

# Aqua CSPの紹介

---

## Aqua CSPとは

---

Aqua CSP(Container Security Platform)は、コンテナとクラウドネイティブアプリケーションのために開発された、フルライフサイクルなセキュリティソリューションです。

## コンテナ時代のセキュリティ

顧客ニーズの変化にすばやく対応するため、DevOpsサイクル(開発→テスト→リリース→運用→...)が必要になってきました。コンテナ技術を採用することにより開発リリースサイクルが加速する事が可能です。

しかし、現在発展中のコンテナ技術はいくつかのセキュリティリスクを抱えています。

- DockerHubなどの公開コンテナイメージの問題
  - 既知の脆弱性が放置されている可能性
  - マルウェアが仕込まれている可能性(crypto miningなど)
- 自社開発コンテナイメージ
  - 意図しない動作(情報の送信、fork bombなど)を行う可能性
- 承認されたコンテナイメージ以外を実行
  - 操作ミスや、想定外の運用
- シークレット管理
  - 第三者が作成した秘密鍵やパスワードがコンテナイメージに含まれている
- 未知の脆弱性・攻撃への対応

## シフトレフトという考え方

脆弱性のあるコンテナをもとに開発を行ったまま運用段階へ入ってしまうと、脆弱性に対するリカバリーコストが高くなります。そのため、テストより前の構築段階での開発者によるセキュリティ対策、シフトレフトが必要になってきます。

このように、DevOpsにSecurityのプロセスを加えることをDevSecOpsと呼びます。

## DevSecOpsの自動化

DevSecOpsを進めるためにはプロセスの自動化を行い、省力的にライフサイクルを回す必要があります。

- CI/CDパイプライン全体を守る
  - セキュリティの自動化でアプリのデリバリーを早める
- コンテナ全般のセキュリティ

- イメージの不変(イミュータブル)の徹底
- ホワイトリストによる制御、異常な挙動の検知
- マイクロサービスレベルのファイアウォールによるアクセス制御
- 状況の可視化とコンプライアンス対応
- クラウドロックインを防止
  - プラットフォームに依存しない
  - ハイブリッドクラウド、クラウド移動が可能

Aqua CSPは、このライフサイクルのセキュリティを支援するソリューションです。

## Aqua CSPが提供するセキュリティ

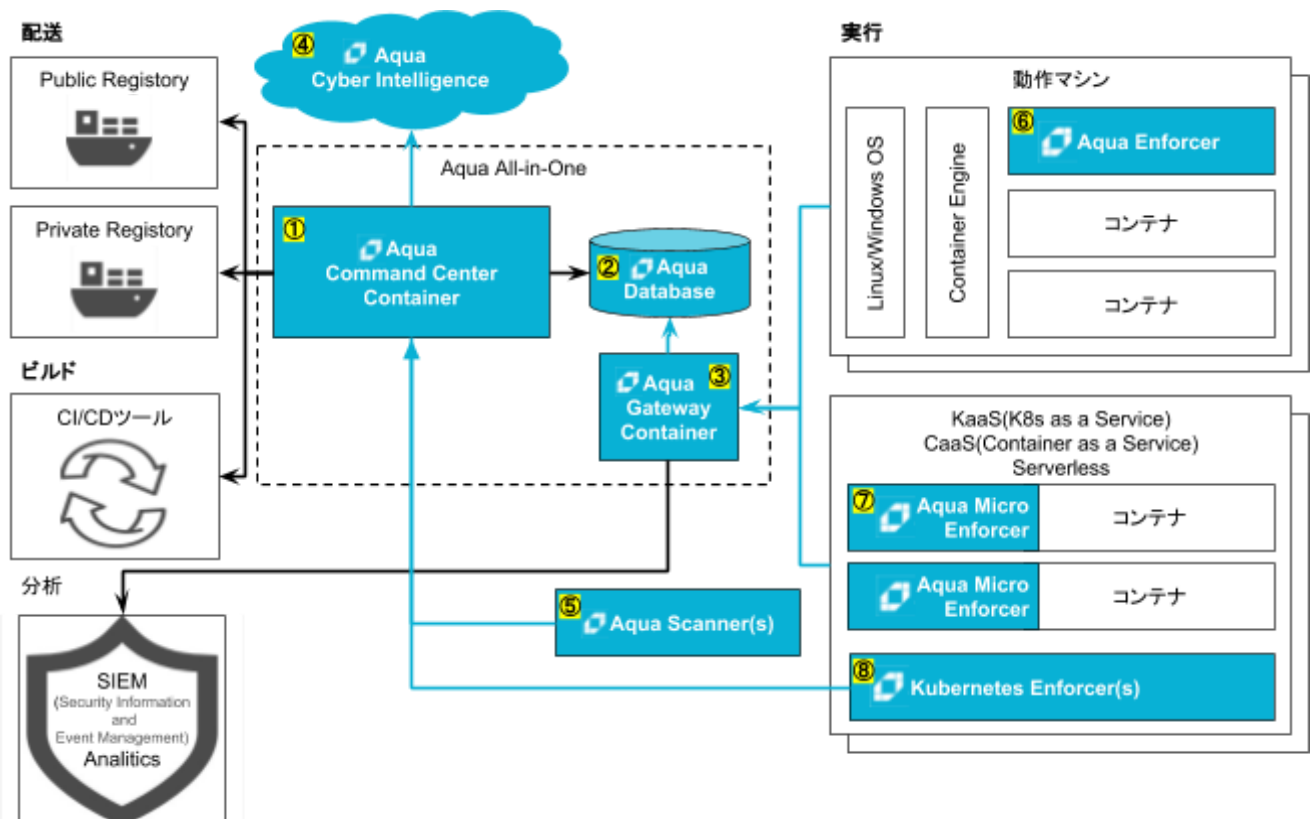
- Image Scan(イメージスキャン)  
イメージ内に含まれる、既知の脆弱性(CVE)やマルウェア、ハードコーディングされたシークレットなど、セキュリティ上のリスクを検知します
- Image Assurance(イメージ保証)  
イメージスキャン結果を元に、イメージからのコンテナ実行の許可/不許可などをポリシー定義できます。
- Runtime Policy(ランタイムポリシー)  
実行中のコンテナを監視し、ポリシーにしたがってコンテナの動作を制御/制限します。また機械学習によりポリシーを生成することも可能です
- Container Firewall(コンテナファイアウォール)  
コンテナのネットワーク接続を視覚化、また、コンテナ単位で接続の許可/拒否をルール設定できます
- Secrets(シークレット管理)
- コンテナに対して、セキュアな方法でパスワードやSSHキーなどのシークレットをデリバリーできます
- CI/CD Integration(CI/CDツール統合)  
多くのCI/CDツールと統合が可能で、開発者がビルド時にイメージスキャンすることが可能になります。これにより、初期段階でのリスクの修正が可能となります
- Compliance(コンプライアンス対応)  
リスクを一覧化し、対応方法などを含む詳細レポートを生成します。また、イメージやコンテナで発生した各種セキュリティイベントも収集されます。

