに入力します。

## ランタイムアラートを作成する

ランタイムアラートを使用して、実行中のイメージをスキャンし、ポリシー違反、ステータス変更、またはスキャンされていないイメージが環境に追加された場合に通知をトリガーします。アラートパ

ラメータを入力し、[Save]をクリックします。

The screenshot shows the 'New Runtime Alert' configuration page. The left sidebar contains navigation icons for Policy Events, Policies, Commands Audit, Captures, Compliance, and Image Scan. The main content area has the following fields:

- Alert Type:** Runtime
- Name:** Alert Name
- Description:** Alert Description
- Scope:** Everywhere
- Trigger:**
  - Unscanned Image
  - Scan Result Change (dropdown: Pass > Fail)
  - CVE Update (dropdown: Pass > Fail, Any Change)
- Notification Channels:** Select notification channel...

## 基本パラメータ

名前とオプションの説明を入力します。

## Scope

**Entire Infrastructure**を使用するか、より狭いスコープを定義します。

Alert Type: Runtime

Name: Updates about redis & scanning-api service

Description: Alert Description

---

Scope:

- kubernetes.namespace.name in sysdigcloud x AND X
- kubernetes.deployment.name in sysdigcloud-redis x sysdigcloud-scanning-api x AND X
- Select a label

Clear All

---

Trigger:

- Unscanned Image
- Scan Result Change Any Change
- CVE Update

## トリガー

### スキャンされていないイメージ

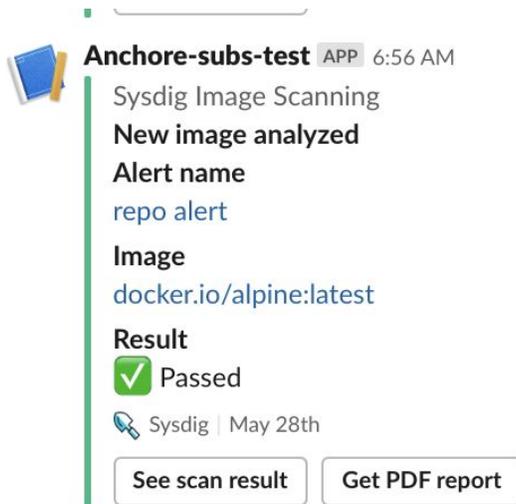
- **Scan**: ボックスをチェックし、スキャンを選択して、スコープ内で検出されたイメージを自動的にスキャンします。  
イメージのスキャンもご覧ください。
- **Alert Only**: 警告を送信しますが、イメージを自動的にスキャンしません。

### スキャン結果の変更

- **Pass/Fail**: 以前に合格したイメージがポリシー評価に失敗したときに通知を受けるには、このオプションを選択します。
- **Any Change**: このオプションを選択すると、以前にスキャンしたイメージ結果に変更があったときに通知されます。

[スキャン結果の変更]がオンになっており、通知チャンネルが構成されている場合、アラートが送信されることに注意してください。チャンネルが設定されていない場合、何も起こりません。

たとえば、次のイメージは「Any Change」が設定されたときにトリガーされたSlack通知を示しています。



## CVEの更新

実行中のイメージに脆弱性が追加、更新、または削除されるたびに通知されるようにするには、このオプションを選択します。

## 通知チャンネル

[+ Add Channel]をクリックして、アラート通知に使用する構成済みの通知チャンネル（電子メールなど）を選択します。

Sysdig Secure環境に通知チャンネルがまだ定義されていない場合は、通知チャンネルのセットアップを参照してください。

## リポジトリアラートを作成する

リポジトリアラートを使用して、リポジトリ内の静的イメージをスキャンし、ポリシー違反、ステータス変更、または環境に新しいイメージが追加された場合に通知をトリガーします。アラートパラ

メータを入力し、[Save]をクリックします。

The screenshot shows the 'New Repository Alert' configuration page. At the top right, there are 'Cancel' and 'Save' buttons. The form is divided into several sections:

- Alert Type:** Set to 'Repository'.
- Name:** 'Production Backend Fail/Vulnerability Update'.
- Description:** 'Alert Description'.
- Registry/Repo/Tag:** Three input fields containing 'gcr.io', 'production-backend', and '\*'.
- Trigger:** Three options are listed: 'New Image Analyzed' (unchecked), 'Scan Result Change' (checked with a dropdown menu set to 'Pass > Fail'), and 'CVE Update' (checked).
- Notification Channels:** A dropdown menu is set to 'Select notification channel...'. Below it, 'Email Channel (test@sysdig.com)' is checked and has a close button (x).

## 基本パラメータ

名前とオプションの説明を入力します。

### Registry/Repo/Tag

アラートで考慮するレジストリスコープを入力します。ワイルドカード\*がサポートされています。レジストリまたはレポジトリにワイルドカードが使用されている場合、唯一のアラートオプションは[新しいイメージの分析]になります。

