HITACHI
Inspire the Next

Hitachi Automation Director

インストールガイド

3021-9-104-80

対象製品
Hitachi Automation Director 8.5.0

輸出管理に関する注意
本マニュアル固有の技術データおよび技術は、米国輸出管理法、および関連の規制を含む米国の輸出管理法の対象となる場合があり、その他の国の輸出または輸入規制の対象となる場合もあります。読者は、かかるすべての規制を厳守することに同意し、マニュアルおよび該当製品の輸出、再輸出、または輸入許可を取得する責任があることを了解するものとします。

商標類

This product includes software developed by the University of California, Berkeley and its contributors.
This software contains code derived from the RSA Data Security Inc. MD5 Message-Digest Algorithm, including various modifications by Spyglass Inc., Carnegie Mellon University, and Bell Communications Research, Inc (Bellcore).
Regular expression support is provided by the PCRE library package, which is open source software, written by Philip Hazel, and copyright by the University of Cambridge, England. The original software is available from ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/
This product includes software developed by Ralf S. Engelschall <rse@engelschall.com> for use in the mod_ssl project (http://www.modssl.org/).
This product includes software developed by IAIK of Graz University of Technology.
This product includes software developed by the Java Apache Project for use in the Apache JServ servlet engine project (http://java.apache.org/).
This product includes software developed by Daisuke Okajima and Kohsuke Kawaguchi (http://relaxngcc.sf.net/).
This product includes software developed by Andy Clark.

本製品は，米国 EMC コーポレーションの RSA BSAFE(R)ソフトウェアを搭載しています。
Java is a registered trademark of Oracle and/or its affiliates.

発行
2016 年 12 月 3021-9-104-80

目次

はじめに

このマニュアルでは、Hitachi Automation Director（HAD）のインストールと構成の方法を説明します。

- 対象読者
- マニュアルの構成
- マイクロソフト製品の表記について
- 関連マニュアル
- このマニュアルで使用している記号
- KB（キロバイト）などの単位表記について

対象読者

このマニュアルは、ストレージ環境内のストレージ、サービス、およびアプリケーションを担当するストレージ管理者を対象としています。

マニュアルの構成

このマニュアルは，次に示す章と付録から構成されています。

第 1 章　概要

Automation Director の概要について説明しています。

第 2 章　Hitachi Automation Director をインストールする

クラスタと非クラスタ両方の環境における Microsoft® Windows®、または非クラスタ環 境における Red Hat Enterprise Linux（RHEL）での、Hitachi Automation Director のインストール方法 について説明してします。

第 3 章　Automation Director を構成する

Automation Director を構成する方法について説明してします。

第 4 章　Hitachi Automation Director を削除する

Hitachi Automation Director を削除する方法について説明してします。

付録 A　Hitachi Automation Director のファイルの場所とポート

Hitachi Automation Director インストールの一部として作成されるすべてのフォルダ が含まれています。

付録 B　hcmds64keytool ユーティリティを使用する

hcmds64keytool ユーティリティは、使用方法について説明しています。

マイクロソフト製品の表記について

このマニュアルでは、マイクロソフト製品の名称を次のように表記しています。

表記	製品名
Internet Explorer	次の製品を区別する必要がない場合の表記です。 • Microsoft ® Internet Explorer ® • Windows ® Internet Explorer ®
Windows	次の製品を区別する必要がない場合の表記です。 • Microsoft ® Windows Server ® 2008 • Microsoft ® Windows Server ® 2008 R2 • Microsoft ® Windows Server ® 2012 • Microsoft ® Windows Server ® 2012 R2
Windows Server 2008	次の製品を区別する必要がない場合の表記です。 • Microsoft ® Windows Server ® 2008 • Microsoft ® Windows Server ® 2008 R2

表記	製品名
Windows Server 2012	次の製品を区別する必要がない場合の表記です。 • Microsoft ® Windows Server ® 2012 • Microsoft ® Windows Server ® 2012 R2

関連マニュアル

- *Hitachi Automation Director ユーザーズガイド* , 3021-9-103
- *Hitachi Automation Director Service Builder ユーザーズガイド* , 3021-9-106
- *Hitachi Automation Director メッセージ*, 3021-9-107
- Hitachi Command Suite ドキュメント
- Hitachi Tuning Manager ドキュメント