## Aqua CSPのコンポーネント構成

AquaCSPは、以下のようなコンポーネントで構成されます。

1. Aqua Server(Aqua Command Center Container)  
Aqua CSPの中枢を司るコンポーネント  
WebUI、イメージスキャン、外部ツールとの連携を行う

2. Aqua Database  
Aqua CSPの設定やセキュリティ監査結果などを保存する
3. Aqua Gateway Container  
Aqua Serverと、Aqua Enforcer間の通信を行う  
外部SIEMツールとの連携を行う
4. Aqua Cyber Intelligence  
Aqua社が提供するサイバーインテリジェンスナレッジベース  
脆弱性情報、マルウェア情報などのセキュリティ関連情報を管理する
5. Aqua Scanners  
イメージスキャンを行い、結果をAqua Serverに送信する
6. Aqua Enforcer  
コンテナ動作マシン内で動作するのランタイムセキュリティモニタ  
AquaServerで設定したポリシーに沿ってコンテナの実行制御を行う
7. Aqua Micro Enforcer  
コンテナ内で動作するランタイムセキュリティモニタ  
Aqua Enforcerが動作していない環境で使用する
8. Aqua Kubernetes Enforcer  
Kubernetesのためにイメージ保証機能を提供する



# Aqua CSPのインストール

## Docker.io へのインストール

ここでは、UbuntuのDocker.ioの環境へ、Aqua CSPをインストールする手順を紹介いたします。

Aqua CSPのインストールには、Aqua 社からライセンスを入手する必要があります。ユーザID、パスワード、Aqua CSPのLicenseを用意しましょう。

以下の作業をDockerが動作するマシンで実行します。

1. Ubuntu(16.04LTS)へ、docker.io をインストールします。

```
$ sudo apt update
$ sudo apt upgrade -y
$ sudo apt install -y docker.io
$ sudo usermod -aG docker $USER
```

2. docker グループを有効化するため、ログインし直します
3. Aqua社提供のレジストリにdocker loginしてから、Aqua CSPのall-in-one イメージをpullします

```
$ echo <AQUA _PASSWORD> | docker login registry.aquasec.com \
-u <AQUA _USERNAME> --password-stdin
$ docker pull registry.aquasec.com/all-in-one:3.5
```

4. Aqua CSPのall-in-one イメージをユーザ権限で起動します。

```
$ docker run --users=host -d -p 8080:8080 -p 3622:3622 --restart=always \
-v /var/run/docker.sock:/var/run/docker.sock \
-v /var/lib/aqua-db/data:/var/lib/postgresql/data \
registry.aquasec.com/all-in-one:3.5
```

## Aqua CSPの設定

1. Aqua CSP画面をブラウザで開きます。DockerをインストールしたマシンのIPアドレスの8080ポートに、Webブラウザで接続してください。
2. 初期画面で、管理者アカウント(administrator)のパスワードを入力します

**aqua**

## Welcome to Aqua Container Security Platform

To setup your Aqua Container Security system, you must create an administrator account. Enter the password for this account (which is called "administrator"). Once setup is done, you are automatically logged in to the system as the administrator, where you can create additional users and start protecting your container environments.