## トリガー

### 分析された新しいイメージ

結果に関係なく、新しいイメージが分析されるたびに警告するボックスをオンにします。

### スキャン結果の変更

- **Pass/Fail** : 以前に合格したイメージがポリシー評価に失敗したときに通知を受けるには、このオプションを選択します。

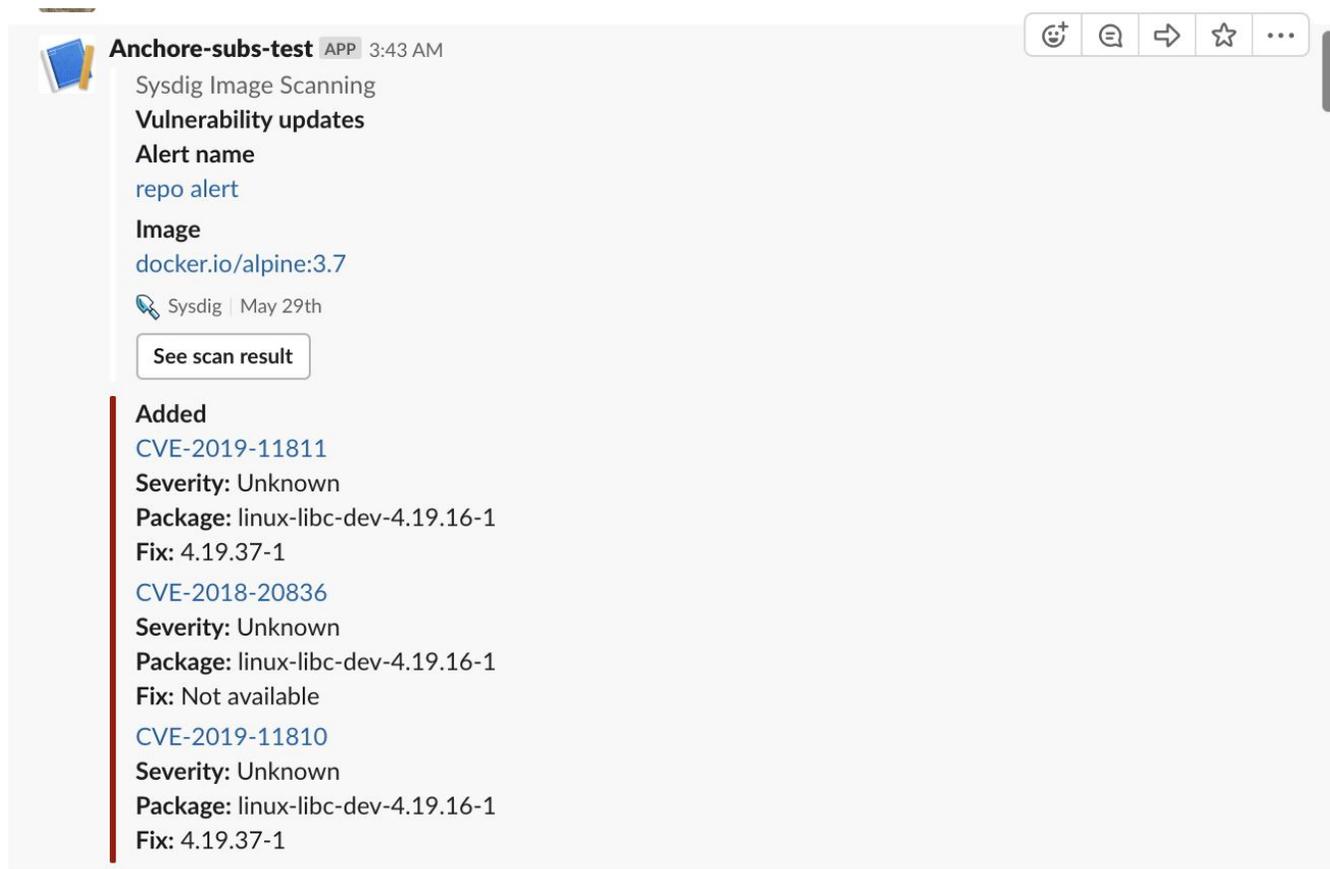
- **Any Change** : このオプションを選択すると、以前にスキャンしたイメージ結果に変更があったときに通知されます。

[スキャン結果の変更]がオンになっており、通知チャンネルが構成されている場合、アラートが送信されることに注意してください。チャンネルが設定されていない場合、何も起こりません。

## CVEの更新

このオプションを選択すると、リポジトリアラートスコープ内のイメージに脆弱性が追加、更新、または削除されるたびに通知されます。

たとえば、次のイメージは、「CVE Update」が設定されたときにトリガーされたSlack通知を示しています。



**Anchore-sub-test** APP 3:43 AM

Sysdig Image Scanning  
**Vulnerability updates**  
Alert name  
repo alert  
Image  
docker.io/alpine:3.7  
Sysdig | May 29th

See scan result

**Added**  
CVE-2019-11811  
Severity: Unknown  
Package: linux-libc-dev-4.19.16-1  
Fix: 4.19.37-1  
CVE-2018-20836  
Severity: Unknown  
Package: linux-libc-dev-4.19.16-1  
Fix: Not available  
CVE-2019-11810  
Severity: Unknown  
Package: linux-libc-dev-4.19.16-1  
Fix: 4.19.37-1

## 通知チャンネル

[+ Add Channel]をクリックして、アラート通知に使用する設定済みの通知チャンネル（電子メールなど）を選択します。

Sysdig Secure環境に通知チャンネルがまだ定義されていない場合は、通知チャンネルのセットアップを参照してください。

## アラートを編集する

- **Image Scanning**モジュールから、[Alerts]タブを選択します。
- リストから目的のアラートを選択します。
- 必要に応じてアラートトリガー、スコープ、および通知チャンネルを編集し、[Save]をクリックします。

## アラートを複製する

- **Image Scanning**モジュールから、[Alerts]タブを選択します。
- リストから目的のアラートを選択します。
- [More]（3つのドット）アイコンをクリックし、ドロップダウンから[Duplicate Alert]をクリックし、[はい]をクリックして確認します。