このマニュアルで使用している記号

このマニュアルでは、次のような表記規則を使用しています。

規則	説明
太字	リスト項目の中で強調する語を示します。
[]	ウィンドウのタイトル、メニュー、メニューオプション、ボタン、フィールド、ラベルなど、ウィンドウ内のテキストを示します。 例：[OK] をクリックします。
斜体	• マニュアルのタイトルまたはテキスト内で強調する語を示します。 • 変数を示します。これは、ユーザーが入力する実際のテキストのプレースホルダ、またはシステムから出力されるプレースホルダです。例： `pairdisplay -g group` （この変数の規則の例外については、山括弧の説明を参照してください。）
Monospace	画面に表示されるテキスト、またはユーザーが入力するテキストを示します。例： `pairdisplay -g oradb`
<>（山括弧）	次のような場合に、変数を示します。 • 変数は、周囲のテキストや他の変数から明確には区切られません。例： `Status-<report-name><file-version>.csv` • 見出しに変数が含まれる場合。
[ ]（角括弧）	オプションの値を示します。例：[a | b]は、a または b を選択できる、あるいはどちらも省略できることを示します。
{ }（波括弧）	必須の値または予期される値を示します。例：{ a | b }は、a または b のどちらかを選択する必要があることを示します。
|（縦線）	2 つ以上のオプションまたは引数から選択できることを示します。例： [a | b]は、a または b を選択できる、あるいはどちらも省略できることを示します。 { a | b }は、a または b のいずれかを選択する必要があることを示します。

KB（キロバイト）などの単位表記について

物理ストレージ容量値（ディスクドライブ容量など）は、以下の値に基づいて計算されます。

物理的容量の単位	値
1 キロバイト（KB）	1,000（10^3）バイト
1 メガバイト（MB）	1,000 KB または $1,000^2$ バイト
1 ギガバイト（GB）	1,000 MB または $1,000^3$ バイト
1 テラバイト	1,000 GB または $1,000^4$ バイト
1 ペタバイト（PB）	1,000 TB または $1,000^5$ バイト
1 エクサバイト（EB）	1,000 PB または $1,000^6$ バイト

論理ストレージ容量値（論理デバイス容量など）は、以下の値に基づいて計算されます。

論理容量単位	値
1 ブロック	512 バイト
1 シリンダー	メインフレーム：870 KB Open 系 • OPEN-V：960 KB • その他：720 KB
1 KB	1,024（2^{10}）バイト
1 MB	1,024 KB または $1,024^2$ バイト
1 GB	1,024 MB または $1,024^3$ バイト
1 TB	1,024 GB または $1,024^4$ バイト
1 PB	1,024 TB または $1,024^5$ バイト
1 EB	1,024 PB または $1,024^6$ バイト

1

概要

この章には、以下の情報が記載されています。

1.1 製品の概要

Hitachi Automation Director は、ストレージおよびデータセンター管理者向けの、エンドツーエンドのストレージプロビジョニングプロセスを自動化および単純化するためのツールとなるソフトウェアソリューションです。この製品の基本要素は、サービステンプレートと呼ばれる、事前にパッケージ化されたオートメーションテンプレートです。これらの事前構成テンプレートは特定の環境とプロセスに合わせてカスタマイズされ、リソースプロビジョニングなどの複雑なタスクを自動化するサービスを作成します。構成が済むと、Automation Director は既存の Hitachi Command Suite アプリケーションと連携して、既存のインフラストラクチャサービスを利用することによって、共通のインフラストラクチャ管理タスクを自動化します。

Automation Director は、次のような機能を備えています。

- オートメーションサービスの作成を容易にする、事前構成されたサービステンプレート
- さまざまなストレージクラスのボリュームのインテリジェントなプロビジョニングのためのオートメーションサービス
- 定義されたサービスへのロールベースのアクセス
- インフラストラクチャーグループから最も性能の高いプールを選択し、プール情報を各タスクに提供してボリューム使用量の詳細を指定する、性能ベースのプール選択
- すべてのオートメーションサービスに割り当てて共有できる共通のサービス管理属性