Username  
administrator

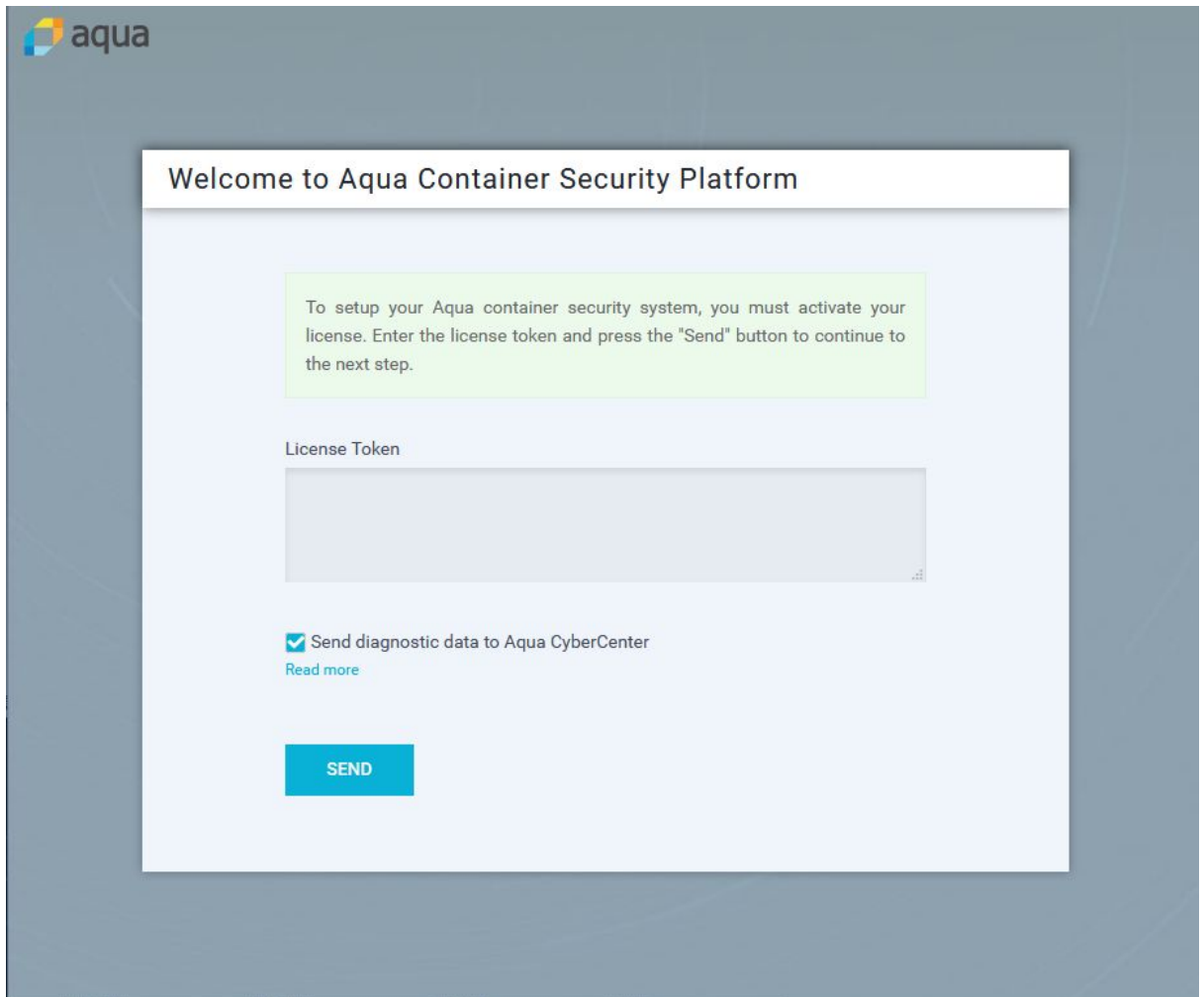
Password

Confirm Password

Keep me signed in for 30 days

**LOGIN**

### 3. Aqua CSP のLicense Tokenを入力します



4. Aqua CSPの初期画面が開きます
5. Enforcersを選んで、右端の : をクリックし、「copy install Comand」でコマンドをコピーします

The screenshot shows the Aqua Enforcers management interface. The sidebar on the left contains navigation options: Dashboard, Images, Workloads, Services, Audit, ADMINISTRATION, Policies, Secrets, Enforcers (highlighted), Compliance, and System. The main content area is titled 'Enforcers' and includes a search filter for group names. Below the filter is a table with the following columns: Name, Security Issues, Mode, Connected, Type, OS, and Orchestrator. A single row is visible for the 'default' group, which has 0 security issues, is in 'Audit Only' mode, and is not connected. A context menu is open over the 'default' row, showing options: Edit Group, Copy Install Command, Previous, 1, and Next.