## アラートを削除する

- **Image Scanning**モジュールから、[Alerts]タブを選択します。。
- リストから目的のアラートを選択します。
- [More]（3つのドット）アイコンをクリックし、ドロップダウンから[Delete Alert]をクリックしてから、[Yes]をクリックして確認します。

## スキャン結果を確認する

スキャン用のビルド環境をセットアップし（該当する場合）、目的のレジストリを追加し、スキャンを手動でトリガーするか、自動的にスキャンするようにアラートを構成すると、イメージスキャンレポートが生成されます。

スキャン結果にアクセスするにはさまざまな方法があります。

- 外部（開発者向け）： Jenkinsなどの外部継続的統合（CI）ツールから。
- 内部（セキュリティ担当者向け）： Sysdig Secureのイメージスキャンモジュールの[Runtime]タブまたは[Scan Results]タブ（旧称「Repositories」）から

### 注意

データ保持設定を使用して、スキャン結果リストの管理に役立てることができます。詳細については、データ保持制限の設定を参照してください。

（オンプレミスインストール用ではSysdig Platformバージョン3.2.0から利用可能です。）

## ランタイムビュー

**Runtime**は、過去1時間に環境で実行されたイメージに関する常に更新されたレポートを提供します。

The screenshot displays the 'Image Scan' interface with the 'Runtime' tab selected. On the left, a sidebar shows 'Entire Infrastructure' expanded, listing namespaces: dev, default, Prod Test, kube-system, and prod. The main area features a donut chart showing 37 failing images (28.3%). Below the chart, a summary states: 'There are 2 Unscanned and 1 Scan In Progress images, with 37 Failing, 12 Warning and 80 Passing images, running as containers in "Entire Infrastructure" over the last hour.'

Category	Image Name	Image ID	Image Created	Running Containers
Unscanned Images	library/nginx - Latest	7372399839f3dfa323423fafadsf23342342	10 days ago	12
	library/ubuntu - 1-alpine	7372399839f3dfa323423fafadsf23342342	10 days ago	12
Scan In Progress	library/redis - Latest	7372399839f3dfa323423fafadsf23342342	2 hours ago	12
Scanned Images	library/redis - Latest	7372399839f3dfa323423fafadsf23342342	2 hours ago	12
	library/postgres - 9.3.22	7372399839f3dfa323423fafadsf23342342	10 days ago	12
	library/mongo - jessie	7372399839f3dfa323423fafadsf23342342	10 days ago	12
	library/centos - 7	7372399839f3dfa323423fafadsf23342342	10 days ago	12

左側の列で、**Entire Infrastructure**を表示するか、ネームスペースにドリルダウンします。

右の列のレポートには、**Unscanned**イメージと**Scanned**イメージが一覧表示されます。[スキャン結果の詳細]ビューにドリルダウンできます。

### スキャンされていないイメージ

スキャンを手動でトリガーするには、スキャンされていないイメージを選択します。

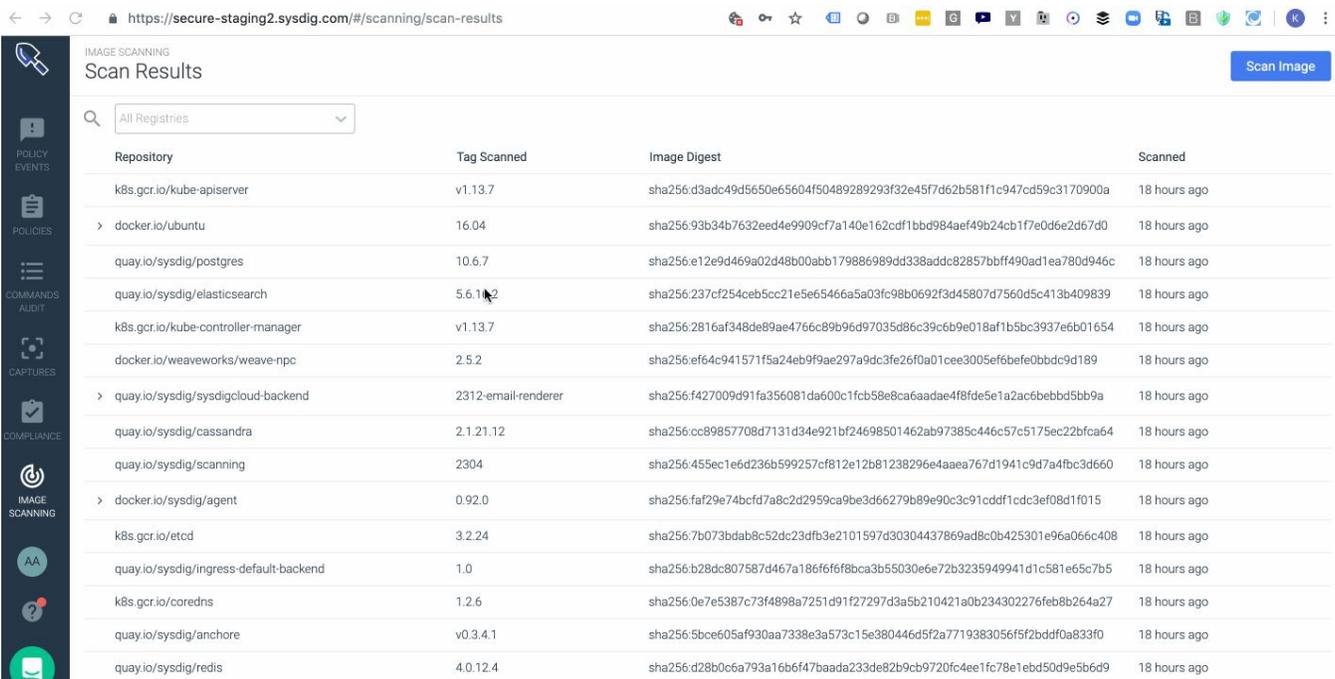
## スキャンイメージ

スキャンされたイメージを選択して、詳細にドリルダウンします：[Summary](#) ページ、[Policy](#)の詳細、[Vulnerability](#)の詳細、[Content](#)違反（ライセンスなど）。