1.2 関連する Hitachi Command Suite 製品について

Hitachi Automation Director は、以下のコンポーネントを含む Hitachi Command Suite の一部です。

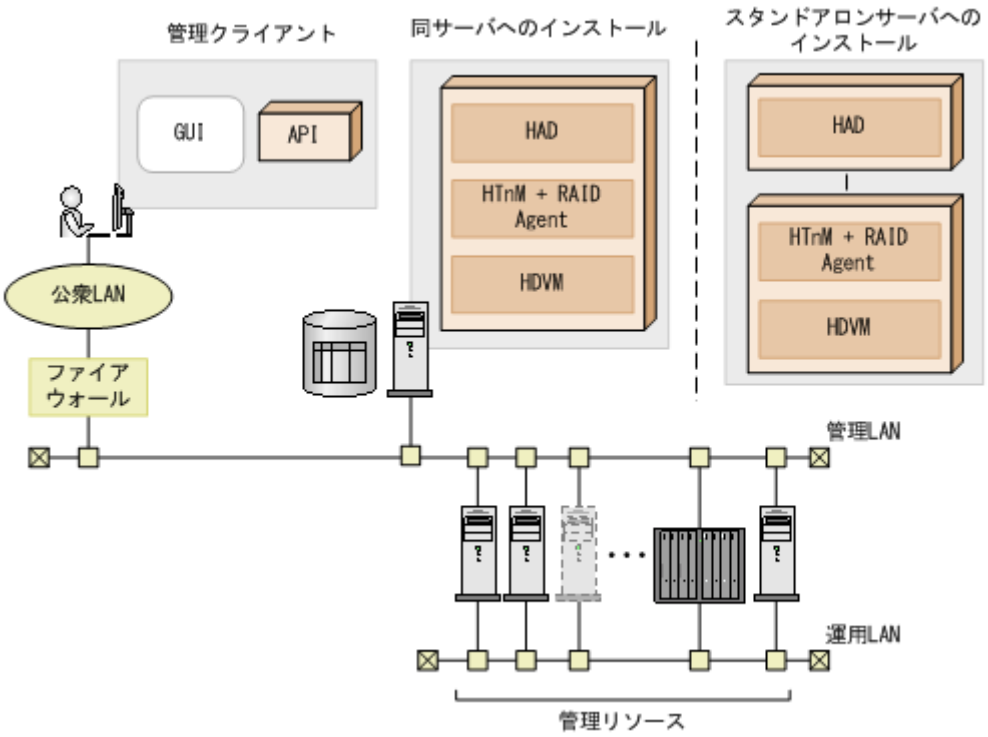
- Hitachi Device Manager
- Hitachi Tiered Storage Manager
- Hitachi Dynamic Link Manager
- Hitachi Replication Manager
- Hitachi Tuning Manager
- Hitachi Global Link Manager
- Hitachi Compute Systems Manager

Automation Director を他の Hitachi Command Suite 製品と同じサーバにインストールすると、共通の設定でユーザーとセキュリティを管理できます。また、Automation Director を Device Manager が稼働しているサーバにインストールすると、2 つの製品によって管理されるホスト情報が自動的に同期されるため、ホスト管理の作業効率が向上します。

メモ　Automation Director と Device Manager の両方を使用した場合に同期されるのはホスト情報のみで、他の種類のリソースの情報は同期されません。

1.3 Hitachi Automation Director システム構成

Hitachi Automation Director 環境をセットアップするには、2 つの方法があります。次の図は、基本的なシステム構成を示しています。



基本的なシステム構成環境は、次のいずれかとしてセットアップできます。

- Hitachi Automation Director はスタンドアロンの製品としてインストールされ、その他の Hitachi Command Suite 製品はインストールされません。
- Hitachi Automation Director と Hitachi Device Manager（HDVM）が同じサーバにインストールされます。

メモ `hcmds64prmset` コマンドを使用して、同一サーバ構成をスタンドアロンセットアップに変更することもできます。

前提となる Hitachi Command Suite 製品

次の表に、サポートされる Hitachi Command Suite 製品を示します。

製品	バージョン
Hitachi Device Manager	8.1.4
Hitachi Tuning Manager*	8.1.1
Hitachi Replication Mananger**	8.1.4
* Tuning Manager は、Tuning Manager の性能データを利用して、Automation Director が一連のプールまたはアレイにわたってプロビジョニングを行うときにインテリジェントなプール選択を実行できるようにする場合に必要です。 ** Hitachi Replication Manager は、Clone（Shadow Image）、Snapshot（Thin Image）、および Copy Topology サービスを使用する場合にのみ必要です。複数の Device Manager／Replication Manager を使用する構成の場合、1 つの Replication Manager だけが通常モードで実行されるように設定してください。残りの Replication Manager は、常に保守モードで動作する必要があります。	