6. Ubuntuのコンソールで、コピーしたコマンドを実行します。このとき、ユーザ権限で動作させるので、「--users=host」をオプションに追加するのを忘れないでください

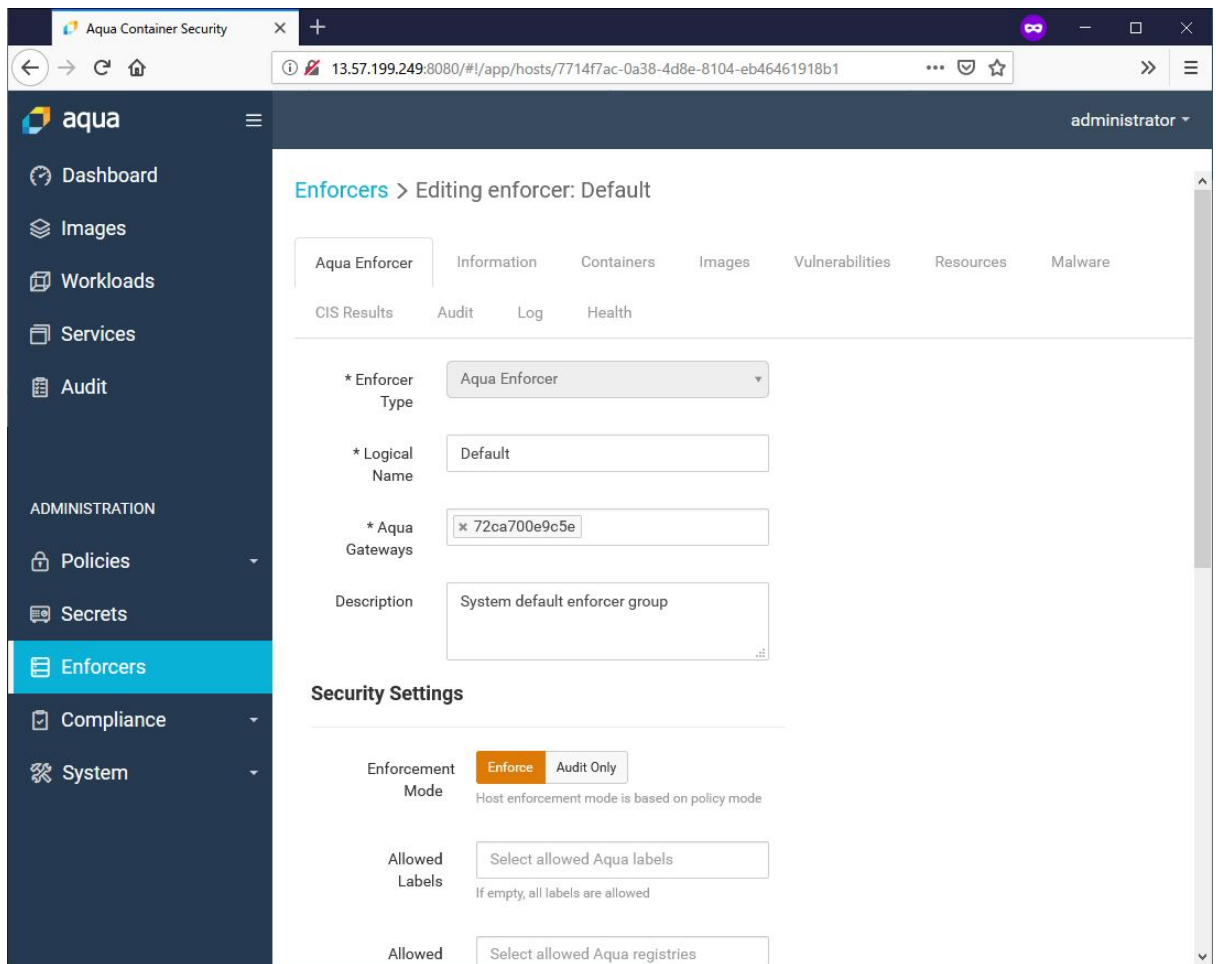
```
$docker run --users=host --rm -e SILENT=yes -e Aqua _TOKEN=トークン \
-e Aqua _SERVER=IPADDR:3622 -e Aqua _LOGICAL_NAME="Default" \
-e RESTART_CONTAINERS="no" -v /var/run/docker.sock:/var/run/docker.sock \
registry.aquasec.com/enforcer:3.5.0
```

7. Aqua のEnforcersで、enforcerが動作していることを確認します。

The screenshot shows the Aqua Enforcers management page. The sidebar on the left contains navigation items: Dashboard, Images, Workloads, Services, Audit, and an ADMINISTRATION section with Policies, Secrets, Enforcers (highlighted), Compliance, and System. The main content area is titled 'Enforcers' and features a refresh icon in the top right. Below the title, there are filters: 'Show enforcers with status:' set to 'All' and a '+ ADD ENFORCER GROUP' button. A search box for 'Filter by group name:' is also present. A table lists enforcers with columns: Name, Security Issues, Mode, Connected, Type, OS, and Orchestrator. One enforcer is listed under the 'default' group, with a green status bar and 'Audit Only' mode. Below the table, there are pagination controls showing 'Showing 1 to 1 of 1 results, up to 20 results per page.' and 'Previous 1 Next' buttons.

8. この状態では、Audit Only(監査のみ)です  
ブロックまでする場合は、Enforcer設定で以下の操作を行い、「Audit Only」から「Enforce」(制御)に切り替えて「Save」を押します





9. Enforcerが所属しているグループで「Edit Group」を選択して、グループ編集画面を表示します

The screenshot shows the Aqua Container Security web interface. The left sidebar contains navigation options: Dashboard, Images, Workloads, Services, Audit, ADMINISTRATION (Policies, Secrets, Enforcers, Compliance, System). The main content area is titled 'Enforcers'. At the top, there's a notification: 'Enforcer Default successfully saved'. Below it, a dropdown menu shows 'All' and a '+ ADD ENFORCER GROUP' button. A search bar is labeled 'Filter by group name:'. A table lists enforcer groups with columns: Name, Security Issues, Mode, Connected, Type, OS, and Orchestrator. The 'default' group is expanded, showing a green progress bar and a 'Audit Only' button. Below this, a table lists individual enforcers with columns: Logical Name, Type, Address, Enforcement Mode, and a status. One enforcer is listed: 'Default.ip-172-31-20...' with Type 'Enforcer', Address '172.31.20.29', Enforcement Mode 'Enforce', and status 'Not Scanned'. A context menu is open over the 'Enforce' button, showing 'Edit Group' and 'Copy Install Command' options. Pagination shows 'Showing 1 to 1 of 1 results, up to 20 results per page.' with 'Previous', '1', and 'Next' buttons.