## スキャン結果ビュー

[Scan Results](#) リストを使用して、スキャン結果のリストを表示します。

- 実行されていないものを含む特定のイメージを検索する
- イメージがデプロイされている場所に基づくフィルター
- さまざまなリポジトリを簡単に参照/展開して、評価されたimage : tagsとその結果を確認します。



Repository	Tag Scanned	Image Digest	Scanned
k8s.gcr.io/kube-apiserver	v1.13.7	sha256:d3adc49d5650e65604f50489289293f32e45f7d62b581f1c947cd59c3170900a	18 hours ago
> docker.io/ubuntu	16.04	sha256:93b34b7632eed4e9909cf7a140e162cdf1bbd984aef49b24cb1f7e0d6e2d67d0	18 hours ago
quay.io/sysdig/postgres	10.6.7	sha256:e12e9d469a02d48b00abb179886989dd338addc82857bfff490ad1ea780d946c	18 hours ago
quay.io/sysdig/elasticsearch	5.6.10_2	sha256:237cf254ceb5cc21e5e65466a5a03fc98b0692f3d45807d7560d5c413b409839	18 hours ago
k8s.gcr.io/kube-controller-manager	v1.13.7	sha256:2816af348de89ae4766c89b96d97035d86c39c6b9e018af1b5bc3937e6b01654	18 hours ago
docker.io/weaveworks/weave-npc	2.5.2	sha256:ef64c941571f5a24eb9f9ae297a9dc3fe26f0a01cee3005ef6befe0bbdc9d189	18 hours ago
> quay.io/sysdig/sysdigcloud-backend	2312-email-renderer	sha256:f427009d91fa356081da600c1fcb58e8ca6aadae4f8fde5e1a2ac6bebbd5bb9a	18 hours ago
quay.io/sysdig/cassandra	2.1.21.12	sha256:cc89857708d7131d34e921bf24698501462ab97385c446c57c5175ec22bfca64	18 hours ago
quay.io/sysdig/scanning	2304	sha256:455ec1e6d236b599257cf812e12b81238296e4aaea767d1941c9d7a4fbc3d660	18 hours ago
> docker.io/sysdig/agent	0.92.0	sha256:faf29e74bcfd7a8c2d2959ca9be3d66279b89e90c3c91cddf1cdc3ef08d1f015	18 hours ago
k8s.gcr.io/etcd	3.2.24	sha256:7b073bdab8c52dc23dfb3e2101597d30304437869ad8c0b425301e96a066c408	18 hours ago
quay.io/sysdig/nginx-default-backend	1.0	sha256:b28dc807587d467a186f6f6f8bca3b55030e6e72b3235949941d1c581e65c7b5	18 hours ago
k8s.gcr.io/coredns	1.2.6	sha256:0e7e5387c73f4898a7251d91f27297d3a5b210421a0b234302276feb8b264a27	18 hours ago
quay.io/sysdig/anchore	v0.3.4.1	sha256:5bce605af930aa7338e3a573c15e380446d5f2a7719383056f5f2bddf0a833f0	18 hours ago
quay.io/sysdig/redis	4.0.12.4	sha256:d28b0c6a793a16b6f47baada233de82b9cb9720fc4ee1fc78e1ebd50d9e5b6d9	18 hours ago

リスト上のスキャンされたイメージにドリルダウンすると、スキャン結果の詳細ビューはランタイムパネルからのものと同じです。

## スキャン結果の詳細

[Scan Results]リストにドリルダウンすると、詳細メニューに脆弱性とポリシー違反のデータが一目でわかるさまざまな方法が表示されます。

- ポリシー概要ビュー
- 脆弱性の概要
- コンテンツの概要

これらの要約は以下を提供します。

- 特定のイメージが失敗した理由の解析しやすいビュー
- どのルールが最も警告および停止アクションを生成したか
- 配置されたさまざまな監査ポリシーに対してイメージがどのように実行されたかの概要
- 重大度の高いCVEをフィルタリングし、利用可能な修正があるものを確認する機能

ポリシーの概要をPDFに、脆弱性の概要をCSVファイルにダウンロードすることもできます。

## ポリシー結果ビュー

### 概要

スキャン結果の詳細のランディングページは、Policy Summaryビューです。

できる事：

- スキャン状態の鳥瞰図を取得する
- 別のスキャン日付を選択してください
- 詳細ページへのドリルダウン

- [Download as PDF]をクリックして、基礎となるすべてのCVEを含む完全なレポートを取得します

docker.io/vulnerables/web-dwaa latest Add to List

Image Digest: sha256:dae203fe11646a86937bf04db0079adef295f426da68a92b40e3b...  
 Image Created: February 25, 2019 11:10 PM | Size: 170.10 MB  
 Image ID: ab0d83586b6e8799bb549ab91914402e47e3bcc7eea0c5cdf43755d56150cc5a | OS / Version: 9 | Layers: 8

---

June 9, 2019 11:04 PM Download PDF

**Summary**

✘ **101** **6** **782** OS Vulnerabilities  608  
 FAILED STOPS WARNS VULs Non-OS Vulnerabilities 0