性能ベースのプール選択

インテリジェントプロビジョニングサービスの性能ベースのプール選択を使用することができます。性能ベースのプール選択を有効にするには、次のファイルの設定をチェックします。

Install location of HDvM¥HicommandServer¥config¥tuningmanager.properties

このファイルには、Device Manager から Tuning Manager に接続するためのプロパティが含まれています。

Device Manager サーバと Tuning Manager サーバが同じマシンにインストールされている場合、システムは次の設定で実行します。

- htnm.servers=1
- htnm.server.0.host=localhost
- htnm.server.0.protocol=http
- htnm.server.0.port=22015

追加情報については、『*Hitachi Command Suite システム構成ガイド*』の次のセクションを参照してください。

- 6.2 ストレージシステムの性能情報を収集するために必要な設定
- A.14 Tuning Manager との連携に関するプロパティ（tuningmanager.properties ファイル）

Automation Director によってサポートされる Hitachi Device Manager サーバの最大数

Automation Director がサポートできる HDvM サーバの最大数は 50 です。追加情報については、Automation Director のリリースノートを参照してください。

1.3.1 前提となる Hitachi Command Suite 製品

次の表に、サポートされる Hitachi Command Suite 製品を示します。

製品	バージョン
Hitachi Device Manager	8.1.4
Hitachi Tuning Manager*	8.1.1
Hitachi Replication Mananger**	8.1.4
* Tuning Manager は、Tuning Manager の性能データを利用して、Automation Director が一連のプールまたはアレイにわたってプロビジョニングを行うときにインテリジェントなプール選択を実行できるようにする場合に必要です。 ** Hitachi Replication Manager は、Clone（Shadow Image）、Snapshot（Thin Image）、および Copy Topology サービスを使用する場合にのみ必要です。複数の Device Manager／Replication Manager を使用する構成の場合、1 つの Replication Manager だけが通常モードで実行されるように設定してください。残りの Replication Manager は、常に保守モードで動作する必要があります。	

1.3.2 性能ベースのプール選択

インテリジェントプロビジョニングサービスの性能ベースのプール選択を使用することができます。性能ベースのプール選択を有効にするには、次のファイルの設定をチェックします。

Install location of HDvM¥HicommandServer¥config¥tuningmanager.properties

このファイルには、Device Manager から Tuning Manager に接続するためのプロパティが含まれています。

Device Manager サーバと Tuning Manager サーバが同じマシンにインストールされている場合、システムは次の設定で実行します。

- htnm.servers=1
- htnm.server.0.host=localhost
- htnm.server.0.protocol=http
- htnm.server.0.port=22015

追加情報については、『*Hitachi Command Suite システム構成ガイド*』の次のセクションを参照してください。

- 6.2 ストレージシステムの性能情報を収集するために必要な設定
- A.14 Tuning Manager との連携に関するプロパティ（tuningmanager.properties ファイル）

メモ　Automation Director がサポートできる HDvM サーバの最大数は 50 です。追加情報については、Automation Director のリリースノートを参照してください。

1.4 Hitachi Automation Director のインストールと構成のワークフロー

次の図は、Hitachi Automation Director のインストールと構成を含む、ワークフローの概要を示しています。

このガイドには、システムのインストール、セットアップ、管理、および保守に関する情報が含まれています。管理 GUI を使用したサービスの作成、管理、および自動プロビジョニングの詳細については、『*Hitachi Automation Director ユーザーズガイド*』を参照してください。

2

Hitachi Automation Director をインストールする

この章では、クラスタと非クラスタ両方の環境における Microsoft® Windows®、または非クラスタ環境における Red Hat Enterprise Linux（RHEL）での、Hitachi Automation Director のインストール方法について説明します。

