



Cisco WAE 7.6.0 インストール ガイド

初版：2022年6月6日

最終更新：2023年3月16日

シスコシステムズ合同会社

〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー

<http://www.cisco.com/jp>

お問い合わせ先：シスコ コンタクトセンター

0120-092-255（フリーコール、携帯・PHS含む）

電話受付時間：平日 10:00～12:00、13:00～17:00

<http://www.cisco.com/jp/go/contactcenter/>

【注意】 シスコ製品をご使用になる前に、安全上の注意（www.cisco.com/jp/go/safety_warning/）をご確認ください。本書は、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

The documentation set for this product strives to use bias-free language. For purposes of this documentation set, bias-free is defined as language that does not imply discrimination based on age, disability, gender, racial identity, ethnic identity, sexual orientation, socioeconomic status, and intersectionality. Exceptions may be present in the documentation due to language that is hardcoded in the user interfaces of the product software, language used based on standards documentation, or language that is used by a referenced third-party product.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <https://www.cisco.com/c/en/us/about/legal/trademarks.html>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

このマニュアルで使用しているIPアドレスと電話番号は、実際のアドレスと電話番号を示すものではありません。マニュアル内の例、コマンド表示出力、ネットワークトポロジ図、およびその他の図は、説明のみを目的として使用されています。説明の中に実際のアドレスおよび電話番号が使用されていたとしても、それは意図的なものではなく、偶然の一致によるものです。

© 2022 Cisco Systems, Inc. All rights reserved.



第 1 章

Cisco WAE の概要

- [Cisco WAE の概要 \(1 ページ\)](#)
- [Cisco WAE アプリケーション \(2 ページ\)](#)

Cisco WAE の概要

Cisco WAN Automation Engine (WAE) のプラットフォームは、ソフトウェアモジュールを相互接続し、ネットワークと通信し、外部アプリケーションとインターフェイスする API を提供するオープンでプログラマブルなフレームワークです。

Cisco WAE は、ネットワークとそのネットワーク上のトラフィック需要の継続的なモニタリングと分析を通じて、現在のネットワークのモデルを作成および維持するためのツールを提供します。このネットワークモデルには、トポロジ、設定、トラフィック情報など、特定の時点でのネットワークに関するすべての関連情報が含まれています。この情報は、トラフィック要求、パス、ノードとリンクの障害、ネットワークの最適化、またはその他の変更によるネットワークへの影響を分析するための基礎として使用できます。

Cisco WAE プラットフォームの重要な機能の一部は次のようなものです。

- **トラフィック エンジニアリングとネットワークの最適化**：TE LSP 構成を計算してネットワークパフォーマンスを改善したり、ローカルまたはグローバルな最適化を実行したりします。
- **デマンドエンジニアリング**：ネットワーク上のトラフィック需要の追加、削除、または変更がネットワークトラフィックフローに与える影響を調べます。
- **トポロジと予測分析**：設計またはネットワーク障害によって引き起こされるネットワークトポロジの変更がネットワークパフォーマンスに与える影響を観察します。
- **TE トンネルプログラミング**：トンネルパスや予約帯域幅などのトンネルパラメータを変更した場合の影響を調べます。
- **サービスクラス (CoS) 対応のオンデマンド帯域幅**：既存のネットワークトラフィックと需要を調べ、ルータ間で一連のサービスクラス固有の需要を許可します。

Cisco WAE アプリケーション

Cisco WAE アプリケーションは Cisco WAE プラットフォームソフトウェアと連携し、ネットワークに関するより深いインサイトを提供します。

- **Cisco WAE Design** : WAE Design GUI は、ネットワークのグラフィカルなレイアウトを提供し、さまざまなリージョンとレイヤのビュー、および使用率とルーティングを表示します。このようなレイアウトにより、障害、変化、トラフィック増加の影響をモデル化、シミュレート、および分析し、ネットワークを最適化して効率を最大化することができます。シミュレーションツールを使用すると、次の方法でインタラクティブな what-if シミュレーションを実行できます。

- 失敗したオブジェクト
- ネットワークトポロジの変更
- トラフィックデマンドの作成と変更（トラフィックフローをシミュレーションする）
- ルーティング設定の変更

インストールの手順については、[Cisco WAE Design GUI インストールガイド](#) を参照してください。

- **Cisco WAE Live** : Cisco WAE Live により、現在および過去のネットワークデータの両方に、迅速かつ簡単にアクセスできます。検索、分析、およびマップツールの組み合わせにより、要約された集約ビューを見つけたり、ネットワークデータの検索に関連する詳細のみにすばやく絞り込んだりするための、柔軟でインタラクティブな手段が提供されます。

インストール手順については、[Cisco WAE Live のインストール \(29 ページ\)](#) を参照してください。

- **オンデマンド帯域幅** : Bandwidth on Demand (BWoD) アプリケーションは、WMD によって提供されるほぼリアルタイムのネットワークモデルを利用して、XTC から WAE に委任された帯域幅制約を含む SR ポリシーのパスを計算して維持します。帯域幅制約を含む SR ポリシーで使用可能な最短パスを計算し、パスに輻輳がないことを確認するには、パス計算要素 (PCE) によってネットワーク上のトラフィック負荷が認識される必要があります。WAE BWoD アプリケーションは、SR ポリシーの帯域幅認識パス計算の委任を新しい XTC REST API を介して副次的に WAE に委任できるようにすることで、XTC の既存のトポロジ対応 PCE 機能を拡張します。ユーザーは、ネットワーク使用率のしきい値（輻輳の定義）やパス最適化基準の設定などのアプリケーションオプションを選択して、BWoD アプリケーションの動作を微調整し、計算するパスに影響を与えることができます。

BWoD アプリケーションの有効化、設定、および適切なシャットダウンについては、[Cisco WAE ユーザーガイド](#) の「自動化アプリケーション」の章を参照してください。

- **帯域幅最適化** : Bandwidth Optimization アプリケーションとは、ネットワークトラフィックを管理するアプローチで、ネットワークで特定の成果を達成するために少数の LSP を展開することに重点を置いています。この種の戦術的なトラフィックエンジニアリングの例として、輻輳が発生しているリンクからトラフィックを移動する LSP の展開、優先度の高

い音声またはビデオトラフィック用の低遅延 LSP の確立、特定のノードまたはリンクを回避する LSP の展開などがあります。WAE は、ネットワークの状態の変化に対応してトラフィックを管理する **Bandwidth Optimization** アプリケーションを提供します。

帯域幅最適化アプリケーションの有効化、設定、および適切なシャットダウンについては、[Cisco WAE ユーザーガイド](#)の「自動化アプリケーション」の章を参照してください。



第 2 章

Cisco WAE のインストール要件

Cisco WAE の要件は、一緒にインストールされるコンポーネントによって異なります。このセクションでは、特に指定がない限り、単一のサーバーにインストールされる個々のコンポーネントの一般的なガイドラインと最小要件を提供します。

ここでは、次の内容について説明します。

- [Cisco WAE サーバー要件](#) (5 ページ)
- [必要なソフトウェアパッケージ](#) (6 ページ)
- [NetFlow の要件](#) (7 ページ)
- [オプティカル収集エージェント](#) (8 ページ)
- [Network Service Orchestrator からの収集](#) (9 ページ)
- [スケールのサポート](#) (9 ページ)
- [Cisco WAE Design の要件](#) (9 ページ)
- [WAE Live の要件](#) (10 ページ)
- [サポートされる Web ブラウザ](#) (11 ページ)
- [Cisco WAE ポート](#) (11 ページ)

Cisco WAE サーバー要件

以下の要件を満たすサーバーに Cisco WAE をインストールできます。

オペレーティングシステム	ソフトウェア	CPU	メモリ	ハードドライブ
Linux-x86_64	最新のパッチが適用された RHEL 8.4	16 コア以上	64 GB	最小 : 500 GB 推奨 : 1 TB

特記事項

- データストアファイルシステムで物理セクターサイズが 2,048 バイト未満のハードディスクを使用すると、起動時に次のエラーが発生することがあります。

[初期化に失敗しました .. (Failed to initialize the system ..)]

WAE Live データベースを設計どおりに機能させるための、データストアファイルシステムの推奨ブロックサイズは 2,048 バイト以上です。

- Cisco WAE ソフトウェアは RHEL 8.4 で認定されています。
- 英語で利用可能な Linux ディストリビューションのみがサポートされています。
- 次の行を追加または更新して `/etc/security/limits.conf` ファイルを変更し、ポーラーを開始するために十分な数のスレッドが、ハードウェアでサポートされていることを確認します。

```
[user] soft stack 8192
[user] soft nproc 257805
```

[user] は、WAE プロセスを開始するユーザー ID です。

ファイル記述子の制限を設定します。

```
[user]      soft      nofile 1000000
[user]      hard      nofile 1000000
# End of file
```



(注) `Limits.conf` ファイルを変更した後、サーバーを再起動します。

必要なソフトウェアパッケージ

ソフトウェア	バージョン
JDK/JRE	OpenJDK 11.0.7 64 ビット (注) 複数のバージョンの Java がセットアップにインストールされている場合は、 <code>[wae.ini]</code> ファイルの [環境 (Environment)] セクションで <code>JAVA_HOME/JRE_HOME</code> を Java 11 に設定します。
Perl	5.16.3
fontconfig	2.13(1)
Python	3.6.x (注) <code>/usr/bin/python3</code> は、インストールされている Python を指す必要があります。
python3-paramiko. noarch	2.7.2
python3-lxml	46-2
python3-requests	2.20.0

ソフトウェア	バージョン
redhat-usb	4.1 ライセンスサーバーに必要です。詳細については、 Cisco WAE Design GUI インストールガイド の「 WAE Design のフローティング ライセンス サーバー 」の章を参照してください。
スーパーバイザ	4.2.1
Which	2.21
ncurses-compat-libs	6.1.7



(注) python3-paramiko、python3-lxml、python3-requests および supervisor は、epel-release リポジトリを追加した後にインストールする必要があります。

NetFlow の要件

NetFlow 収集 (WAE サーバ要件を除き、専用) サーバーごとのメモリサイズと CPU

集中型 NetFlow (スナップショットが存在するサーバー)		
	メモリ	CPU
pmacct	32 GB	
flow_cluster_broker	2 GB	
flow_cluster_controller	2 GB	
flow_cluster_agent	8 GB	
flow_collector_ias flow_collector_dmd	8 GB	
合計	52 GB	8 コア以上

分散型 NetFlow (エージェントが存在するサーバー)		
	メモリ	CPU
pmacct	32 GB	
flow_cluster_agent	8 GB	

分散型 NetFlow (エージェントが存在するサーバー)		
合計	40 GB	8 コア以上
分散型 NetFlow (スナップショットが存在するサーバー)		
	メモリ	CPU
flow_cluster_broker	2 GB	
flow_cluster_controller	2 GB	
flow_cluster_ias flow_cluster_dmd	8 GB	
合計	12 GB	8 コア以上



- (注)
- 100 Mbps のフローエクスポート帯域幅ごとに1つのフロー収集サーバー (pmacct) が必要です。
 - 英語版の Linux のみがサポートされています。
 - フロー収集には Linux カーネル 2.6.32 以降が必要です。
 - 上記の収集サーバーインスタンスごとのメモリ要件は、NetFlow トラフィックのおおよその数値が 100 Mbit/s であるという仮定に基づいています。

オプティカル収集エージェント

[ベンダー (Vendor)]	サポートされるノードのバージョン	ソフトウェア
Cisco	Cisco Network Convergence Systems (NCS) 2000 シリーズルータ、リリース 11.3	Cisco Evolved Programmable Network Manager 5.1.4
Cisco	Cisco Network Convergence Systems (NCS) 2000 シリーズルータ、リリース 12.3	Cisco Evolved Programmable Network Manager 6.0