**Breakdown**

	STOPS	WARNS
Default Policy	101	1
vulnerabilities: package	101	0
dockerfile: effective_user	0	1
Default Configuration Policy - Dockerfile Best Practices	0	5
dockerfile: instruction	0	4
dockerfile: effective_user	0	1

## 過去のスキャンの日付を選択

ドロップダウンから、分析するスキャンの日付を選択します。

## スキャンポリシーの詳細を確認する

リストされたポリシーを選択して、評価でトリガーされたSTOPおよびWARNアクションの詳細を確認し、

影響を受ける基本的なルールも同様です。

docker.io/vulnerables/web-dvwa latest

Image Digest: sha256:dae203fe11646a86937bf04db0079adef295f426da68a92b40e3b... Image Created: February 25, 2019 11:10 PM Size: 170.10 MB

Image ID: ab0d83586b6e8799bb549ab91914402e47e3bcc7eea0c5cdf43755d56150cc6a OS / Version: 9 Layers: 8

June 9, 2019 11:04 PM

Default Configuration Policy - Dockerfile Best Practices

Evaluation Rules

dockerfile: instruction

- WARN Dockerfile directive 'RUN' check 'like' matched against '\*apt-get upgrade\*' for line '/bin/sh -c apt-get update && apt-get upgrade -y && DEBIAN\_FRONTEND=noninteractive apt-get install -y debconf-utils && echo mariadb-server mysql-server/root\_password password vulnerables | debconf-set-selections && echo mariadb-server mysql-server/root\_password.again password vulnerables | debconf-set-selections && DEBIAN\_FRONTEND=noninteractive apt-get install -y apache2 mariadb-server php php-mysql php-gd php-pear php-gd && apt-get clean && rm -rf /var/lib/apt/lists/\*'
- WARN Dockerfile directive 'HEALTHCHECK' not found, matching condition 'not\_exists' check
- WARN Dockerfile directive 'USER' not found, matching condition 'not\_exists' check
- WARN Dockerfile directive 'ADD' check 'exists' matched against '\*' for line 'file:a71e077a42995a68ffe4834d85cfe26af4ea12aa8ed43decc03cc487124b170 in /'

dockerfile: effective\_user

- WARN User root found as effective user, which is explicitly not allowed list

## 脆弱性の概要を確認する

確認するオペレーティングシステム関連または非オペレーティングシステム関連の脆弱性の概要を選択します。

できる事：

- 脆弱性ステータスの鳥瞰図を取得する
- 完全な詳細を取得するには、CVE番号をクリックします
- 重大度による検索またはフィルター
- 異なるタグを選択して、それらの間の脆弱性の比較を確認します
- [Download CSV]をクリックして、脆弱性データをCSVファイルとして取得します

IMAGE SCANNING  
Scan Results > docker.io/nginx 1.17.8 - 1/21/2020

Image Digest sha256:401ff5d136d690b2eef61055aab7b74bc2ed89114fb5e423963249db7ac0188e Image Scanned January 21, 2020 11:38 PM Size 128.13 MB  
Image ID 5ad3bd0e67a9c542210a21a3c72f56ef6387c9b774c2506d239e655a2593ed0 Distro / Version debian / 10 Layers 3

January 30, 2020 11:34 AM

Operating System

Vulnerabilities	Severity	Fix	Package	Package Name	Package Path	Package	Version	PE	Feed	Feed
CVE-2010-4052	Negligible	None	libc6-2.28-10	libc6	None	dpkg	1.16.1 - 1/24/2020			
CVE-2010-4051	Negligible	None	libc6-2.28-10	libc6	None	dpkg	latest - 1/9/2020			
CVE-2011-3374	Negligible	None	libapt-pkg5.0-1.8.2	libapt-pkg5.0	None	dpkg	latest - 12/28/2019			
CVE-2017-18018	Negligible	None	coreutils-8.30-3	coreutils	None	dpkg	latest - 11/22/2019		vulnerabilities	deb
CVE-2018-6829	Negligible	None	libgrypt20-1.8.4-5	libgrypt20	None	dpkg	latest - 1/9/2019		vulnerabilities	deb
CVE-2010-4756	Negligible	None	libc6-2.28-10	libc6	None	dpkg	1.8.2	None	vulnerabilities	deb
CVE-2019-9192	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	8.30.3	None	vulnerabilities	deb
CVE-2019-1010024	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	1.8.4-5	None	vulnerabilities	deb
CVE-2019-1010022	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2019-1010022	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2019-1010022	Negligible	None	libc6-2.28-10	libc6	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2010-4051	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2010-4052	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2018-20796	Negligible	None	libc6-2.28-10	libc6	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2018-20796	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2010-4756	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2019-1010023	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2019-1010023	Negligible	None	libc6-2.28-10	libc6	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2019-1010025	Negligible	None	libc-bin-2.28-10	libc-bin	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2019-1010024	Negligible	None	libc6-2.28-10	libc6	None	dpkg	2.28-10	None	vulnerabilities	deb
CVE-2013-0340	Negligible	None	libexpat1-2.2.6-2+deb10u1	libexpat1	None	dpkg	2.2.6-2+deb10u1	None	vulnerabilities	deb

Download CSV

docker.io/nginx.pdf

Show All

## 脆弱性の比較

脆弱性の比較により、ユーザーは同じレポジトリ内の2つの異なるタグを比較して、バージョンXでバージョンYと比較して新しい脆弱性または修正された脆弱性を確認できます。