グループ編集画面で、「Audit Only」を「Enforce」に変更して「Save」を押します

Aqua Container Security administrator

13.57.199.249:8080/#/app/~2Fapp~2Fhosts~2Fbatch.installs/

aqua

- Dashboard
- Images
- Workloads
- Services
- Audit

ADMINISTRATION

- Policies
- Secrets
- Enforcers**
- Compliance
- System

Description: System default enforcer group

### Security Settings

Enforcement Mode: **Enforce** | Audit Only  
Host enforcement mode is based on policy mode

Allowed Labels: Select allowed Aqua labels  
If empty, all labels are allowed

Allowed Registries: Select allowed Aqua registries  
If empty, all registries are allowed

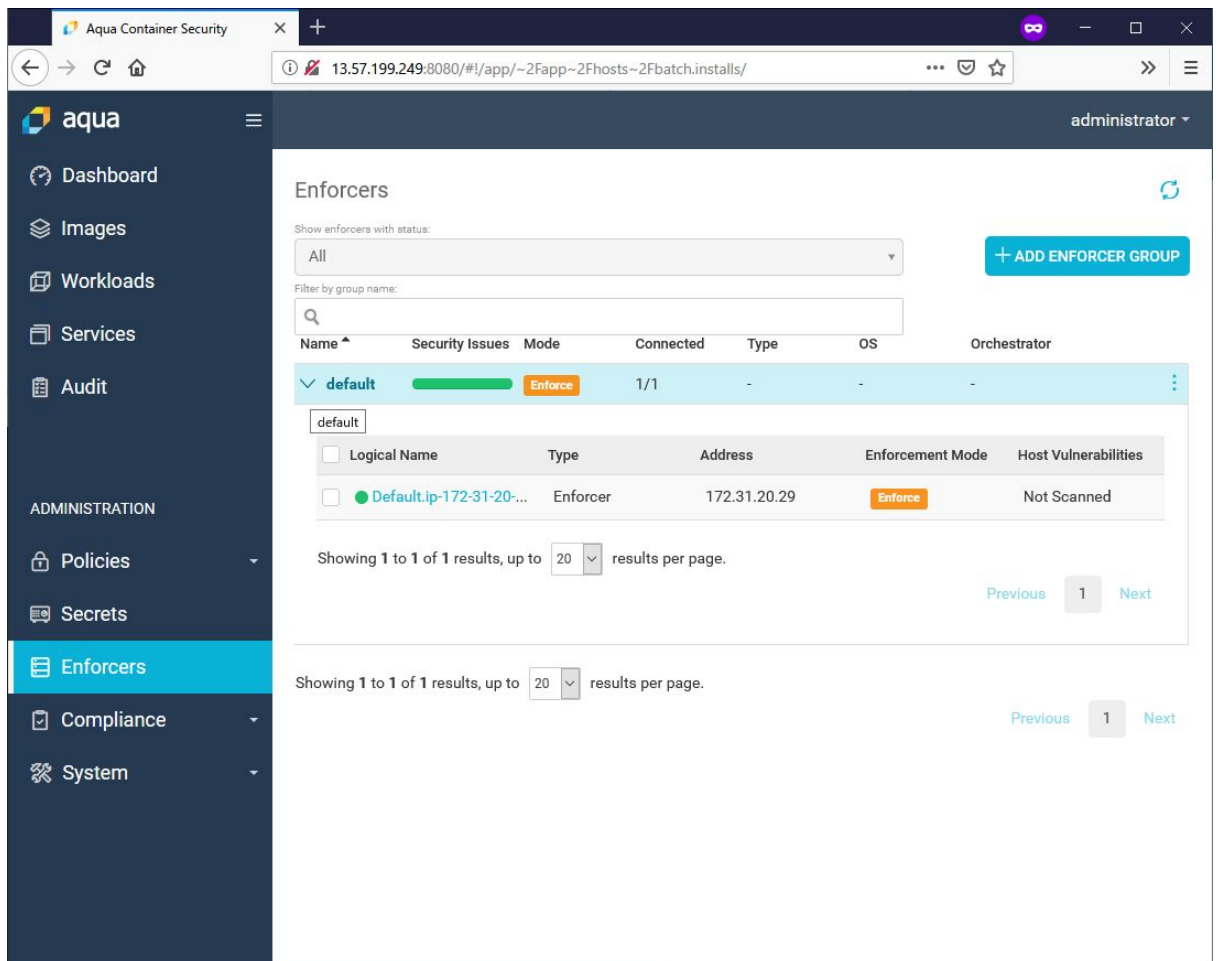
### Auditing

- Audit host successful login events
- Audit host failed login events

### Advanced Settings

- Enable container firewall protection and network map
- Enable system call tracing (beta)