2.1 インストールの前提条件

Hitachi Automation Director をインストールする前に、以下のタスクを完了してください。

- .NET Framework 3.5.1 が管理サーバにインストールされていることを確認します。インストールするには、管理サーバで実行しているオペレーティングシステムの手順に従います。.NET Framework をインストールする前に、前提条件となる IIS のバージョンがサーバにインストールされていることを確認します。
- 環境と管理サーバがすべてのハードウェアおよびソフトウェア要件を満たしていることを確認します。システム要件の詳細については、Automation Director のリリースノートを参照してください。
- Automation Director によって使用されるポートが使用可能であることを確認します。管理サーバのポートが他の Hitachi Command Suite 製品によって使用されておらず、競合が存在しないことを確認します。ポートが別の Hitachi Command Suite 製品によって使用されていた場合、どちらの製品も正しく動作しないことがあります。
- 関連マシンの名前を解決します。
- このガイドに含まれているインストールおよび構成タスクを完了するために、Windows 管理者権限が取得されていることを確認します。
- サーバ上のセキュリティ監視、ウイルス検出、プロセス監視ソフトウェアを無効にします。
- Windows のサービスまたは開いているコマンドプロンプトを閉じます。
- サーバが他の Hitachi Command Suite 製品を実行している場合は、それらの製品のサービスを停止します。
- サーバのシステム時刻が正しいことを確認します。Hitachi Command Suite が別のサーバにインストールされている場合は、Automation Director サーバの時刻を Hitachi Command Suite サーバに同期します。
- RHEL 環境の Automation Director インストールに必要なファイアウォール例外を、手動で再追加します。これらの例外は、インストール時に自動的に再構成されません。

関連参照

- 2.1.2 名前解決設定を変更する
- 2.1.1 サーバ時刻を変更する

2.1.1 サーバ時刻を変更する

Automation Director サーバのオペレーティングシステム時刻設定が Hitachi Command Suite 管理サーバと同期していることが重要です。

Automation Director のタスクおよびアラート発生時刻は、管理サーバの時刻設定に基づきます。したがって、サーバのオペレーティングシステムの時刻設定が正確かどうかを確認することが重要です。必要に応じて、Automation Director をインストールする前にリセットしてください。Hitachi Command Suite 共通コンポーネントおよび Hitachi Command Suite 製品サービスが実行しているときに Automation Director サーバの時刻を変更した場合、Automation Director が正しく動作しないことがあります。

NTP など、サーバの時刻を自動的に調整するサービスを使用する場合は、次のようにサービスを構成する必要があります。

- サービスにより時刻の不一致が検出されたときに調整されるよう、設定を構成します。

- 特定の時刻差を超えない範囲内で時刻設定の調整が行われるようにします。最大範囲値に基づいて、時刻差が固定範囲を超えないように頻度を設定してください。

特定の時刻差の範囲内で時刻を調整できるサービスの例としては、Windows Time サービスがあります。

メモ 米国またはカナダのタイムゾーンで Automation Director を実行するときには、新しい夏時間（DST）ルールをサポートするように管理サーバのオペレーティングシステムを構成する必要があります。サーバがサポートを提供しないかぎり、Automation Director は新しい DST ルールをサポートできません。

サーバの時刻を自動的に調整する機能を使用できない場合や、システム時刻を手動で変更する場合は、以下のステップを実行します。

1. Hitachi Command Suite 共通コンポーネントと、以下を含むすべての Hitachi Command Suite 製品のサービスを停止します。
   - HBase 64 Storage Mgmt Web Service
   - HBase 64 Storage Mgmt Web SSO
   - HBase 64 Storage Mgmt SSO Service
   - HCS Device Manager Web Service
   - HBase 64 Storage Mgmt Comm Server
   - HiCommand Suite Tuning Manager
   - HiCommand Performance Reporter
   - HCS Tuning Manager REST Application
   - HAutomation Engine Web Service
   - Device Manager Server Service
   - Tiered Storage Manager Server
2. 管理サーバの現在時刻を記録してから、時刻をリセットします。
3. サービスを再起動する時間を決めます。

- マシンの時刻を戻した場合（サーバの時刻が進んでいた場合）は、サーバのクロックが記録した時刻（変更を加えたときのサーバの時刻）を示すまで待ってから、マシンを再起動します。
- マシンの時刻を進めた場合は、すぐにマシンを再起動します。