これにより、開発者は最新のイメージを以前のバージョンと簡単に比較して、対処された脆弱性と新規の脆弱性を簡単に報告できます。

- ページ上部のドロップダウンメニューから、タグによる脆弱性スキャン結果を選択します。
- 右側の[Compare to]ドロップダウンから別のタグを選択します。
- 比較レポートが表示され、TOTAL、NEW、FIXED、およびバージョン間の残りのSHAREDによる脆弱性が強調表示されます。見出しをクリックして、詳細な脆弱性リストを確認します。



IMAGE SCANNING

Scan Results > docker.io/mysql 5.7 - 11/13/2019 ➡ Add to List

Image Digest sha256:ba2eda1bf1249bd7e7160fa6c446d03b3261f75d3de91bc2125967d39db1525c Image Scanned November 13, 2019 11:42 AM Size 422.05 MB  
Image ID cd3ed0dff7e89f4330db6eaca2a127423df03be229d04a7e9f614fb0849121f Distro / Version debian / 9 Layers 11

January 29, 2020 9:44 PM  Critical High Medium Low Negligible Unknown Has fix Compare '5.7 - 11/13/2019' to 5.6.22 - 4/10/2019 X | v

**Operating System** Download CSV

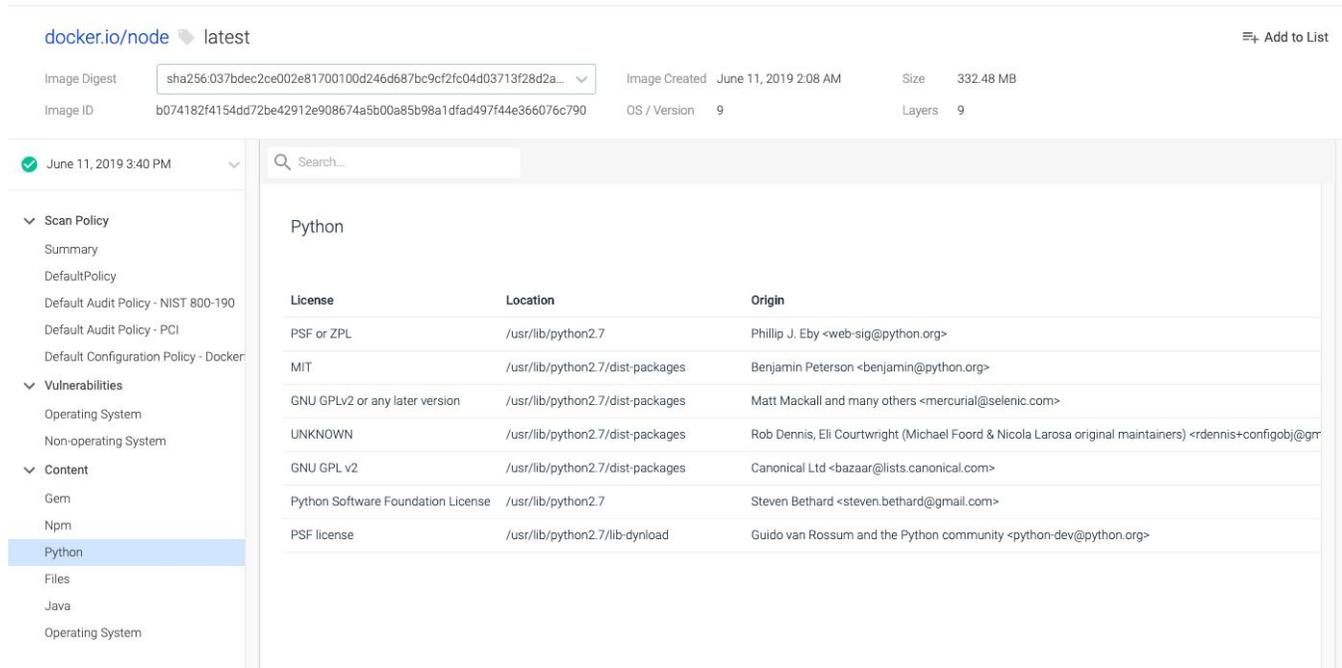
26 total 0 new in 5.7 - 11/13/2019 26 fixed in 5.7 - 11/13/2019 0 shared