Network Service Orchestrator からの収集

ソフトウェア/ドライバ	バージョン
IOS NED	6.66.1
IOS-XR NED	7.30.1
Junos NED	4.6.17
Network Services Orchestrator	5.4.2
トラフィック エンジニアリング	Cisco WAE の担当者にお問い合わせください。

スケールのサポート

パラメータ	規模
ネットワークデバイスの総数	3000
インターフェイスの総数	100000
デマンドの総数	100000
ポリシーの総数 (SR または RSVP または両方)	5000

Cisco WAE Design の要件

WAE Design は、Linux オペレーティングシステムにおける 64 ビットインストールです。

オペレーティングシステム	ソフトウェア	CPU	メモリ
Linux-x86_64	最新のパッチが適用された RHEL 8.4	最小：8 コア 推奨：16 コア以上	最小：16 GB 推奨：64 GB

特記事項

- スタンドアロンの WAE Design システムでは、WAE Collector を使用する必要はありません。
- 英語で利用可能な Linux ディストリビューションのみがサポートされています。

- 一部の WAE 機能には、Perl (5.10 以上) のインストールが必要です。必要なソフトウェアパッケージ (6 ページ) を参照してください。
- 一部の WAE 機能には Python のインストールが必要です。必要なソフトウェアパッケージ (6 ページ) を参照してください。
- FlexLM ライセンス：詳細については、『Cisco WAE Design GUI インストールガイド』を参照してください。
- 英語版の Linux のみがサポートされています。

WAE Live の要件



(注) WAE Live は、WAE サーバーとは別のサーバーにインストールする必要があります。

要件	～ 1000 ノードのネットワーク	～ 2000 ノードのネットワーク
サポートされるオペレーティングシステム	RHEL 8.4	RHEL 8.4
CPU	コア X 8、スレッド X 16	コア X 16、スレッド X 32
メモリ	24 GB	48 GB
ディスク速度	200 MB/s	320 MB/s
ディスク サイズ	3 TB	10 TB
ネットワークオブジェクト数	100,000	500,000



- (注)
- 英語版の Linux のみがサポートされています。
 - その他の Red Hat Enterprise Linux ディストリビューションも動作しますが、公式にはサポートされていません。

カーネルパラメータ

カーネルパラメータ	値
SHMALL	4294967296

カーネルパラメータ	値
SHMMAX	4398046511104
SHMMNI	4096
SEMMNS	32000
SEMMSL	250
SEMOPM	32
ファイル記述子の最大数	65535

サポートされる Web ブラウザ

ブラウザ	バージョン
Google Chrome	62 以降
Firefox	56 以降

Cisco WAE ポート

ポート	プロトコル	タイプ	説明
*:8080	TCP	リスニング	Cisco WAE サーバー
*:8443	TCP	リスニング	Cisco WAE サーバー、 Live サーバー
2022 ~ 2023	TCP	リスニングと発信	Cisco WAE サーバー
*:2024	TCP	リスニング	Cisco WAE サーバー
4569	TCP	リスニングと発信	Cisco WAE サーバー
4570	TCP	リスニング	Cisco WAE サーバー HA
8080	TCP	発信	XTC 収集
164	UDP	発信	SNMP ベースの NIMO
22	TCP	発信	Telnet 経由の収集
23	TCP	発信	SSHによる収集

ポート	プロトコル	タイプ	説明
*:2181	TCP	リスニング	メッセージブローカ
*:9092 ~ 9094	TCP	リスニング	メッセージブローカ
8443	TCP	リスニングと発信	Cisco WAE オプティカル EPNM エージェント
8161	TCP	リスニング	NetFlow JMS OOB
61616	[TCP]	リスニング	NetFlow JMS IB
9090	TCP	リスニング	NetFlow HTTP
2100	UDP	リスニング	NetFlow
179	TCP	リスニング	NetFlow BGP
*:8843	TCP	リスニング	Cisco WAE Coordinated Maintenance (スタンダアロンまたはWAEサーバーの一部として)



第 3 章

Cisco WAE のインストール

ここでは、次の内容について説明します。

- [スーパーバイザのインストールと設定 \(13 ページ\)](#)
- [WAE イメージの確認 \(14 ページ\)](#)
- [Cisco WAE のインストール \(15 ページ\)](#)
- [Multi WAE のインストール \(17 ページ\)](#)
- [Cisco WAE 7.x からのアップグレード \(21 ページ\)](#)
- [Cisco WAE 7.x から Multi WAE へのアップグレード \(22 ページ\)](#)
- [Cisco WAE ライセンスのインストール \(23 ページ\)](#)
- [Cisco WAE の開始と停止 \(24 ページ\)](#)
- [Cisco WAE 7.x からの設定の移行 \(24 ページ\)](#)
- [パッケージまたはテンプレートの更新 \(25 ページ\)](#)
- [Cisco WAE のインストールのトラブルシューティング \(25 ページ\)](#)

スーパーバイザのインストールと設定

WAE をインストールする前に、スーパーバイザをインストールして設定します。



- (注) 次の設定手順は、スーパーバイザが `yum` を使用してインストールされている場合にのみ機能します。他の方法を使用してスーパーバイザをインストールする場合は、`supervisorctl` を非ルートユーザーとして実行するように設定する必要があります。

ステップ 1 スーパーバイザをインストールして確認します。

```
sudo yum install -y epel-release
sudo yum install -y supervisor
supervisord -version
4.2.1
```

ステップ 2 WAE を実行している OS ユーザーの書き込み権限を持つディレクトリを作成します。

```
sudo mkdir -p /opt/supervisor/run
sudo mkdir -p /opt/supervisor/log
sudo chown -R [USER-NAME]:[GROUP-NAME] /opt/supervisor
```

ステップ3 スーパーバイザ設定を、ルートユーザーとして実行しないように更新します。

pid ファイルを /opt/supervisor/run/supervisor.pid に指定し、ユーザーを WAE を実行している OS ユーザーとして指定します。

ルートとして /etc/supervisord.conf を開いて編集します。

- [unix_http_server] セクションで次を実行します。
 - ;file=/var/run/supervisor/supervisor.sock を file=/opt/supervisor/run/supervisor.sock に変更します
 - ;chown=nobody:nogroup を chown=[USER-NAME]:[GROUP-NAME] に変更します
- [supervisord] セクションで次を実行します。
 - ;logfile=/var/log/supervisor/supervisord.log を logfile=/opt/supervisor/log/supervisord.log に変更します。
 - ;pidfile=/var/run/supervisord.pid を pidfile=/opt/supervisor/run/supervisord.pid に変更します
 - ;minfds=1024 を minfds=1000000 に変更します
 - ;minprocs=200 を minprocs=257805 に変更します

(注) [supervisord] セクションの下ではユーザーを設定しないでください。
- [supervisorctl] セクションで次を実行します。
 - ;serverurl=unix:///var/run/supervisor/supervisor.sock を
serverurl=unix:///opt/supervisor/run/supervisor.sock に変更します

ステップ4 スーパーバイザを開始します。

```
sudo systemctl start supervisord
sudo supervisorctl status all
```

ステップ5 システムの起動時にスーパーバイザが起動できるようにします。

```
sudo systemctl enable supervisord
sudo systemctl status supervisord
```

WAE イメージの確認

ステップ1 Cisco ダウンロード ソフトウェア サイトから、Cisco WAE 7.6.0 ソフトウェアパッケージをダウンロードします。

ステップ2 証明書とデジタル署名は両方ともダウンロードされたファイル (wae-linux-v7.6.0.signed.bin) に組み込まれています。

ステップ3 自己解凍型の署名付きバイナリを実行します。実行するとリリースバイナリが抽出され、署名ファイルを使用して検証されます。

署名済みイメージの検証

```
[admin@wae-vm-21 workspace.signed]$ ./wae-linux-v7.6.0.signed.bin
Unpacking...
Verifying signature...
Downloading CA certificate from http://www.cisco.com/security/pki/certs/crcam2.cer ...
Successfully downloaded and verified crcam2.cer.
Downloading SubCA certificate from http://www.cisco.com/security/pki/certs/innerspace.cer ...
Successfully downloaded and verified innerspace.cer.
Successfully verified root, subca and end-entity certificate chain.
Successfully fetched a public key from WAE-CCO_RELEASE.cer.
Successfully verified the signature of wae-linux-v7.6.0.bin using WAE-CCO_RELEASE.cer
```

ステップ4 生成された wae-linux-v7.6.0.bin は、WAE の Linux インストーラです。

Cisco WAE のインストール

始める前に



(注) 以前の WAE 7.x リリースから WAE 7.6.0 にアップグレードする場合は、[Cisco WAE 7.x からのアップグレード \(21 ページ\)](#) を参照してください。

- まだ存在していない場合は、(グループに割り当てられる) UNIX ユーザーを作成します。インストールを実行するには、この UNIX ユーザーである必要があります。
- Java 11 および Python 3.6.x がシステムにインストールされていることを確認してください。JAVA_HOME 環境変数はjdk-11.0を指し、/usr/bin/python3はインストールされたPythonを指している必要があります。
- スーパーバイザがインストールおよび設定されていることを確認します。[スーパーバイザのインストールと設定 \(13 ページ\)](#) を参照してください。
- デジタル署名された Cisco WAE 7.6.0 イメージをダウンロードして確認します。[WAE イメージの確認 \(14 ページ\)](#) を参照してください。
- WAE で BW-OPT アプリケーションが機能するように、requests.auth python パッケージがインストールされていることを確認してください。

ステップ1 実行中の場合は WAE を停止します。

ステップ2 次のコマンドを使用して、インストールファイルのアクセス許可を変更します。

```
chmod +x wae-linux-v7.6.0.bin
```

ステップ3 ターゲットディレクトリを指定してインストーラを実行します。

```
./wae-linux-v7.6.0.bin <wae-dir>
```

ステップ4 送信元 waerc のインストールディレクトリに移動します。環境を設定し、パスを指定してランタイムディレクトリを作成します。

```
cd <wae-dir>
source waerc
wae-setup --dest <target-runtime-dir>
```

ステップ5 Cisco WAE 管理者パスワードの設定を求められます。

```
WAE admin password:
Confirm password:
```

ステップ6 WAE をインストールして設定した後（wae-setup の実行後など）、/etc/supervisord.d/ 内から wae.ini ファイルへのソフトリンクを作成し、WAE 設定をスーパーバイザに追加します。

```
sudo ln -sf <target-runtime-dir>/wae.ini /etc/supervisord.d/
```

(注) • この手順は、スーパーバイザがインストールおよび設定された後にのみ実行してください。

- JAVA_HOME/JRE_HOME を設定する必要がある external-executable-nimo ベースのネットワークを使用する場合は、target-runtime-dir/wae.ini ファイル内のセクション [program:waectl] を編集し、環境内に JAVA_HOME="valid_jdk_path" を含めます。

たとえば、[program:waectl] の下で編集し、次を追加します。

```
JAVA_HOME:environment=HOME="/home/wae", NCS_JAVA_VM_OPTIONS="-Xmx32G -Xms16G
-XX:+UseG1GC -XX:+HeapDumpOnOutOfMemoryError
-XX:HeapDumpPath=/home/wae/test/run/logs/ -Djava.io.tmpdir=/home/wae/test/run/work/",
TMPDIR="/home/wae/test/run/work/", JAVA_HOME="/usr/"
```

- 新しい wae.ini の変更を有効にするには、**supervisorctl update** を実行します。

ステップ7 スーパーバイザ設定を更新します。

```
sudo supervisorctl update
```

ステップ8 WAE プロセスの開始

```
sudo supervisorctl start wae:*
wae:zookeeper: started
wae:waectl: started
wae:kafka: started
wae:wae-monitor: started
```