これで、すべてEnforce状態になりました

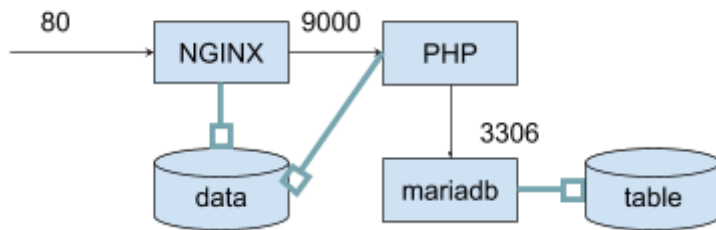


## Aqua CSPを試してみる

Aqua CSPの使用を想定した、DevOpsの例を考えてみます。ここではWordPressを例に、開発→テスト→リリース→運用までをAqua CSPをどう使用するのかを説明します。

### WordPressの構成

WordPressをDockerで提供する場合、WordPress公式イメージを使うことが多いと思います。今回は公式イメージではなく、nginx、php、mariadbを使用します。それぞれ最新のバージョンを使用したいと思います。



WordPressのデータはデータコンテナに配置し、mariadbのテーブルデータは別のデータコンテナに配置します。

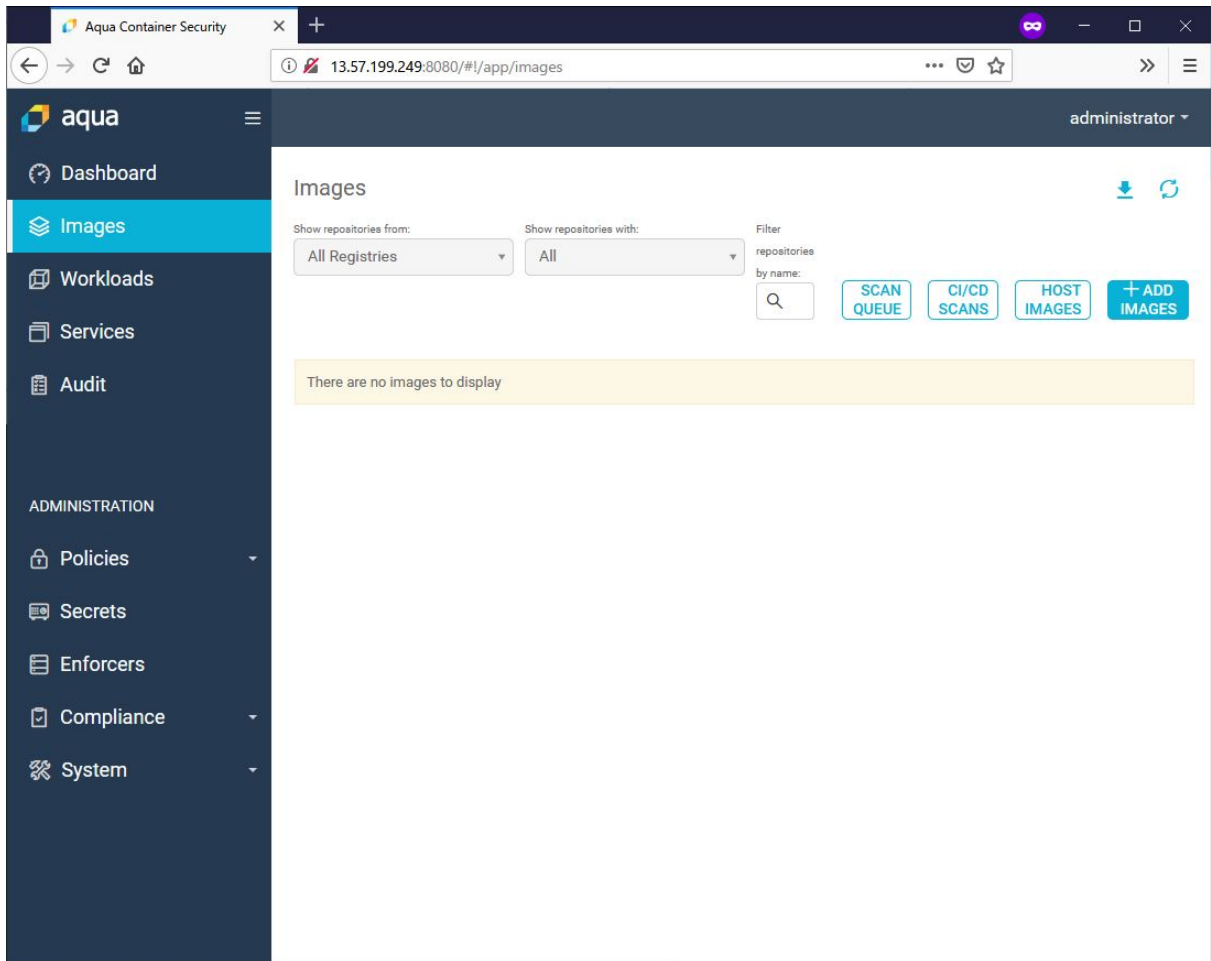
NGINXはhttpを外部にポートフォワードし、PHPはphp-fpm(9000番ポート)でNGINXからのみ接続されます。PHPからmariadbへは、3306番ポートで接続します。

## Image Assurance(イメージ保証)

構成が決定したので、使用するソフトウェアのバージョンを使用したいと思います。通常DockerHubを検索すると出てきますが、今回はAqua CSPのImage Scanを使用してどのイメージが最新か、脆弱性がないかということをも確認したいと思います。