Automation Director 管理サーバが正しい時刻を反映していることを確認します。

2.1.2 名前解決設定を変更する

Automation Director と Hitachi Command Suite を 2 台の異なるマシンにインストールした場合は、クライアントに接続する Automation Director サーバの名前を解決する必要があります。

Automation Director がインストールされているマシンの名前も解決する必要があります。

Automation Director を Hitachi Command Suite と同じマシンにインストールした場合は、Automation Director にアクセスするためにブラウザを実行するマシンの名前を解決する必要があります。

`user.httpsd.conf` ファイルの最初の行で `ServerName` プロパティとして設定されている管理サーバのホスト名からシステムが IP アドレスを解決できるように、構成設定を更新します。次のコマンドを実行して、IP アドレスがホスト名に解決されることを確認します。`ping` *`management-server-host-name`*.

2.1.3 ポートの衝突を回避する

Automation Director を新しくインストールする前に、管理サーバ上で Automation Director が使用するポートが他の製品によって使用されていないことを確認してください。ポートが別の製品によって使用されていた場合、どちらの製品も正しく動作しないことがあります。

必要なポートが使用中でないことを確認するには、**netstat** コマンドを使用します。

2.2 Hitachi Automation Director をインストールする

このマニュアルでは、単体インストールメディアから製品インストーラを使用して Hitachi Automation Director をインストールする方法を説明します。

メモ　Automation Director を他の Hitachi Command Suite 製品とともにインストールする場合は、システムがすべての製品のインストール要件を満たしていることを確認してください。

Windows 環境

操作手順

1. システムがインストール前のチェックリストに記載されているすべての管理サーバ前提条件を満たしていることを確認します。
2. .NET Framework 3.5 SP1（3.5.1）がインストールされていることを確認します。
3. サーバが Hitachi Command Suite 共通コンポーネントを使用する製品を実行している場合は、以下のサービスを停止します。
   - HBase 64 Storage Mgmt Web Service
   - HBase 64 Storage Mgmt Web SSO
   - HBase 64 Storage Mgmt SSOService
   - HCS Device Manager Web Service
   - HBase 64 Storage Mgmt Common Server
   - HiCommand Suite Tuning Manager
   - HiCommand Suite Performance Reporter
   - HCS Tuning Manager REST Application
   - HAutomation Engine Web Service
   - Device Manager Server Service
   - Tiered Storage Manager Server Service
4. インストールメディアを DVD ドライブに挿入します。
5. インストールウィザードを起動します。
   ＜*Automation Director のインストールメディア*＞¥HAD_SERVER¥setup.exe を実行します。
6. 画面の指示に従って、必要な情報を指定します。
   ほとんどの場合、デフォルトのインストール選択項目を受け入れてください。
   ［Install Complete］ウィンドウが開きます。
7. ［Finish］をクリックします。

メモ

- SSL 通信が有効な環境、または Hitachi Command Suite 共通コンポーネントのポート番号が変更された環境に Automation Director をインストールする場合、[Install Complete] ウィンドウで [After the installation finishes, start the Hitachi Command Suite GUI] チェックボックスを選択してもグラフィカルユーザーインタフェースが起動しないことがあります。
- この問題が発生した場合は、変更された管理サーバ情報をチェックしてから、Web ブラウザのアドレスバーに Automation Director の URL を入力して、インタフェースを起動します。

操作結果

これで、Automation Director がインストールされます。

Linux 環境

`install.sh` を実行して、Automation Director をインストールします。

Linux での Automation Director のインストール先ディレクトリは、デフォルトでは`/opt/HiCommand/Automation` です。

関連参照

- 2.4 インストール後のタスク

2.3 クラスタ環境で Hitachi Automation Director をインストールする

ここでは、クラスタ環境での Hitachi Automation Director の新規インストールと構成について説明します。

メモ Automation Director は、Windows クラスタ環境にのみインストールできます。Linux 環境では、Automation Director のクラスタ化はサポートされていません。

2.3.1 クラスタ環境での Automation Director の使用について

Hitachi Automation Director を使用するときには、Microsoft Windows Server Failover Clustering を使用してフェイルオーバー管理サーバをセットアップすることで信頼性を高めることができます。