- (注)
- wae:waectl は WAE プログラムです。
 - wae:kafka および wae:zookeeper は、トラフィックの収集と内部メッセージに必要です。
 - wae:wae-monitor はモニタリングサービスです。
 - wae:logrotate はログローテーション用です。

ステップ 9 WAE プロセスのステータスの確認

```
sudo supervisorctl status
wae:kafka RUNNING pid 1540, uptime 28 days, 14:03:40
wae:logrotate RUNNING pid 1178, uptime 28 days, 15:10:11
wae:wae-monitor RUNNING pid 11520, uptime 0:00:12
wae:waectl RUNNING pid 1177, uptime 28 days, 15:10:11
wae:zookeeper RUNNING pid 1736, uptime 28 days, 14:03:39
```

(注) すべての WAE プロセスを停止するには、次のコマンドを使用します。

```
sudo supervisorctl stop wae:*
```

ステップ 10 WAE 7.x.x リリースから WAE 7.6.0 リリースへ設定を移行するには、Cisco ダウンロード ソフトウェア サイトから、Cisco WAE アップグレードスクリプトを使用します。

(注) サーバー/VM を再起動すると、すべての WAE サービスは自動的に再起動されず、停止状態になります。WAE サービスは手順 8 で説明したコマンドを使用して開始できます。

Multi WAE のインストール

始める前に

- python3 ベースの Ansible バージョン 2.10.7 以降をインストールします。次のコマンドを使用します。

```
sudo yum install ansible
```

- すべてのリモートホストに Java 11 をインストールします。
- すべてのリモートホストとプレイブックが実行されているホストに Python3 をインストールします。



- (注)
- RHEL 8.4 では、**waerc** が提供されていないターミナルからプレイブックを実行します。
 - スプリット数が増えるたびに、スケールプライマリで WAE を再起動します。

- マルチ WAE に参加しているサーバー間でパスワードなしの ssh を有効にします (セルフ ssh を含む)。

ステップ 1 ansible.cfg をエクスポートします。カスタムの ansible.cfg ファイルは、playbooks/ansible.cfg で提供されません。使用するコマンドは、次のとおりです。

```
export ANSIBLE_CONFIG=<path-to-the-ansible-config-file>
```

ステップ 2 プレイブックを実行する予定のマシンで、自分自身に SSH を実行し、**playbooks/known_hosts** ファイルの **self** にエントリを追加します。Multi WAE のインストールにより、異なるマシン間で単一の **username** と **wae_dir** のみがサポートされます。**-u** フラグを渡すと、**ansible-playbook** コマンドを呼び出すときに、CLI から **ansible_ssh_user** を渡すこともできます。

```
ansible-playbook wae_install.yml -u <username> --ask-pass
```

ステップ 3 **Playbooks/visudo** ファイルの最後に次の行を追加して、パスワードなしで **sudo** コマンドを実行できることを確認します。

```
<username> ALL=(ALL) NOPASSWD:ALL
```

ステップ 4 **Playbooks/hosts** ファイルを変更して、マシンの IP アドレスを含めます。**hosts** ファイルには、**[remote]**、**[primary]**、**[secondary]** の 3 つのグループがあります。

```
[remote]
'element-1' ansible_ssh_user='TARGET_SSH_USER'
'element-2' ansible_ssh_user='TARGET_SSH_USER'
'element-3' ansible_ssh_user='TARGET_SSH_USER'
```

```
[primary]
'element-1' ansible_ssh_user='TARGET_SSH_USER'
```

```
[secondary]
'element-2' ansible_ssh_user='TARGET_SSH_USER'
```

where,

```
[remote] - indicate the set of hosts in which the playbooks are to be run
[primary] - is the host which should be set as primary when configuring HA. Must be one of the
host present in [remote] group.
[secondary] - is the host which should be set as secondary when configuring HA. Must be one of the
host present in [remote] group.
```

- (注)
- **[remote]** グループは、すべてのプレイブックの実行に必須です。
 - **[primary]** および **[secondary]** グループは、**ha_config** プレイブックの実行にのみ必要です。

ステップ 5 プレイブックに必要な入力パラメータを **group_vars/all** ファイルに設定します。このファイルは **playbooks/group_vars/all** (次の表を参照) にあり、プレイブックを実行します。次の表に、使用可能なプレイブックの詳細を示します。

表 1: Ansible プレイブックの詳細

プレイブック	説明	パラメータ	使用方法
wae_install.yml	wae_install.yml プレイブックは、WAE バイナリをコピーし、スーパーバイザを使用してサーバーを稼働させるために必要な関連チェックとタスクを実行することにより、リモートマシンに WAE をインストールします。	<ul style="list-style-type: none"> • WAE_USER_NAME : すべてのリモートマシンに既存の WAE ユーザー (sudo 対応)。 • WAE_BIN_PATH : ansible-playbook が実行されているマシンにある WAE バイナリへの絶対パス。 • WAE_DIR : wae-install および wae-run ディレクトリを保持する WAE ディレクトリの絶対パス。 • DELETE_SIGNED : インストールの完了後に署名された WAE イメージをクリーンアップする必要があるかどうかを示すために使用されるフラグ。 デフォルト値は False です。 	<pre>ansible-playbook wae_install.yml -i <path_to_inventory_file> --ask-pass</pre>
kafka_config.yml	kafka_config.yml プレイブックは、内部および外部リスナーに適切な設定を設定することにより、リモートマシンに kafka を展開します。	<ul style="list-style-type: none"> • WAE_DIR : wae-install および wae-run ディレクトリを保持する WAE ディレクトリの絶対パス。 	<pre>ansible-playbook kafka_config.yml -i <path_to_inventory_file> --ask-pass</pre>

プレイブック	説明	パラメータ	使用方法
ha_config.yml	ha_config.yml プレイブックは、指定の WAE の実行中に 2 つのノード間で HA を展開します。	<ul style="list-style-type: none"> • WAE_USER_NAME : すべてのリモートマシンに既存の WAE ユーザー (sudo 対応)。 • WAE_DIR : wae-install および wae-run ディレクトリを保持する WAE ディレクトリの絶対パス。 • WAE_HA_XML_TEMPLATE : 2 つのノードの CDB にロードされる WAE HA 設定を含む XML テンプレート。 	<pre>ansible-playbook ha_config.yml -i <path_to_inventory_file> --ask-pass</pre>

プレイブック	説明	パラメータ	使用方法
load_config.yml	load_config.yml プレイブックは、リモート WAE サーバーに WAE 設定をロードすることを目的としています。	<ul style="list-style-type: none"> • WAE_DIR : wae-install および wae-run ディレクトリを保持する WAE ディレクトリの絶対パス。 • WAE_CFGS_SRC_DIR : ansible-playbook が実行されているマシン上に設定が存在するディレクトリの絶対パス。 • WAE_CFGS : 設定ファイルの名前のリスト。ファイルは WAE_CFGS_SRC_DIR に存在する必要があります。 • WAE_TMP_CFGS_DEST_DIR : 設定ファイルがコピーされるリモートマシンのディレクトリへの絶対パス。 存在しない場合は、ディレクトリが作成されます。デフォルト値は /tmp/wae_cfgs です。 	<pre>ansible-playbook load_config.yml -i <path_to_inventory_file> --ask-pass</pre>

Cisco WAE 7.x からのアップグレード

始める前に

- [Cisco ダウンロードソフトウェア](#) サイトから、Cisco WAE アップグレードスクリプトをダウンロードします。

- デジタル署名された Cisco WAE 7.6.0 イメージをダウンロードして確認します。 [WAE イメージの確認 \(14 ページ\)](#) を参照してください。
- Java 11 および Python 3.6.x がシステムにインストールされていることを確認してください。 JAVA_HOME 環境変数は jdk-11.0 を指し、 /usr/bin/python3 はインストールされた Python を指している必要があります。
- 次のコマンドを使用して、pexpect をインストールします。

```
sudo pip3 install pexpect
```
- スーパーバイザがインストールおよび設定されていることを確認します。 [スーパーバイザのインストールと設定 \(13 ページ\)](#) を参照してください。
- アップグレードを実行する前に HA を無効にします。アップグレードスクリプトは、以前の WAE インストールに存在する特定の機能パックに関連する設定を処理しません。次の操作を実行できます。
 - アップグレードを実行する前に、機能パックに関連する設定を削除します。または、
 - WAE を手動でインストールし ([Cisco WAE のインストール \(15 ページ\)](#) を参照)、新しい WAE インストールに機能パックをインストールしてから、設定をインポートします ([Cisco WAE 7.x からの設定の移行 \(24 ページ\)](#) を参照)。

ステップ 1 7.x がインストールされたマシンにログインします。

ステップ 2 wae_upgrade スクリプトを実行します。

(注) --wae-bin オプションとして渡されるインストールファイルは、デジタル署名された Cisco WAE 7.6.0 イメージの検証後に取得されたイメージです。

```
# ./wae_upgrade --upgrade --old-install-dir <WAE_7.x_INSTALL_DIR> --old-run-dir <WAE_7.x_RUN_DIR>
--new-install-dir <WAE_7.6.0_INSTALL_DIR> --new-run-dir <WAE_7.6.0_RUN_DIR> --cfg-dir
<dir_to_save_config> --wae-bin <WAE_7.6.0_INSTALLATION_FILE>
where
--old-install-dir    indicates the directory where 7.x WAE is installed
--old-run-dir        indicates the directory where the run time for 7.x WAE resides
--new-install-dir    indicates the directory where 7.6.0 WAE must be installed
--new-run-dir        indicates the directory where the run time for 7.6.0 WAE will reside
--cfg-dir            indicates the folder where the config is to be saved. This config will be
changed to match 7.6.0 and pushed to 7.6.0
--wae-bin            indicates the path to WAE 7.6.0 installation file.
```

Cisco WAE 7.x から Multi WAE へのアップグレード

ステップ 1 WAE インストールを Cisco WAE 7.6 にアップグレードします。 [Cisco WAE 7.x からのアップグレード \(21 ページ\)](#) を参照してください。

- ステップ 2** ansible playbook `load_config.yml` を使用してエージェントと NIMO を設定し、アップグレードされた WAE インスタンスで Multi WAE を手動で設定します。[Multi WAE のインストール \(17 ページ\)](#) を参照してください
- ステップ 3** エクスポートオプションを指定してアップグレードスクリプトを実行し、更新された WAE インスタンスから設定を収集します。[Cisco WAE 7.x からの設定の移行 \(24 ページ\)](#) を参照してください。
- ステップ 4** `wae_install` および `load_config` プレイブックを使用して、他の WAE インスタンスに WAE をインストールし、設定します。

Cisco WAE ライセンスのインストール

Cisco WAE のすべての機能を使用するには、ライセンスが必要です。ライセンスの取得について質問がある場合は、シスコのサポート担当者またはシステム管理者にお問い合わせください。

Cisco WAE は、Cisco Smart Licensing と従来のライセンスをサポートしています。従来のライセンスから Cisco Smart Licensing への切り替えを希望する場合は、Cisco WAE アカウント担当者にお問い合わせください。2種類のライセンスの違いについては、[Cisco.com](#) で紹介している Cisco Smart Licensing の概要を参照してください。

Cisco Smart Licensing の詳細については、『*Cisco WAE ユーザーガイド*』の「Smart Licensing」の章を参照してください。

従来のライセンスのインストール

従来のライセンスをインストールするには、次の手順を実行します。

- ステップ 1** `license_install` ツールを実行し、ライセンスファイルの名前（.lic 拡張子付き）を渡します。デフォルトでは、ツールによって、新しいライセンスで付与されたすべての機能と既存のライセンスの機能がマージされます。

```
license_install -file <path>/<license_name>.lic
```