Found In	Vulnerabilities	Severity	Fix	Package	Package Name	Package Path	Package Type
5.6.22 - 4/10/2019	CVE-2015-8778	High	2.13-38+deb7u10	libc6-2.13-38+deb7u7	libc6	None	dpkg
5.6.22 - 4/10/2019	CVE-2015-1472	High	2.13-38+deb7u8	libc6-2.13-38+deb7u7	libc6	None	dpkg
5.6.22 - 4/10/2019	CVE-2014-9402	High	2.13-38+deb7u8	libc6-2.13-38+deb7u7	libc6	None	dpkg
5.6.22 - 4/10/2019	CVE-2014-4043	High	2.13-38+deb7u8	libc6-2.13-38+deb7u7	libc6	None	dpkg
5.6.22 - 4/10/2019	CVE-2015-0860	High	1.16.17	dpkg-1.16.15	dpkg	None	dpkg
5.6.22 - 4/10/2019	CVE-2018-6913	High	5.14.2-21+deb7u6	perl-base-5.14.2-21+deb7u2	perl-base	None	dpkg
5.6.22 - 4/10/2019	CVE-2018-6913	High	5.14.2-21+deb7u6	perl-5.14.2-21+deb7u2	perl	None	dpkg
5.6.22 - 4/10/2019	CVE-2017-1000366	High	2.13-38+deb7u12	multiarch-support-2.13-38+deb7u7	multiarch-support	None	dpkg
5.6.22 - 4/10/2019	CVE-2015-8778	High	2.13-38+deb7u10	multiarch-support-2.13-38+deb7u7	multiarch-support	None	dpkg
5.6.22 - 4/10/2019	CVE-2014-9402	High	2.13-38+deb7u8	multiarch-support-2.13-38+deb7u7	multiarch-support	None	dpkg
5.6.22 - 4/10/2019	CVE-2018-6913	High	5.14.2-21+deb7u6	perl-modules-5.14.2-21+deb7u2	perl-modules	None	dpkg
5.6.22 - 4/10/2019	CVE-2015-8779	High	2.13-38+deb7u10	libc6-2.13-38+deb7u7	libc6	None	dpkg
5.6.22 - 4/10/2019	CVE-2014-4043	High	2.13-38+deb7u8	multiarch-support-2.13-38+deb7u7	multiarch-support	None	dpkg
5.6.22 - 4/10/2019	CVE-2017-1000366	High	2.13-38+deb7u12	libc6-2.13-38+deb7u7	libc6	None	dpkg
5.6.22 - 4/10/2019	CVE-2015-1472	High	2.13-38+deb7u8	multiarch-support-2.13-38+deb7u7	multiarch-support	None	dpkg
5.6.22 - 4/10/2019	CVE-2014-4043	High	2.13-38+deb7u8	libc-bin-2.13-38+deb7u7	libc-bin	None	dpkg
5.6.22 - 4/10/2019	CVE-2015-8779	High	2.13-38+deb7u10	multiarch-support-2.13-38+deb7u7	multiarch-support	None	dpkg
5.6.22 - 4/10/2019	CVE-2014-9402	High	2.13-38+deb7u8	libc-bin-2.13-38+deb7u7	libc-bin	None	dpkg
5.6.22 - 4/10/2019	CVE-2016-1238	High	5.14.2-21+deb7u4	perl-5.14.2-21+deb7u2	perl	None	dpkg
5.6.22 - 4/10/2019	CVE-2015-1472	High	2.13-38+deb7u8	libc-bin-2.13-38+deb7u7	libc-bin	None	dpkg

[Load More...](#)

## コンテンツの詳細を確認する

node、ruby、python、java、OSパッケージ、およびコンテナ内のファイルをナビゲートして、特定のパッケージまたはファイルに関する詳細を検索します。



The screenshot shows the Docker Hub interface for the 'node:latest' image. The main content area displays a search for 'Python' and a table of license information for Python packages. The table has three columns: License, Location, and Origin. The left sidebar shows a navigation menu with 'Python' selected.

License	Location	Origin
PSF or ZPL	/usr/lib/python2.7	Phillip J. Eby <web-sig@python.org>
MIT	/usr/lib/python2.7/dist-packages	Benjamin Peterson <benjamin@python.org>
GNU GPLv2 or any later version	/usr/lib/python2.7/dist-packages	Matt Mackall and many others <mercurial@selenic.com>
UNKNOWN	/usr/lib/python2.7/dist-packages	Rob Dennis, Eli Courtwright (Michael Foord & Nicola Larosa original maintainers) <rdennis+configobj@gm
GNU GPL v2	/usr/lib/python2.7/dist-packages	Canonical Ltd <bazaar@lists.canonical.com>
Python Software Foundation License	/usr/lib/python2.7	Steven Bethard <steven.bethard@gmail.com>
PSF license	/usr/lib/python2.7/lib-dynload	Guido van Rossum and the Python community <python-dev@python.org>

# データ保持制限を設定する

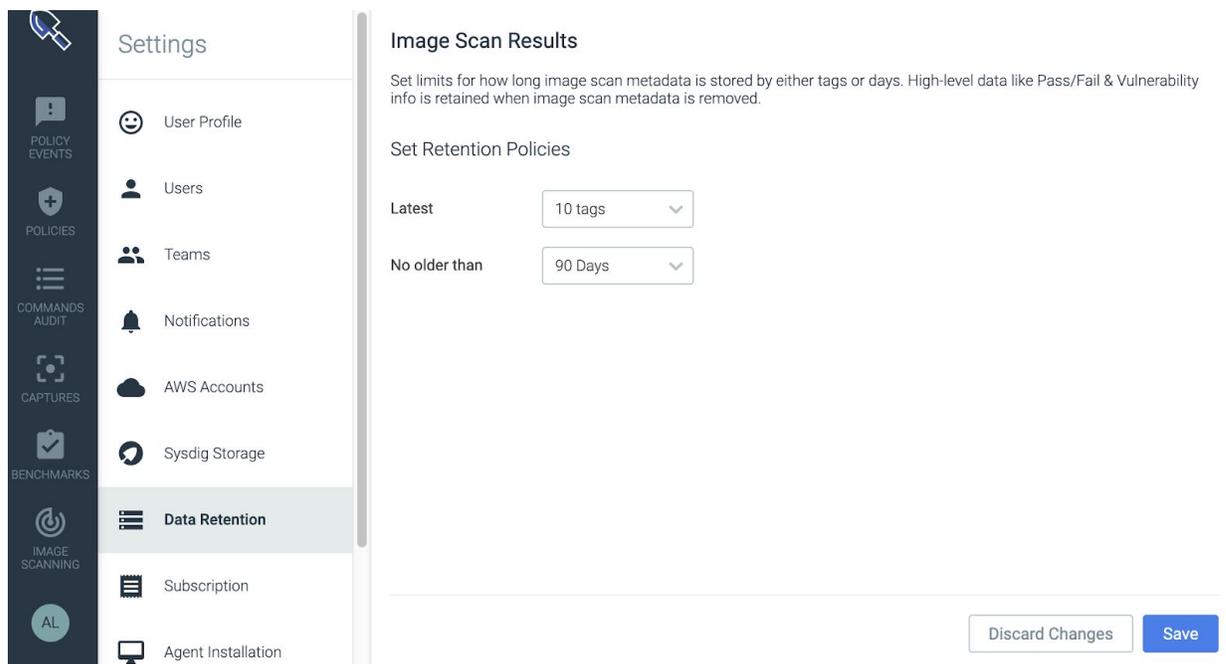
Sysdig Secure管理者は、次の目的で、イメージスキャン結果のデータ保持制限を設定できます。