クラスタ環境で Automation Director を使用するときには、次のように、1 台の Automation Director サーバをアクティブノードに、もう 1 台をスタンバイノードに指定します。

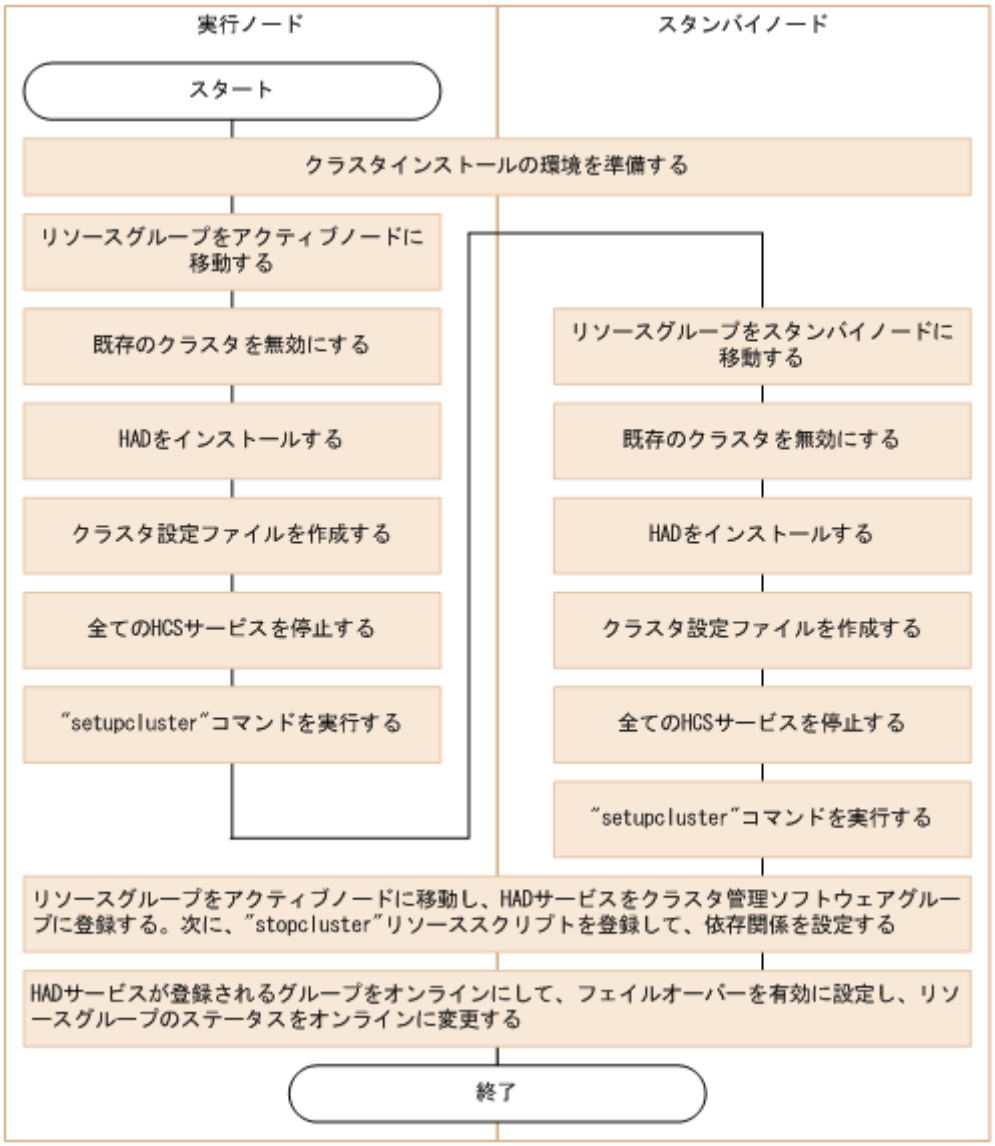
- アクティブノード
  アクティブノードは、クラスタを使用するシステムでサービスを実行しているホストです。
  障害が発生した場合、クラスタサービスがフェイルオーバーを実行し、スタンバイノードがシステムリソースの操作を引き継ぐため、サービスは中断されません。
- スタンバイノード
  スタンバイノードは、障害発生時にアクティブノードからシステムリソースの操作を引き継ぐホストです。