- ステップ 2** プロンプトが表示されたら、ライセンスをインストールするディレクトリに関連付けられた数字を入力します。

- (注)
- オプション 2 (<wae-dir>/etc) が選択されている場合、新しいビルドのインストール時にライセンスを再インストールする必要があります。
 - オプション 1 (/cariden/etc) が選択されている場合、ライセンスの有効期限が切れていない限り、ライセンスを再インストールする必要はありません。
 - ライセンスがインストールされたら、`license_check` コマンドを実行して、インストールされたライセンスを確認できます。

ステップ3 WAE を停止してから起動し、インストールされているライセンスを取得します。

スマートライセンスのインストール

スマートライセンスをインストールするには、次の手順を実行します。

ステップ1 スマートライセンスを設定するには、ユーザーガイドの「スマートライセンス」のセクションを参照してください。

ステップ2 WAE を停止してから起動し、インストールされているライセンスを取得します。

Cisco WAE の開始と停止

Cisco WAE ランタイムディレクトリから関連する Cisco WAE CLI コマンドを入力して、Cisco WAE サービスを開始または停止します。

- WAE の開始

```
sudo supervisorctl start wae:*  
wae:zookeeper: started  
wae:waectl: started  
wae:kafka: started  
wae:wae-monitor: started
```

- WAE の停止

```
sudo supervisorctl stop wae:*
```

Cisco WAE 7.x からの設定の移行

Cisco WAE アップグレードスクリプトユーティリティを使用して、WAE 7.x から設定を移行できます。

始める前に

- WAE 7.x から WAE 7.6.0 パッケージへ設定を移行するための Cisco WAE アップグレードスクリプトを、[Cisco ダウンロード ソフトウェア](#) サイトからダウンロードします。
- 設定の移行に進む前に、WAE 7.6.0 をインストールして WAE プロセスを開始します。[Cisco WAE のインストール \(15 ページ\)](#) を参照してください
- 次のコマンドを使用して、pexpect をインストールします。

```
sudo pip3 install pexpect
```
- アップグレードを実行する前に HA を無効にします。

- アップグレードスクリプトは、以前の WAE インストールに存在する特定の機能パックに関連する設定を処理しません。次の操作を実行できます。
 - エクスポートする前に、機能パックに関連する設定を削除します。または、
 - 設定をインポートする前に、新しい WAE インストールに機能パックをインストールします。

ステップ 1 7.x 設定のバックアップを取るには、7.x がインストールされているマシンにログインし、`--export` オプションを指定して `wae_upgrade` スクリプトを実行します。

```
# ./wae_upgrade --export --install-dir <WAE_7.x_INSTALL_DIR> --run-dir <WAE_7.x_RUN_DIR> --cfg-dir
<dir_to_save_exported_config>
Where:
  --install-dir  indicates the directory where 7.x WAE is installed
  --run-dir      indicates the directory where the run time for 7.x WAE resides
  --cfg-dir      indicates the folder where backup of 7.x configuration must reside
```

ステップ 2 7.x 設定を 7.6.0 に復元するには、7.6.0 がインストールされているマシンにログインし、`--import` オプションを指定して `wae_upgrade` スクリプトを実行します。

```
# ./wae_upgrade --import --install-dir <WAE_7.6.0_INSTALL_DIR> --run-dir <WAE_7.6.0_RUN_DIR>
--cfg-dir <dir_to_import_saved_config>
Where:
  --install-dir  indicates the directory where 7.6.0 WAE is installed
  --run-dir      indicates the directory where the run time for 7.6.0 WAE resides
  --cfg-dir      indicates the folder where backup of 7.x configuration resides
```

パッケージまたはテンプレートの更新

`<wae_run_time_directory>/packages` ディレクトリに、パッケージまたはテンプレートが更新または追加された場合、Cisco 式の WAE CLI を使用してパッケージのリロードを要求します。

```
$ packages reload
```

たとえば、`wae.conf` ファイルを編集するときパッケージのリロードを実行します。

Cisco WAE のインストールのトラブルシューティング

Cisco WAE のステータスを確認するには、`sudo supervisorctl status` と入力します。

Cisco WAE には YANG ランタイムの標準ログ機能が搭載されています。Cisco WAE は、`<wae-run-time>/logs` ディレクトリの複数のログファイルにログを記録します。

LDAP 認証ログは、`[wae-run-time]/logs/wae-ldap-auth.log` ファイルに記録されます。`[wae-install-dir]lib/exec/test-java-ssl-conn`にあるツールは、証明書の問題のデバッグに役立つ情報を提供する LDAP 認証や EPNM 通知など、Java アプリケーションの SSL 接続をテストするのに便利です。

最も有用なログは `<wae-run-time>/logs/wae-java-vm.log` です。ほとんどの Cisco WAE パッケージは、このファイルにログを記録します。一部の Cisco WAE パッケージは、`<wae-run-time>/logs/wae-python-vm-<package-name>.log` にログを記録します。次の例は、Python-VM ベースのログを示しています。

```
[wae@wae logs]$ pwd
/home/wae/wae-run/logs
[wae@host logs]$ ls -ltr wae-python-vm*
-rw-rw-r-- 1 wae wae 0 Feb 26 07:50 wae-python-vm-cisco-wae-opm-tte.log
-rw-rw-r-- 1 wae wae 0 Feb 26 07:50 wae-python-vm-cisco-wae-get-plan.log
-rw-rw-r-- 1 wae wae 0 Feb 26 07:50 wae-python-vm-cisco-wae-dmdmesh-creator-nimo.log
-rw-rw-r-- 1 wae wae 0 Feb 26 07:50 wae-python-vm-cisco-wae-layout-nimo.log
-rw-rw-r-- 1 wae wae 0 Feb 26 07:50 wae-python-vm-cisco-wae-opm-load-plan.log
-rw-rw-r-- 1 wae wae 0 Feb 26 07:50 wae-python-vm-cisco-wae-dmddeduct-nimo.log
-rw-rw-r-- 1 wae wae 0 Feb 26 07:50 wae-python-vm-cisco-wae-archive.log
-rw-rw-r-- 1 wae wae 2238 Feb 26 07:50 wae-python-vm.log
-rw-rw-r-- 1 wae wae 270 Feb 26 08:20 wae-python-vm-nso_wae_nodes_insert.log
```

デフォルトで、ログレベルは [情報 (INFO)] に設定されています。次の方法でロギングを設定できます。

- ランタイムディレクトリの `wae.conf` ファイルで、さまざまなログのログレベルを定義します。`wae.conf` ファイルについては、『Cisco WAE ユーザーガイド』を参照してください。
- エキスパートモードを使用して、一部のネットワーク インターフェイス モジュール (NIMO) のログ機能を設定します。たとえば、トポロジ NIMO や `lsp-snmp-nimo` モジュールなどのロギング機能を設定できます。エキスパートモードについては、[Cisco WAE ユーザーガイド](#) を参照してください。
- Cisco WAE CLI を使用して、さまざまな NIMO コンポーネントのログレベルを定義します。ログレベルを定義するには、コマンドラインで次のコマンドを入力します。

```
admin@wae% set java-vm java-logging logger <nimo-component> level <level-x>
```

レベルタイプは、`level-info`、`level-debug`、`level-all` です。ログは `wae-java-vm.log` に保存され、トラブルシューティングに使用できます。

次の表に、基本的な NIMO コンポーネントを示します。

NIMO コンポーネント	説明
<code>com.cisco.wae</code>	一般的なデバッグ
<code>com.cisco.wae.nimo.topo</code>	トポロジベースの NIMO デバッグ
<code>com.cisco.wae.nimo.lspconfig</code>	NED デバッグによる LSP 設定

NIMO コンポーネント	説明
com.cisco.wae.nimo.lsp	LSP デバッグ
com.cisco.wae.nimo.snmptrafficpoller	SNMP トラフィックポーラーのデバッグ
com.cisco.wae.dare	集約のデバッグ
com.cisco.wae.nimo.optical	オプティカル NIMO デバッグ

ssh: symbol lookup error

waerc ファイルが送信元である場合、waerc によって LD_LIBRARY_PATH 環境変数に設定された WAE 固有の openssl ライブラリが原因で、ssh および scp コマンドが失敗することがあります。エラーメッセージ「ssh: symbol lookup error: /lib64/libk5crypto.so.3: undefined symbol: Camellia_cbc_encrypt, version OPENSSL_1_1_0」が表示されます。

この問題を解決するには、次の手順のいずれかを実行します。

- ssh、scp、または waerc が送信元に指定されていないターミナルセッションで、openssl を使用するその他の操作を使用します。
- waerc を送信元に設定した後に ssh または scp コマンドを使用している場合は、LD_LIBRARY_PATH 環境変数を waerc を送信元に設定する前の値に設定します。



第 4 章

Cisco WAE Live のインストール

ここでは、次の内容について説明します。

- [WAE Live イメージの確認 \(29 ページ\)](#)
- [Cisco WAE Live のインストール \(30 ページ\)](#)
- [Cisco WAE Live 7.1.x から Cisco WAE Live 7.6.0 へのアップグレード \(31 ページ\)](#)
- [WAE 6.4.10 以降の Live データの WAE Live 7.1.x への移行 \(32 ページ\)](#)
- [Cisco WAE Live データストア \(36 ページ\)](#)
- [Cisco WAE Live ライセンスのインストール \(43 ページ\)](#)
- [認証失敗エラーのトラブルシューティング \(44 ページ\)](#)

WAE Live イメージの確認

ステップ 1 [Cisco ダウンロード ソフトウェア](#) サイトから、Cisco WAE Live 7.6.0 ソフトウェアパッケージをダウンロードします。

ステップ 2 証明書とデジタル署名は両方ともダウンロードされたファイル (WAE-Live-7.6.0-Linux_x86-64.signed.bin) に組み込まれています。

ステップ 3 自己解凍型の署名付きバイナリを実行します。実行するとリリースバイナリが抽出され、署名ファイルを使用して検証されます。

署名済みイメージの検証

```
[admin@wae-vm-21 workspace.signed]$ ./WAE-Live-7.6.0-Linux_x86-64.signed.bin
Unpacking...
Verifying signature...
Downloading CA certificate from http://www.cisco.com/security/pki/certs/crcam2.cer ...
Successfully downloaded and verified crcam2.cer.
Downloading SubCA certificate from http://www.cisco.com/security/pki/certs/innerspace.cer ...
Successfully downloaded and verified innerspace.cer.
Successfully verified root, subca and end-entity certificate chain.
Successfully fetched a public key from WAE-CCO_RELEASE.cer.
Successfully verified the signature of WAE-Live-7.6.0-Linux_x86-64.bin using WAE-CCO_RELEASE.cer
```

ステップ 4 生成された WAE-Live-7.6.0-Linux_x86-64.bin は、WAE の Linux インストーラです。

Cisco WAE Live のインストール

始める前に

- Cisco WAE Live は、Cisco WAE 7.6.0 サーバソフトウェアがインストールされているのと同じマシンにはインストールできません。
- すべての Cisco WAE Live サーバ要件が満たされていることを確認します ([WAE Live の要件 \(10 ページ\)](#) を参照してください)。
- Cisco WAE Live をルートユーザーとしてインストールしないでください。
- サーバに Cisco WAE 7.6.0 Live ライセンスがあることを確認します。
- Java 11 がシステムにインストールされており、`JAVA_HOME` 環境変数が `jdk-11.0` を指していることを確認してください。
- サーバの Automatic Update of Date and Time (日付と時刻の自動更新) を有効にし、Date and Time (日付と時刻) の形式を 24 時間表示に選択します。NTP サーバを使用する場合、この設定は必要ありません。
- WAE Live は `$CARIDEN_ROOT/software/live` にインストールされています。
例: `/opt/wae/software/live` または `/opt/cariden/software/live`。
- Cisco WAE Live 6.4.9 以前のデータを移行する場合は、最初に Cisco WAE Live 6.4.10 にアップグレードしてください。次に、Cisco WAE Live 7.6.0 のインストールを求めるプロンプトが表示されたときに、Cisco WAE Live 6.4.x で使用したのと同じインストールディレクトリを入力します。たとえば、Cisco WAE Live 6.4.x で `$CARIDEN_ROOT` が `/opt/cariden` として定義されている場合、Cisco WAE 7.6.0 の `$CARIDEN_ROOT` も同様に `/opt/cariden` として定義されていることを確認します。