- ユーザーを混乱させる古いイメージを保存しない
- Sysdigアプリケーションのパフォーマンスを改善する
- ストレージコストを削減します。

制限は、タグまたは日付で定義できます。

デフォルトのデータ保持制限を調整するには：

1. 管理者としてSysdig Secureにログインし、ナビゲーションバーのチームスイッチャーボタンから[Settings]メニューを選択します。
2. データ保持を選択します。



The screenshot shows the Sysdig Secure Settings interface. On the left is a dark sidebar with navigation icons and labels: POLICY EVENTS, POLICIES, COMMANDS AUDIT, CAPTURES, BENCHMARKS, IMAGE SCANNING, and AL. The main content area is titled 'Settings' and lists various configuration options: User Profile, Users, Teams, Notifications, AWS Accounts, Sysdig Storage, Data Retention (highlighted), Subscription, and Agent Installation. The 'Data Retention' section is expanded to show 'Image Scan Results' settings. It includes a descriptive text: 'Set limits for how long image scan metadata is stored by either tags or days. High-level data like Pass/Fail & Vulnerability info is retained when image scan metadata is removed.' Below this, there are two dropdown menus: 'Latest' set to '10 tags' and 'No older than' set to '90 Days'. At the bottom right of the settings panel are 'Discard Changes' and 'Save' buttons.

3. 制限を設定します。

- 最新：スキャン結果を保持する必要があるバージョンを（タグで）いくつ定義するか。
- 以下よりも古い：スキャン結果を保持する日数（30/60/90）を設定します。

これらの設定は、Sysdig Secure UIのスキャン結果ビューに影響します。



# レポート

## イメージスキャンレポート

### 概要

この機能を有効にするには、Sysdigサポートにお問い合わせください

レポート機能により、ユーザーは静的スコープまたはランタイムスコープに対してスキャンのコンテンツを照会し、イメージのリスク、露出、またはコンポーネントを示すレポートを生成できます。

ユースケースには次のものが含まれます。

- 新しいCVEが発表されました。そのCVEにさらされている私の米国東部クラスターで実行中のすべてのイメージを見つけてみましょう
- タグprodがあり、30日以上前に修正された脆弱性があるGoogle Container Registry内のすべてのイメージを表示する
- billingネームスペースで実行されている修正を含む重大度の高い脆弱性を持つすべてのイメージを表示する

## レポートを実行する

1. [Scanning]>[Reports]を選択します。

レポートインターフェイスが表示されます。

IMAGE SCANNING  
Reports **BETA**

Type: Vulnerability | Package | Policy

Scope: Static | Registry | Repository | Tag

Condition: +

Run | Reset

2. 適切なクエリパラメータを選択し、[Run]をクリックします。

- **Type** : 表示される列と、レポート出力のフィルタリングに使用できる条件を変更します
  - 脆弱性脆弱性ID、重大度、修正、パッケージ名などに基づいて脆弱性のリストを取得します。
  - パッケージ
  - ポリシー
- **Scope** : このレポートの一部で照会されているイメージ
  - **Static** : レジストリコンテキストに基づいてイメージを評価します。サンプルProdイメージポリシーで「Prod」タグを持つすべてのイメージを評価するには、次の割り当て (registry/repo/tag) を使用します。\*/\*/Prod
- **Runtime** : ランタイムコンテナ、クラウドプロバイダー、Kubernetesなどのオーケストレーターから公開されたラベルに基づいてイメージを評価します
- **Condition** : レポート結果をさらにフィルター処理して、意味のある結果を生成する方法。クエリタイプの内訳ごとに詳細が表示されます。

3. オプション : [Download CSV]をクリックして、レポートをキャプチャします。

## 注意

レポートを正常に生成するには、少なくとも1つの条件を選択するか、リポジトリスコープを追加する必要があります。

## Query by Vulnerability

このレポートは、静的またはランタイムスコープ内のイメージ内のパッケージにマッピングされた脆弱性の行を返します。下のイメージでは、特定の脆弱性（CVE-2017-8831）を検索すると、CVEを含む環境でアクティブに実行されている2つのイメージが表示されます。

利用可能な条件フィールドは次のとおりです。

- Vuln ID
- Severity
- Fix Available?
- Package Name
- Package Version
- Age

## Query by Package

このレポートには、パッケージのバージョンがある環境でアクティブに実行されているすべてのイメージが表示されます。また、複数のイメージが同じパッケージ名バージョンを実行しているかどうか、および関連するCVEがあるかどうかを示します。

利用可能な条件フィールドは次のとおりです。

- Package Name
- Package Version



## Query by Policy

ポリシーレポートには、発生したすべてのポリシー評価、成功したか失敗したか、およびイメージが成功したか失敗した可能性がある理由が表示されます。合格または不合格の理由には、ホワイトリスト、ブラックリスト、または単に標準的なポリシー評価が含まれます。

- Evaluation Results (Pass/Fail)
- Reason (Whitelist, Blacklist, Error, Policy Evaluation)
- Age