通常latestが最新ということが分かりますが、構成時にどのバージョンが使われるか不定となるため、latestと指定することはできるだけ避けます。

Image Scanでは、Docker Hubを検索してImageの脆弱性や構成要素を確認することが出来ます。



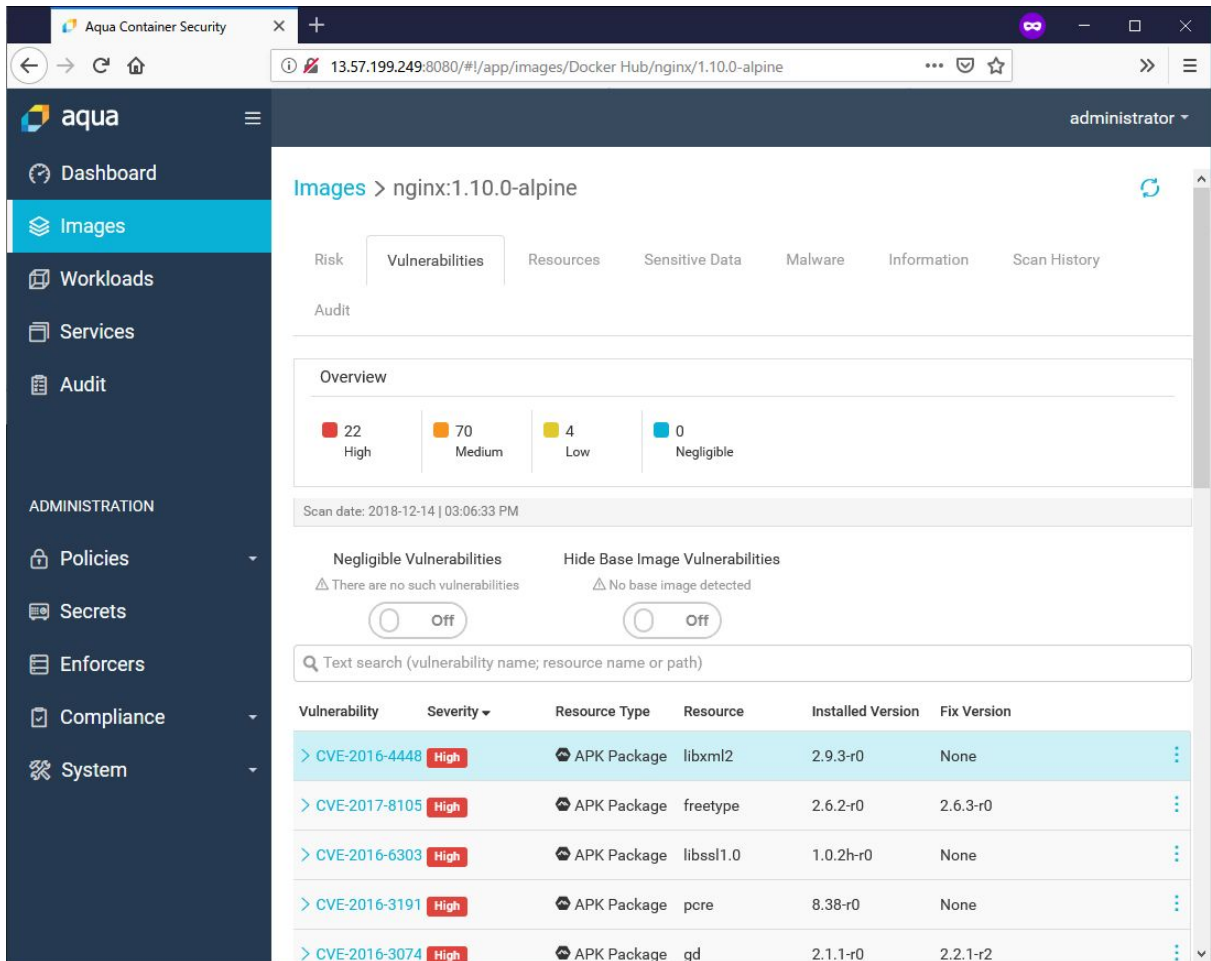
nginxでいくつかの確認を行うと、2018/12/21 時点では、nginx-1.15.7が最新ということが分かりました。

The screenshot shows the Aqua security dashboard interface. The left sidebar contains navigation options: Dashboard, Images (selected), Workloads, Services, Audit, and an ADMINISTRATION section with Policies, Secrets, Enforcers, Compliance, and System. The main content area is titled 'Images' and includes filters for 'Show repositories from:' (All Registries), 'Show repositories with:' (All), and a search box for 'Filter repositories by name:'. Action buttons include 'SCAN QUEUE', 'CI/CD SCANS', 'HOST IMAGES', and '+ ADD IMAGES'. A table displays scan results for 'nginx' repositories:

Repository	Security Issues	Image Profile	Non-compliant	Registry
<input checked="" type="checkbox"/> nginx		<input checked="" type="checkbox"/> None	0/2	Docker Hub
<input type="checkbox"/> nginx:1.10.0-alpine	22 70 4		Select Aqua Labels...	
<input type="checkbox"/> nginx:1.15.7-alpine	0 3 0		Select Aqua Labels...	

At the bottom, it indicates 'Showing 0 to 0 of 0 results, up to 20 results per page.' with 'Previous', '1', and 'Next' navigation links.