ステップ 1 [Cisco ダウンロード ソフトウェア](#) サイトから、Cisco WAE Live パッケージに移動してダウンロードします。

ステップ 2 サーバにログインし、Cisco WAE Live package `WAE-Live-7.6.0-Linux_x86-64.signed.bin` をローカルディレクトリにコピーして、`bash shell` を開始します。

ステップ 3 署名済みパッケージを解凍します。

```
# ./WAE-Live-7.6.0-Linux_x86-64.signed.bin
```

ステップ 4 `.bin` ファイルへの実行可能アクセスを許可します。

```
# chmod +x WAE-Live-7.6.0-Linux_x86-64.bin
```

ステップ 5 Cisco WAE Live パッケージをインストールします。

```
# bash WAE-Live-7.6.0-Linux_x86-64.bin
```


ステップ 6 プロンプトが表示されたら、`yum` コマンドを使用して必要なソフトウェアパッケージをインストールします。

ステップ 7 インストール画面の指示に従います。

ステップ 8 インストール後、環境変数と `source ~/.profile` を設定して、必要な設定を取得します。

```
# source ~/.profile
```

ステップ 9 Cisco WAE Live データストアをインストールします。詳細については、[WAE Live データストアのインストール \(36 ページ\)](#) を参照してください。

ステップ 10 Cisco WAE Live サービスを開始します。

```
# wae-live-start
```

(注) データストアは、Cisco WAE Live を開始する前に設定する必要があります。

ステップ 11 サポートされているブラウザのいずれかを起動し、`https://server-ip:8443` と入力します。ここでの `server-ip` は、WAE Live がインストールされているサーバーの IP アドレスです。`admin` ユーザーのデフォルトパスワードは「admin」です。`user` ユーザーのデフォルトパスワードは「user」です。最初のログイン時に、デフォルトのログイン情報を変更するように求められます。

Cisco WAE Live 7.1.x から Cisco WAE Live 7.6.0 へのアップグレード

始める前に

このアップグレードを実行するには、Cisco WAE 7.1 以降がインストールされている必要があります。Cisco WAE 6.4.x のインストールについては、[WAE 6.4.10 以降の Live データの WAE Live 7.1.x への移行 \(32 ページ\)](#) を参照してください。

ステップ 1 Web サーバーと `mld` を停止します。

```
# wae-live-stop
```

ステップ 2 Cisco WAE Live 7.6.0 をインストールします。詳細については、[Cisco WAE Live のインストール \(30 ページ\)](#) を参照してください。

ステップ 3 アップグレードを実行します。

```
# mld -action upgrade
```

ステップ 4 Web サーバーと `mld` を起動します。

```
# wae-live-start
```

WAE 6.4.10 以降の Live データの WAE Live 7.1.x への移行

Live 7.1 では、Live ソフトウェアを Live 6.4.10 と同じ場所にインストールします。

始める前に

- **WAE 6.4.10 以降から WAE 7.1.x にのみデータを移行できます。**以前の WAE 6.x リリースがインストールされている場合は、WAE 7.1.x のアップグレードに進む前に、少なくとも WAE 6.4.10 にアップグレードしておく必要があります。
- WAE Live 7.1.x およびデータストアは、WAE 6.4.x とは別のマシンにインストールする必要があります。インストール手順については、[Cisco WAE Live のインストール \(30 ページ\)](#) を参照してください。さらに、WAE Live 7.1.x のインストールディレクトリおよびデータストア (mld) オプションは、WAE 6.4.x のインストールに使用されたものと同じディレクトリパスおよび mld オプションを使用する必要があります。たとえば、WAE 6.4.10 が /opt/cariden にインストールされている場合、別のサーバーの /opt/cariden に WAE Live 7.1.x もインストールする必要があります。CPU、メモリ、ストレージなどの mld パラメータも同じ値にする必要があります。既存の mld パラメータを表示するには、config.xml ファイルを調べます。
- この手順を実行する前に、WAE Live 7.1.x データストアをインストールする必要があります。データストアのインストール手順については、[WAE Live データストアのインストール \(36 ページ\)](#) を参照してください。
- WAE 7.1.x にアップグレードした後も、同じ WAE 6.4.x のユーザー ID (UID) とグループ ID (GID) を引き続き使用する必要があります。

ステップ 1 WAE Live 7.1 ソフトウェアを WAE Live 6.4.10 と同じ場所にインストールします。

WAE Live 6.4.10 で CARIDEN_ROOT = /opt/cariden の場合、WAE Live 7.1 を /opt/cariden にインストールします。

Live 7.1 のライセンスが設定されていることを確認してください。

ステップ 2 WAE Live 7.1 サーバーで、WAE Live 6.4.10 と同じパラメータを使用して mld をインストールします。

例：

パラメータ

```
-cpus
-mem
-storage
-mldata
-datastore
-backup
```

ステップ 3 WAE Live 7.1.x サーバーから、Web サーバーを停止します。

```
embedded_web_server -action stop
```

ステップ 4 WAE Live 6.4.x Live サーバーから、次の手順を実行します。

a) Web サーバーを停止します。

```
service wae-web-server stop
```

b) WAE Live データストアをバックアップします。次に例を示します。

```
ml_backup -L 0
```

ステップ 5 WAE Live 7.1.x サーバーから、次の手順を実行します。

a) WAE Live データストアをバックアップします。次に例を示します。

```
ml_backup -L 0
```

b) `$CARIDEN_ROOT/software/mld/current/scripts/sqlhosts.ml` でパラメータを編集します。

例 :

WAE Live 6.4.x サーバーでは、`sqlhosts.ml` に次のものがあります。

```
ml_remote onsoctcp 172.131.130.112 9089
mltcp      onsoctcp 127.0.0.1      9088
ml         onipcshm 127.0.0.1      dummy
```

WAE Live 7.1.x サーバーで、`sqlhosts.ml` ファイルを次のように変更します。

```
ml_remote onsoctcp <Live71_mld_IP_address> <Live71_port>
mltcp      onsoctcp 127.0.0.1      9088
ml         onipcshm 127.0.0.1      dummy
```

ステップ 6 データストアディレクトリ (`config.xml` の属性) が同じであることを確認します。WAE 7.1.x サーバーに不足しているファイルがある場合は、`touch` コマンドを使用して同じ名前でもゼロサイズのファイルを作成し、新しく作成されたファイルのファイル許可を 600 に変更します。

例 :

WAE Live 6.4.10 サーバーで、次の手順を実行します。

```
[cariden@wodl114 archives]$ cd $CARIDEN_ROOT/software/mld/current/data/
[cariden@wodl114 data]$ ls -la

total 63591328

drwxr-xr-x  2 cariden caridenstaff      4096 Dec  5 12:55 .
drwxr-xr-x 10 cariden caridenstaff      4096 Dec  5 12:35 ..
-rw-----  1 cariden caridenstaff 2147483648 Dec  5 13:01 catdbs001
. . . . .
-rw-----  1 cariden caridenstaff 4294967296 Dec  5 14:31 sbpace000
-rw-----  1 cariden caridenstaff 2147483648 Dec  5 14:54 tempdbs000
-rw-----  1 cariden caridenstaff 2147483648 Dec  5 14:54 tempdbs001
-rw-----  1 cariden caridenstaff 4294967296 Dec  5 14:54 tsdbs000
```

```
-rw----- 1 cariden caridenstaff 4294967296 Dec  5 14:54 tsdbs001
-rw----- 1 cariden caridenstaff 4294967296 Dec  5 14:31 tsdbs002000
-rw----- 1 cariden caridenstaff 4294967296 Dec  5 14:31 tsdbs002001
```

例：

WAE Live 7.1.x サーバーでは、不足しているファイルは `tsdbs002000` と `tsdbs002001` です。

```
$CARIDEN_ROOT/software/mld/current/data
[cariden@wod1113 data]$ ls -la
total 46814024
drwxr-xr-x  2 cariden caridenstaff    4096 Dec  5 01:56 .
drwxr-x--- 10 cariden caridenstaff     143 Dec  5 01:44 ..
-rw-----  1 cariden caridenstaff 2147483648 Dec  5 22:52 catdbs001
. . . . .
-rw-----  1 cariden caridenstaff 4294967296 Dec  5 22:50 sbospace000
-rw-----  1 cariden caridenstaff 2147483648 Dec  5 22:52 tempdbs000
-rw-----  1 cariden caridenstaff 2147483648 Dec  5 22:52 tempdbs001
-rw-----  1 cariden caridenstaff 4294967296 Dec  5 22:52 tsdbs000
-rw-----  1 cariden caridenstaff 4294967296 Dec  5 22:52 tsdbs001
```

次に、WAE Live 7.1.x が WAE Live 6.4.10 と同じ内容になるように、ゼロサイズのファイルを作成します。

```
touch tsdbs002000
touch tsdbs002001
chmod go-r tsdbs002000
chmod go-r tsdbs002001
```

ステップ 7 WAE Live 6.4.x データ ストア バックアップ ファイルを WAE Live 7.1.x サーバーにコピーします。

例：

WAE Live 6.4.x の場合

```
scp $CARIDEN_ROOT/software/mld/current/backups/fullbackups/wod1114_1_L0
user@live71:$CARIDEN_ROOT/software/mld/current/backups/fullbackups/
```

ステップ 8 コピーした WAE 6.4.x データ ストア バックアップ ファイルの名前を WAE 7.1.x バックアップファイルに変更します。

例：

WAE Live 7.1.x バックアップファイルの名前が `wod1113_1_L0` である場合は、

```
mv $CARIDEN_ROOT/software/mld/current/backups/fullbackups/wod1114_1_L0
$CARIDEN_ROOT/software/mld/current/backups/fullbackups/wod1113_1_L0
```

ステップ 9 WAE 7.1.x mld を停止し、データストアのバックアップを復元します。

```
mld -action stop
ml_restore -directory $CARIDEN_ROOT/software/mld/current/backups/fullbackups
```

ステップ 10 健全性チェックを実行します。このプロセスにはしばらく時間がかかる場合があります。

```
mld -sanity all
```

ステップ 11 mld を再起動します。

```
mld -action restart
```

ステップ 12 WAE Live 6.4.x サーバーから、次の手順を実行します。

a) config.xml ファイルをチェックして、次の属性が特定のディレクトリに設定されているかどうかを確認します。設定されていない場合、その属性は MLData 属性の下で指定され、デフォルトのパスは \$CARIDEN_ROOT/data/mldata/ です。

- AppData
- バックアップ (Backup)
- Map.ArchivePath
- ReportData

これらの属性が設定されている場合は、それぞれのディレクトリから WAE Live 7.1.x サーバーの同じディレクトリにデータをコピーします。

b) 上記の属性が設定されていない場合は、tar コマンドを使用してそれぞれのディレクトリをすべてパックし、mldata.tar を WAE Live 7.1.x サーバーの同じディレクトリにコピーします。

例 :

```
tar -cvf mldata.tar appdata/ archives/ customdata/ jobs/ plans/ reports/
```

mldata.tar を WAE Live 7.1.x サーバーにコピーします。

```
scp mldata.tar cariden@wodl113:$CARIDEN_ROOT/data/mldata/
```

WAE Live 7.1.x サーバーで、MLdata プロパティ値がある場所に移動します。次に例を示します。

```
cd $CARIDEN_ROOT/data/mldata/
tar -xvf mldata.tar
```

ステップ 13 WAE Live 6.4.x サーバーから、次の内容を \$CARIDEN_ROOT/etc から WAE Live 7.1.x の同じパスにコピーします。

- \$CARIDEN_ROOT/etc/config/config.xml
- \$CARIDEN_ROOT/etc/matelive
- \$CARIDEN_ROOT/etc/user_manager

例 :

```
# tar -cvf etc.tar config/ matelive/ user_manager/
config/
```

```

config/config.xml
config/config.xml.bak
matelive/
user_manager/
user_manager/auth.db.properties
user_manager/auth.db.script

# scp etc.tar <WAE_64x_host>:$CARIDEN_ROOT/etc/.

```

WAE Live 7.1.x サーバーから、\$CARIDEN_ROOT/etc ディレクトリをコピーします。

```

# cd $CARIDEN_ROOT/etc
# tar -xvf etc.tar

```

ステップ 14 WAE Live 7.1.x Web サーバーとデータストアを起動します。

```
wae-live-start
```

Cisco WAE Live データストア

このセクションでは、Cisco WAE Live データストアのインストール、アップグレード、バックアップ、および復元の方法について説明します。また、m1_purge ツールを使用してデータを消去する方法についても説明します。

インストール時にデフォルトが使用された場合、\$CARIDEN_HOME は /opt/cariden/software/live/current と同じです。

WAE Live データストアのインストール

次の手順では、m1d_tool を使用して Cisco WAE Live データストアをインストールする方法について説明します。m1d ツールは、m1d サーバーと空のデータストアディレクトリの両方をインストールします。

始める前に

- パフォーマンスを向上させるには、-datastore オプションで指定されたディレクトリ用に別の ext2 パーティションを作成します。
- 作成する実稼働環境のタイプを確認しておきます。



- (注)
- データストアを作成してしまうと、インストールオプション (ユーザー名を含む) の変更は難しくなります。
 - 「デモ」データストアは、パイロット用です。「デモ」で開始した場合、実稼働環境に移行する際にデータストアを再作成する必要があり、「デモ」データストア内のデータは失われます。

• `-demo` または `-storage`、`-cpu` および `-memory` オプションが必要です。mld コマンドとオプションの詳細については、[mld オプション \(37 ページ\)](#) を参照してください。

ステップ 1 WAE Live が実行されている場合は、Web サーバーを停止します。

```
# wae-live-stop
```

ステップ 2 適切な mld コマンドを入力して、データストアをインストールします。mld コマンドとオプションについては、[mld オプション \(37 ページ\)](#) を参照してください。

例 1 : インストール中にプロンプトを表示し、サイズに関する推奨事項を取得するコマンド :

```
# mld -installchk
```

例 2 : デモデータストアのサイズで mld をインストールするコマンド :

```
# mld -action install -demo true
```

例 3 : 小規模な mld サーバーを `$CARIDEN_ROOT/data/matelive` にインストールし、2つの CPU を予約し、542 GB のディスクストレージを予約し、2.2 GB (2,200 MB) のメモリを割り当てるコマンド :

```
# mld -action install -mldata $CARIDEN_ROOT/data/matelive -cpus 2 -storage 1:1:540 -memory 200:55:2000
```

ステップ 3 mld と Web サーバーを起動します。

```
# wae-live-start
```

mld オプション

オプション	説明	デフォルト
<code>-version</code>	データストアのバージョンを表示します。	

オプション	説明	デフォルト
-action	<p>install : 新しい mld サーバーとデータストアをインストールし、mld サーバーを起動します。</p> <p>upgrade : 既存の mld サーバーを更新し、mld サーバーを起動します。</p> <p>start : mld サーバーを起動する別の方法です。</p> <p>stop : mld サーバーを停止する別の方法です。</p> <p>status : mld サーバーのステータスを表示する別の方法です。</p> <p>restart : mld サーバーを停止してから再起動する別の方法です。</p>	デフォルトのインストールディレクトリ \$CARIDEN_ROOT/software/mld/current
-installchk	インストール中に表示され、サイズに関する推奨事項を提供します。	
-action install でのみ使用 (オプションが指定されていない場合、インストールでは -installchk と同じタスクが実行されます。)		
-demo true	デモデータストアをインストールします。 (注) -demo および -storage オプションの両方が使用されている場合、-demo が優先されます。	
-storage <n:n:n>	予想されるデータストアサイズに基づいてディスクとメモリを割り当てます。<n:n:n> はデータ : インデックス : 時系列 (GB 単位) です。詳細および推奨値については、-installchk および -verbose オプションを使用してください。 (注) データストアがデモサイズより大きい場合は、-action install を使用するときこのオプションが必要です。	

オプション	説明	デフォルト
-memory <n:n:n>	データストアの要求されたメモリを割り当てます。<n:n:n>はデータ：インデックス：時系列（GB 単位）です。詳細および推奨値については、-installchk および -verbose オプションを使用してください。	
-mldata <directory>	すべてのアプリケーションデータが格納されるディレクトリを指定します。このディレクトリには、データストア、レポート出力、およびその他のアプリケーションデータが含まれます。	\$CARIDEN_ROOT/data/mldata
-datastore <directory>	データストアが初期化されるディレクトリを指定します。 このディレクトリは、一度設定すると変更できません。ただし、シンボリックリンクの使用は可能です。	\$CARIDEN_ROOT/data/mldata/datastore
-cpus <#>	データストアと mld サーバー用の CPU の数を予約します。	全 CPU の半分
-action install または -action upgrade でのみ使用します		
-mld <directory>	mld サーバーがインストールされているディレクトリを指定します。 このディレクトリは、一度設定すると変更できません。ただし、シンボリックリンクの使用は可能です。	\$CARIDEN_ROOT/software/mld/current
-backup <directory>	データストアのバックアップを保存するディレクトリを指定します。	\$CARIDEN_ROOT/data/mldata/backup

データストアのバックアップ

Cisco WAE Live は、プランファイルからの時系列派生データをバックアップします。トランザクションログや、その他の WAE Live データ（アプリケーションデータ、レポートデータなど）はバックアップされません。

バックアップに必要な容量は、インストールサイズとシステムの稼働時間によって異なります。

ベストプラクティス

- 別のディスクドライブにバックアップを実行するか、バックアップの完了後に別の物理デバイスにバックアップをコピーします。
- トラフィックのピーク時間外にバックアップを実行します。
- mld サーバーとデータストアを最初にインストールするときに、別の物理ディスクにバックアップディレクトリを設定します。この設定により、すべてのバックアップのデフォルトのバックアップディレクトリが設定されます。

```
mld -action install -backup <backup_directory>
```

- バックアッププロセスではデータストアのコピーが作成されますが、アプリケーションデータやレポートデータなど、他の Cisco WAE Live データはバックアップされません。したがって、ある程度定期的に、バックアップされないデータを別の物理ディスクなどの安全な場所にコピーします。
- 少なくとも週単位または月単位で完全なバックアップを実行し、その間に大量の増分バックアップを実行します。
- 手動バックアップを実行する代わりに、cron ジョブから ml_backup を呼び出します。
- スケジュールが重ならないように、一度に1つのバックアップのみを実行します。同時バックアップの実行はサポートされていません。それぞれのバックアップの間は少なくとも1時間は空けるようにしてください。完了したら、バックアップが完了したことを1時間以内に確認します。

バックアップ手順



注意 以前のmldインストールディレクトリを削除すると、すべてのデータが削除される可能性があります。現在の場所を確認するには、コマンド `mld -diag -c | egrep ROOTPATH` を入力します。

ml_backup ツールを使用すると、複数のレベルでバックアップを実行してディスク領域を節約できます。OS ファイルシステムのバックアップを使用してデータストアの復元はできません。ml_backup ツールを使用して、データストアの復元に使用する完全なバックアップを実行します。

ml_backup を実行して、いつでも手動バックアップを実行できます。ただし、バックアップレベルを初めて使用するときは、この順序でバックアップを実行する必要があります。



(注) データストアと Webサーバーの両方を実行したままにします。

シーケンス	Enter	説明
1	ml_backup または ml_backup -L 0	レベル 0：すべてをバックアップします。
2	ml_backup -L 1	レベル 1：最新のレベル 0 バックアップが実行されてからのすべてをバックアップします。
3	ml_backup -L 2	レベル 2：最新のレベル 1 バックアップが実行されてからのすべてをバックアップします。



- (注) プランファイルプロセスを頻繁に実行する大規模なシステムの場合、レベル 0 バックアップの間に使用できるレベル 1 およびレベル 2 の増分バックアップは少なくなります。レベル 1 またはレベル 2 のバックアップが利用できない場合、次のエラーが表示されます。

```
Archive failed - The existing level-0 backup for DBspace rootdbs is too old to allow any
incremental backup.
```

このエラーが表示されたら、レベル 0 バックアップを実行します。

すべてのデフォルトを使用してバックアップを実行するには、ml_backup と入力します。このツールは、デフォルトのバックアップディレクトリを使用して、完全バックアップを作成します。

- デフォルトのバックアップディレクトリを上書きするには、-directory オプションを使用します。デフォルトのバックアップディレクトリは \$CARIDEN_ROOT/data/mldata/backup です。
- 別のバックアップレベルを設定するには、-L オプションを使用します。

次の例では、バックアップディレクトリを \$CARIDEN_ROOT/data/waelive/backups に設定し、最後のレベル 0 バックアップが実行された後の新しいデータのみをバックアップします。このバックアップは、デフォルトのレベル 0 を使用して ml_backup を 1 回実行したことを前提としています。

```
ml_backup -directory /data/waelive/backups -L 1
```

データストアの復元

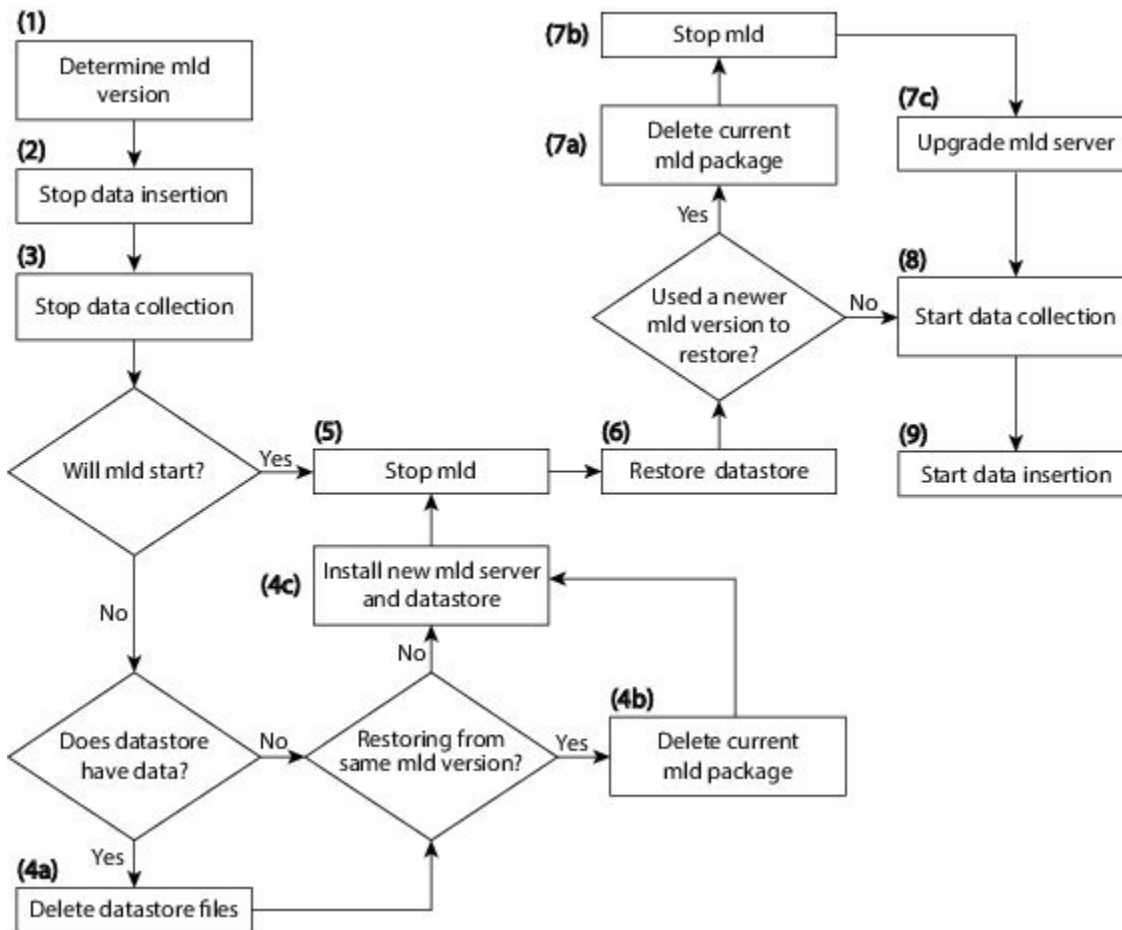
データストアの復元を試みる前に、次の点に注意してください。

- データストアを復元するには、データストアのバックアップが必要です。データストアのバックアップについては、「[データストアのバックアップ](#)」セクションを参照してください。