Image scanを実施した結果、古いバージョンが安定していそうですが高い脆弱性が多く含まれています。各イメージでどのような脆弱性が含まれているかは、スキャン結果をクリックすると確認することができます。



同様に、PHP、mariadbのイメージを確認して以下のバージョンを使用します。

Image	Version	Digest	Created	Size
\$ docker images mariadb	10.4.1-bionic	72df4d4ba022	3 days ago	375MB
nginx	1.15.8-alpine	315798907716	31 hours ago	17.8MB
php	7.3.0-fpm-stretch	6b807ac9f1f3	2 weeks ago	372MB

Image Scanはスキャン時にイメージをpullし、スキャン実施後にそのイメージを削除します。

今回、NGINXはalpine3.8(apk)をベースOSにしたイメージを採用しています。Image ScanではRHEL/CentOS(yum/RPM)、Debian/Ubuntu/apt/dpkg)を使用したLinuxイメージでも問題なくスキャンし、脆弱性を確認できます。今回は比較のため、mariadbのベースOSをUbuntuに、PHPのOSベースをDebianにしています。

Aqua Enforcerでは、Audit Only モードでは警告を出すだけでですが、Enforceモードにすることで登録したイメージだけを実行可能にしたり、脆弱性チェックで高い脆弱性が検出されたイメージを実行不可能にしたりすることが可能です。



## Runtime Policy(ランタイムポリシー)

Dockerでは、1Dockerイメージで1サービスが動作することが推奨されています。とはいえ、プロセスからforkしたり、別プロセスを起動したりすることは普通にあることです。今回の構成では、NGINXはPHPへの通信とWordPressが提供するファイルの転送だけ、MariaDBはDBサーバと機能が決まっています。しかし、PHPはWordPressのプラグインによってはサブプロセスを起動したり、セキュアではない動作を行うことがあります。実際WordPressはアタックの対象になりやすいですが、WordPressの脆弱性自体はそれほど多くはなく、プラグインに脆弱性が含まれていたり、危険な動作をする場合がほとんどです。

Aqua CSPでは、起動するコマンドの制限したり、起動するプロセス数を制限をすることが出来ます。またEnforceモード動作確認を実施することで想定外のプロセスが起動したり、負荷試験でどのくらいプロセス数が増えるかを確認することが可能です。

## Container Firewall(コンテナファイアウォール)

各コンテナが使用するポートは決まっています、通常はコンテナイメージ間やコンテナイメージと起動マシンで使用するポートがポートフォワードにより設定されます。

コンテナファイアウォールを使用すると、コンテナ間の通信を監査し、必要以外の通信が存在しないことを保証できます。

今回のWordPressの構成では、コンテナ間の通信がPHP(9000番)、MariaDB(3306番)と決まっていますので、例えばNGINXからMariaDBへ通信することはないはずです。

通常のLinuxのネットワークセキュリティでは、NGINX→PHP→MariaDBという経路のインバウンドポートやIPアドレスを制御し、それ以外のアクセスを禁止します。

Docker構成の場合は、Dockerイメージ内で外部へのポートは開放せずポートフォワードによりインバウンド通信を行うため、IPアドレスで制限するということはしないかもしれません。むしろアウトバウンドを制限することで外部への想定外の通信を遮断することが求められます。

Aqua CSPのWebUIでは、実際に動作しているコンテナ間のネットワーク接続の様子や使用するポートの確認、実際に動作させてどのネットワークが使用されているか、不要な通信が行われていないかを監査することが出来ます。また、この監査を元にファイアウォールの設定を行い、想定外の通信を禁止することが可能です。

WordPressの場合では、各Dockerイメージのアウトバウンドとしてプライベートポート(Linuxでは32768 ~60999)のみの通信を許可、および特定コンテナ宛(MariaDB⇄PHP、PHP⇄NGINX)への通信のみを有効にすることで、外部との不要な通信を防ぐことが出来ます。また、複数のサービスを動作させ同じイメージが混在するような場合、ラベルを付けることで通信相手を限定するという事も可能です。

これにより、ワームやbotなどが仕込まれたり、管理者が許可しないネットワーク接続操作(許可されていないパッケージやスクリプトのダウンロード)を防ぐことが出来ます。

## Secrets(シークレット管理)

MySQLのDB名、DBユーザ名、DB名パスワードは、簡単なパスワードやデフォルトパスワードを使用すると類推されてアタックされやすくなります。WordPressのパスワードは設定ファイルに記述されますが、当然コンテナを起動しているマシンから覗き見るのが可能です。

Aqua CSPのシークレット管理機能により、このようなパスワードや暗号鍵を管理し、実行しているプロセスのみがその情報を使用し、可動しているコンテナ外から容易にのぞき見ることを防止できます。

## CI/CD Integration(CI/CDツール統合)

これからのコンテナ開発では、開発時に動作確認を自動化することが重要になってきます。これは、以下の要件に応えるために必要になってきます。

1. 開発サイクルの短期間化により顧客ニーズにすばやく応えるため
2. 使用しているイメージで使用しているソフトウェア脆弱性対策
3. ソフトウェアバージョンアップ

Aqua CSPでは、既存のCI/CDツールと組み合わせることで、WordPress、プラグイン、テーマのバージョンアップに対するのテスト・動作確認だけではなく、コンテナイメージに対しても定期的なImageScanや、テスト中の不正な動作、通信の確認などを確認することで、CI/CDツールと連携して継続的なテストとデプロイを実現できます。