- 適切なディスクとディスク領域があることを確認してください。たとえば、データが破損している場合は、新しいディスクが必要になります。スペースの問題により復元する場合は、既存のディスクにさらにスペースを追加します。
- 単一のデバイス設定である場合、WAE Live データストアの復元中にデータの収集が中断されます。
- バックアップデータストアが別のデバイスにある場合は、次の前提条件を満たしていることを確認します。
 - 両方のデバイスのユーザー名とユーザー ID (uid) は同じである必要があります。
 - バックアップデータストア名は、名前の一部としてホスト名を使用します。バックアップデータストア名の、このホスト名の部分は、復元先のデバイスのホスト名と同じになるように変更する必要があります。

例：バックアップデータストア名は `akdobi.acme.com_1_L0` です。データストアが復元されているデバイスのホスト名は `akgudei.acme.com` です。この場合、バックアップデータストア名を `akgudei.acme.com_1_L0` に変更します。

図 1: データストアの復元ワークフロー



データストアからのデータの削除

ml_purge ツールは、指定されたタイムスタンプより前のすべてのデータを削除します。

始める前に

ml_purge を実行する前に、挿入が実行されていないことを確認します (ml_insert_ctl -status)。操作中に ml_purge によって作成されたロックが原因で、挿入が失敗する場合があります。スケジューラされた挿入を防ぐために、スケジューラを一時停止する必要がある場合があります (ml_insert_ctl -disable-scheduler)。

ml_purge を実行する方法は次のとおりです。

```
# ml_purge <timestamp>
```

ここでの <timestamp> は UTC 形式 (year-month-day T hour:minutes) です。次に例を示します。

```
# ml_purge 2017-01-31T00:00
```

Cisco WAE Live ライセンスのインストール

Cisco WAE は、Cisco Smart Licensing と従来のライセンスをサポートしています。ライセンスのタイプに基づいて、次のいずれかの手順を使用します。

従来のライセンスのインストール

始める前に

サーバーに Cisco WAE Live ライセンスがあることを確認します。

ステップ 1 Cisco WAE Live が実行されていない場合は、開始します。

```
# embedded_web_server -action start
```

ステップ 2 サポートされているブラウザ (https://<server_IP>:8443) で、Cisco WAE Live UI を開始します。デフォルトのユーザー名は「admin」、デフォルトのパスワードは「admin」です。

ステップ 3 [システム (System)] > [ライセンス (Licenses)] を選択します。

ステップ 4 [従来のライセンスのアップロード (Upload Traditional License)] をクリックします。

ステップ 5 [ライセンスの選択 (Select Licenses)] をクリックします。

ステップ 6 場所を参照するか、ライセンスファイルの名前 (.lic 拡張子) を入力し、[開く (Open)] をクリックします。

ステップ 7 [ライセンスのアップロード (Upload license)] をクリックします。

ステップ 8 Cisco WAE Live を再起動します。

```
# embedded_web_server -action restart
```

スマートライセンスのインストール

始める前に

WAE サーバーに Cisco スマートライセンスがあることを確認します。

ステップ 1 Cisco WAE Live が実行されていない場合は、開始します。

```
# embedded_web_server -action start
```

ステップ 2 サポートされているブラウザ (https://<server_IP>:8443) で、Cisco WAE Live UI を開始します。デフォルトのユーザー名は「admin」、デフォルトのパスワードは「admin」です。

ステップ 3 [システム (System)] > [ライセンス (Licenses)] を選択します。

ステップ 4 [スマート ソフトウェア ライセンシングの有効化 (Enable Smart Software Licensing)] をクリックします。

ステップ 5 WAE サーバー名または IP アドレス、ポート、ユーザー名、およびパスワードを入力し、[有効化 (Enable)] をクリックします。

ステップ 6 Cisco WAE Live を再起動します。

```
# embedded_web_server -action restart
```

次のタスク

Cisco WAE Live の使用を開始して、プランファイルを収集できます。Cisco WAE 7.6.0 からプランファイルを収集するには、[設定 (Settings)] > [データ送信元 (Data Source)] に移動し、[7.6.0 リモートアーカイブ (7.6.0 Remote Archive)] オプションをクリックします。適切な Cisco WAE 7.6.0 ネットワークおよびサーバーの詳細を入力します。Cisco WAE Live の使用については、[Cisco WAE Live ユーザーガイド](#)を参照してください。

認証失敗エラーのトラブルシューティング

WAE コレクタの管理者パスワードが変更された場合、パスワードの変更が原因でライセンスチェックが失敗するため、WAE Live の UI に Live コンポーネントが表示されません。この問題を解決するには、次の手順に従います。

ステップ 1 WAE サーバーの管理パスワードを変更します (新しいパスワード)。

ステップ 2 WAE Live サーバーで、新しいパスワードを使用して `license_install` ツールを実行します。

次に例を示します。

```
license_install -smart-lic-host <WAE-server-IP> -smart-lic-port 2022 -smart-lic-username admin  
-smart-lic-password <new-password>
```

- ステップ 3** WAE Live サーバーで、\$HOME/.cariden/etc に移動し、**MATE_Smart.lic** ファイルを \$CARIDEN_ROOT/etc に移動します。
- ステップ 4** WAE Live サーバーで `license_check` コマンドを実行します。
認証エラーメッセージが表示されなくなったことを確認します。
- ステップ 5** WAE Live UI から、[設定 (Settings)]> [データ送信元 (Data Source)]に移動し、パスワードを更新します。
-



第 5 章

セキュリティ

- [主要なセキュリティ概念 \(47 ページ\)](#)
- [証明書のインストール \(49 ページ\)](#)

主要なセキュリティ概念

製品のセキュリティの最適化を目指す管理者は、次のセキュリティ概念をよく理解しておく必要があります。

HTTPS

Hypertext Transfer Protocol Secure (HTTPS) では、チャンネルを介して送信されるデータの暗号化に、セキュア ソケット レイヤ (SSL) またはその後続の標準規格である Transport Layer Security (TLS) が使用されます。SSL で複数の脆弱性が見つかったため、では現在 TLS のみがサポートされています。



(注) TLS は大まかに SSL と呼ばれることが多いため、本ガイドでもこの表記に従います。

SSL は、プライバシー、認証、およびデータ整合性を組み合わせることで、クライアントとサーバーの間のデータ転送を保護します。これらのセキュリティメカニズムを有効にするために、SSL は証明書、秘密キー/公開キー交換ペア、および Diffie-Hellman 鍵共有パラメータを使用します。

SSL 証明書

SSL 証明書と秘密キー/公開キー ペアは、ユーザー認証および通信パートナーの ID 検証に使われるデジタル ID の一種です。VeriSign や Thawte などの認証局 (CA) は、エンティティ (サーバーまたはクライアント) を識別するための証明書を発行します。クライアントまたはサーバー証明書には、発行認証局の名前とデジタル署名、シリアル番号、証明書が発行されたクライアントまたはサーバーの名前、公開キー、および証明書の有効期限が含まれます。CA は、1 つ以上の署名証明書を使用して SSL 証明書を作成します。各署名証明書には、CA 署名の作

成に使用される照合秘密キーがあります。CAは署名付き証明書（公開キーが埋め込まれている）を簡単に入手できるようにしているため、誰でもその証明書を使用して、SSL証明書が実際に特定のCAによって署名されたことを確認できます。

一般に、証明書の設定には次の手順が含まれます。

1. サーバーの ID 証明書を生成する。
2. サーバーに ID 証明書をインストールする。
3. 対応するルート証明書をクライアントまたはブラウザにインストールする。

実行する必要がある具体的なタスクは、ご利用の環境によって異なります。

1 方向 SSL 認証

これは、クライアントが適切なサーバー（中間サーバーではなく）に接続していることを保証する必要がある場合に使用される認証方法で、オンラインバンキングの Web サイトなどのパブリックリソースに適しています。認証は、クライアントがサーバー上のリソースへのアクセスを要求したときに開始されます。リソースが存在するサーバーは、その ID を証明するために、サーバー証明書（別名 SSL 証明書）をクライアントに送信します。クライアントは受信したサーバー証明書を、クライアントまたはブラウザにインストールする必要がある別の信頼できるオブジェクト（サーバールート証明書）と照合して検証します。サーバーの検証後、暗号化された（つまりセキュアな）通信チャネルが確立されます。ここで、サーバーは HTML フォームへの有効なユーザー名とパスワードの入力を求めます。SSL 接続が確立された後にユーザークレデンシャルを入力すると、未認証の第三者による傍受を防ぐことができます。最終的に、ユーザー名とパスワードが受け入れられた後、サーバー上に存在するリソースへのアクセスが許可されます。



(注) クライアントは複数のサーバーとやり取りするために、複数のサーバー証明書を格納する必要がある場合があります。



クライアントにルート証明書をインストールする必要があるかどうかを判断するには、ブラウザの URL フィールドでロック アイコンを探します。通常このアイコンが表示される場合は、必要なルート証明書がすでにインストール済みであることを示します。多くの場合、これはより大きいいずれかの認証局 (CA) によって署名されたサーバー証明書に該当します。一般的なブラウザではこれらの CA からのルート証明書が含まれているからです。

クライアントがサーバー証明書に署名した CA を認識しない場合は、接続がセキュリティで保護されていないことを意味します。これは必ずしも大きな問題ではなく、接続するサーバーの ID が検証されていないことを示しているだけです。この時点で、次の 2 つの操作のいずれかを実行できます。1 つは必要なルート証明書をクライアントまたはブラウザにインストールできます。ブラウザの URL フィールドにロック アイコンが表示された場合は、証明書が正常にインストールされたことを意味します。もう 1 つは、クライアントに自己署名証明書をインストールできることです。信頼できる CA によって署名されたルート証明書とは異なり、自己署名証明書は作成者である個人またはエンティティによって署名されます。自己署名証明書を使用して暗号化チャネルを作成できますが、接続するサーバーの ID が検証されていないため、固有のリスクが伴うことを理解しておいてください。

証明書のインストール

このセクションには、Cisco WAE サーバーへのセキュリティ証明書のインストール、Cisco WAE の調整されたメンテナンス、および Cisco WAE Live に関する情報が含まれています。

Cisco WAE サーバーの証明書のインストール

Cisco WAE にはデフォルトの証明書が付属しています。この証明書は「信頼された CA」からのものではないため、ブラウザには保護されていない接続の警告が表示されます。これは予期されている動作です。この警告は、適切な認証局 (CA) 発行の証明書を適用することで削除できます。

ステップ 1 プライベートサーバーキーを作成し、安全な場所に保存します。次に例を示します。

```
# openssl genrsa -out server.key 2048
```

ステップ 2 証明書署名要求 (CSR) を作成します。CA は、CSR を使用して、Web サイトを安全であると識別する証明書を作成します。次に例を示します。

```
# openssl req -sha256 -new -key server.key -out server.csr
```

ステップ 3 CSR を認証局に送信して、証明書 (server.crt など) を取得します。

(注) WAE は、PEM フォーマットの server.crt のみをサポートします。サーバー証明書を DER から PEM フォーマットに変換するには、次のコマンドを使用できます。

```
sudo openssl x509 -inform der -in <input certificate filename> -out <output certificate filename>
```

ステップ 4 server.key および server.crt ファイルの場所を示すように <key-file/> および <cert-file/> 要素を変更し、<WAE_run_directory>/wae.conf を変更します。

ステップ 5 Cisco WAE サーバを再起動します。

```
# sudo supervisorctl stop wae:*
# sudo supervisorctl start wae:*
```

Cisco WAE Live の証明書のインストール

Cisco WAE Live には、証明書が信頼されていないことをブラウザに示す、デフォルトの証明書が含まれています。これは予期されている動作です。この警告は、適切な CA 発行の証明書を適用することで削除できます。

Cisco WAE Live の CA 証明書をインストールするには、次の手順を実行します。

始める前に



(注) この手順は、Cisco WAE Live 7.1.1 以降にのみ適用されます。

- このタスクを実行するには、Cisco WAE ユーザー権限を持つ管理者である必要があります。
- ツール keytool は、jdk/jre で展開されます。keytool path が PATH に含まれていることを確認してください。



(注) 前の例は、シェルが sh、ksh、または bash の場合に適用できます。他のシェルには同等のコマンドを使用します。

- ログアウトして再度ログインするか、適切なプロファイルのファイル名を使用して次のコマンドを入力します。

```
# source ~/.profile
```

ステップ 1 任意の認証局 (CA) から証明書を取得するには、証明書署名要求 (CSR) を作成する必要があります。CSR の作成手順は、次のとおりです。

a) デフォルトの証明書を削除します。次に例を示します。

```
# keytool -storepass changeit -delete -alias cisco -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore
```

b) 自己署名証明書を作成します。次に例を示します。

```
# keytool -storepass changeit -genkey -alias tomcat -keyalg RSA -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore
```

- c) CSR を作成します。次に例を示します。

```
# keytool -storepass changeit -certreq -keyalg RSA -alias tomcat -file certreq.csr -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore
```

- d) CSR ファイルを認証局に提出して、証明書を取得します。

- e) (オプション) Cisco WAE Live を再起動して、新しい証明書をすぐに使用します。

```
# embedded_web_server -action stop
# embedded_web_server -action start
```

ステップ2 証明書をインストールします。

- a) 証明書を取得したCAから、チェーン証明書（ルート証明書とも呼ばれます）をダウンロードします。

- b) チェーン証明書をキーストアにインポートします。

```
# keytool -storepass changeit -import -alias root -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore -trustcacerts -file
<filename_of_the_chain_certificate>
```

- c) 新しい証明書をインポートします。

```
# keytool -storepass changeit -import -alias tomcat -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore -file <your_certificate_filename>
```

- d) Cisco WAE Live を再起動します。

```
# embedded_web_server -action stop
# embedded_web_server -action start
```

LDAP サーバーの証明書のインストール

Cisco WAE は、Lightweight Directory Access Protocol (LDAP) を使用して外部ユーザーの認証と許可をサポートしています。

LDAPS プロトコルを使用するには、SSL 証明書を取得してキーストアに追加します。

- ステップ1** 次のコマンドを使用して、自己署名証明書を `cert.pem` ファイルに保存します。

```
# openssl s_client -connect <ldap-host>:<ldap-ssl-port> </dev/null 2>/dev/null | sed -n
'/^-----BEGIN/,/^-----END/ { p }' > cert.pem
```

- ステップ2** `WAE_RUN` ディレクトリから次のコマンドを実行して、デフォルトのキーストアパスを取得します。

```
# $WAE_ROOT/lib/exec/test-java-ssl-conn <ldap-host> <ldap-ssl-port> 2>1 | grep "trustStore is:"
```

上記のコマンドを実行すると、証明書が取得されるディレクトリを見つけるのに役立ちます。次のようなディレクトリになります。

```
trustStore is: /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.102-4.b14.e17.x86_64/jre/lib/security/cacerts
```

- ステップ3** 次のコマンドを使用して、証明書をデフォルトのキーストアにインポートします。

```
# sudo keytool -import -keystore <default-key-store-path> -storepass changeit -noprompt -file cert.pem
```

EPN-M サーバーの証明書のインストール

L1 収集に Cisco Evolved Programmable Network Manager (Cisco EPN Manager) エージェントを使用する場合は、証明書をインストールします。

ステップ 1 次のコマンドを使用して、自己署名証明書を `cert.pem` ファイルに保存します。

```
# openssl s_client -connect <epnm-host>:<epnm-port> </dev/null 2>/dev/null | sed -n  
'/^-----BEGIN/,/^-----END/ { p }' > cert.pem
```

ステップ 2 次のコマンドを使用して、デフォルトのキーストアパスを取得します。通常、デフォルトのキーストアパスは `/etc/pki/java/cacerts` です。

```
# $WAE_ROOT/lib/exec/test-java-ssl-conn <epnm-host> <epnm-port> 2>1 | grep "trustStore is:"
```

ステップ 3 次のコマンドを使用して、証明書をデフォルトのキーストアにインポートします。

```
# sudo keytool -import -keystore <default-key-store-path> -storepass changeit -noprompt -file cert.pem
```



第 6 章

次のステップ

次のトピックでは、Cisco WAE で開始するために実行する次の手順について説明します。操作を実行するには、WAE UI、WAE エキスパートモード、または WAE CLI にアクセスします。詳細については、『Cisco WAE ユーザーガイド』を参照してください。

- [Cisco WAE にログインします。](#) (53 ページ)
- [ネットワークモデルの構築](#) (55 ページ)

Cisco WAE にログインします。

このセクションでは、使用可能な Cisco WAE インターフェイス（Cisco WAE UI、エキスパートモード、および Cisco WAE CLI）にログインする方法について説明します。これらのインターフェイスの詳細については、[Cisco WAE ユーザーガイド](#)を参照してください。

Cisco WAE UI へのログイン

以下の手順に従って、Cisco WAE Web UI にログインします。

始める前に

適切なサービスがすべて動作していることを確認します。すべてのサービスはインストール後に自動的に開始されます。Cisco WAE の開始または停止の方法については、[Cisco WAE の開始と停止](#) (24 ページ) を参照してください。

ステップ 1 サポートされているブラウザのいずれかを起動します。[Cisco WAE のインストレーション要件](#) (5 ページ) を参照してください。

ステップ 2 ブラウザのアドレスバーに `https://server-ip:8443` と入力します。ここで、`server-ip` は Cisco WAE がインストールされたサーバーの IP アドレスです。

Cisco WAE ユーザーインターフェイスに [ログイン (Login)] ウィンドウが表示されます。

ステップ 3 Web UI のユーザー名とパスワードを入力します。

ステップ 4 [ログイン (Login)] をクリックします。

ホームページが表示され、Web UI を使用できるようになりました。

次のタスク

Cisco WAEにログインすると、ネットワークトポロジ収集を開始して、ネットワークモデルを作成できます。ネットワークモデルの作成については、[Cisco WAE ユーザー ガイド](#)を参照してください。

エキスパートモードへのログイン

エキスパートモードにアクセスする前に、WAE UI にログインする必要があります。

始める前に

適切なサービスがすべて動作していることを確認します。すべてのサービスはインストール後に自動的に開始されます。Cisco WAE の開始または停止の方法については、[Cisco WAE の開始と停止 \(24 ページ\)](#) を参照してください。

-
- ステップ 1** サポートされているブラウザのいずれかを起動します。[Cisco WAE のインストール要件 \(5 ページ\)](#) を参照してください。
- ステップ 2** ブラウザのアドレスバーに **https://server-ip:8443** と入力します。ここで、*server-ip* は Cisco WAE がインストールされたサーバーの IP アドレスです。
- Cisco WAE UI に [ログイン (Login)] ウィンドウが表示されます。
- ステップ 3** Cisco WAE UI のユーザー名とパスワードを入力します。
- ステップ 4** [ログイン (Login)] をクリックします。
- ホームページが表示され、Web UI を使用できるようになりました。
- ステップ 5** Cisco WAE UI の右上隅で、ツールアイコンをクリックしてエキスパートモードにアクセスします。

次のタスク

Cisco WAEにログインすると、ネットワークトポロジ収集を開始して、ネットワークモデルを作成できます。[Cisco WAE ユーザー ガイド](#)を参照してください。

WAE CLI へのログイン

WAE CLI にログインするには、次の手順を実行します。

-
- ステップ 1** WAE ランタイムディレクトリに移動し、`wae_cli` と入力します。


```
# wae_cli -u admin
admin@wae#
```

(注) `wae_cli --help` と入力して、すべての WAE CLI オプションを表示できます。

ステップ2 (オプション) 設定の操作を有効にするには、コンフィギュレーション モードに切り替えます。

```
admin@wae# config
admin@wae%#
```

例

次に例を示します。

```
waerun# wae_cli -u admin
admin@wae# config
admin@wae%#
```

ネットワークモデルの構築

このトピックには、ネットワークモデルを構築するために必要なタスクについての高度な説明が記載されています。詳細については、[Cisco WAE ユーザー ガイド](#)を参照してください。

1. デバイス認証グループ、SNMP グループ、およびネットワーク プロファイル アクセスを構成します。
2. (オプション) エージェントを設定します。この手順は、XTC、LAG およびポートインターフェイス、マルチレイヤ、Netflow、またはテレメトリ情報を収集する場合にのみ必要です。
3. トポロジ NIMO を使用して、集約されたネットワークと送信元を設定します。
4. デマンド、トラフィック、レイアウト、インベントリなどの追加の収集を設定します。
5. 収集をいつ実行するかをスケジュールします。
6. プランファイルが定期的に保存されるアーカイブファイルシステムの場所と間隔を設定します。
7. (オプション) Cisco WAE アプリケーションでプランファイルを表示します。



第 7 章

Cisco WAE のアンインストール

- [Cisco WAE のアンインストール \(57 ページ\)](#)

Cisco WAE のアンインストール

この手順では、Cisco WAE のインストールを削除する方法について説明します。



- (注) Cisco WAE のインスタンスが複数インストールされているはずですが、アンインストール手順を実行するときは、正しい Cisco WAE インストールディレクトリとランタイムディレクトリを削除していることを確認してください。

ステップ 1 すべての WAE プロセスを停止するには、次のコマンドを使用します。

```
# sudo supervisorctl stop wae:*
```

ステップ 2 親ディレクトリに移動し、Cisco WAE インストールディレクトリとランタイムディレクトリを削除します。

```
# rm -rf <wae_installation_directory>
# rm -rf <wae_run_time_directory>
```

例

次に例を示します。

```
# sudo supervisorctl stop wae:*
# cd
# rm -rf waeinstall
# rm -rf waerun
```


翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。