メモ アクティブノードがスタンバイノードにフェイルオーバーした場合、実行中のタスクは失敗するので、スタンバイノード上でタスクを再び実行する必要があります。

2.3.2 クラスタインストールワークフロー

Hitachi Automation Director をクラスタ構成でインストールするときには、一連のステップに従って、実行ノードとスタンバイノードを準備する必要があります。

以下に、クラスタ環境をセットアップするための一般的なワークフローを示します。



メモ　初めて Hitachi Automation Director をクラスタ環境にインストールするとき、または非クラスタ環境からクラスタ環境に移行するときには、クラスタ内のすべてのノードが同じディスク構成を持つことと、すべての Hitachi Command Suite 製品が各ノードの同じ場所（ドライブ名、パスなどを含む）にインストールされていることを確認してください。

メモ　既にクラスタ構成でインストールされている Hitachi Automation Director のアップグレードを行うときには、更新されたインストールを実行する前に、リソーススクリプトを無効にする必要があります。

関連タスク

- 2.3.4 アクティブノードで Hitachi Automation Director クラスタ化をセットアップする
- 2.3.5 スタンバイノードで Hitachi Automation Director クラスタ化をセットアップする

2.3.3 クラスタ管理ソフトウェアを使用してクラスタ構成をチェックする

クラスタ環境で Hitachi Automation Director をセットアップするときには、クラスタ管理ソフトウェアを使用して現在の環境設定を確認し、追加の設定を構成する必要があります。

クラスタ環境で Hitachi Automation Director をセットアップする前に、クラスタ環境ソフトウェアを使用して、以下の項目をチェックします。

- 他の Hitachi Command Suite 製品のサービスが登録されているグループが存在するかどうかをチェックします。
  Hitachi Command Suite のサービスが登録されているグループが既に存在する場合は、そのグループを使用します。グループが、Hitachi Command Suite 製品に関するリソースのみで構成されていることを確認します。
  Hitachi Command Suite のサービスが登録されているグループが存在しない場合は、クラスタ管理ソフトウェアを使用して、Hitachi Automation Director のサービスを登録するグループを作成します。

  メモ グループ名に次の文字を使用することはできません：!" % &) * ^ | ; = , < >

- サービスを登録するグループに、アクティブノードとスタンバイノード間で継承できる共有ディスクとクライアントアクセスポイントが含まれていることを確認します。クライアントアクセスポイントは、クラスタ管理 IP アドレスと論理ホスト名です。
- クラスタ管理ソフトウェアを使用してリソースの割り当て、削除、および監視が問題なくできることを確認します。

クラスタ環境で使用されるサービスは、クラスタ管理ソフトウェアでグループとして登録することによってフェイルオーバーできます。これらのグループは、クラスタ管理ソフトウェアと OS のバージョンによって、「リソースグループ」や「roles」など異なる名前で呼ばれることがあります。

2.3.4 アクティブノードで Hitachi Automation Director クラスタ化をセットアップする

クラスタ構成のアクティブノード上の管理サーバで、Hitachi Automation Director の新しいインストールを完了することができます。

操作手順

1. クラスタ管理 IP アドレスと共有ディスクをオンラインにします。クラスタインストールのリソースグループがアクティブノードに移動されることを確認します。
2. ほかの Hitachi Command Suite 製品でクラスタ環境が構築されている場合は、次のコマンドを使用して、Hitachi Command Suite 製品のサービスが登録されるクラスタグループをオフラインにして、フェイルオーバーを無効にします。

   - Hitachi Command Suite 製品のバージョン 8.1.2 以降がインストールされていない場合：
     integrated-installation-media¥HCS¥ClusterSetup
     ¥hcmds64clustersrvstate /soff /r *HCS-cluster-group-name*

- Hitachi Command Suite 製品のバージョン 8.1.2 以降がインストールされている場合：
  HCS-Common-Component-installation-directory¥ClusterSetup
  ¥hcmds64clustersrvstate /soff /r *HCS-cluster-group-name*
  ここで、
  r - Hitachi Command Suite 製品のサービスが登録されるグループの名前を指定します。グループ名にスペースが含まれる場合は、グループ名を引用符（"）で囲む必要があります。たとえば、グループ名が HCS cluster の場合は、"HCS cluster"と指定します。

3. アクティブノード上の Hitachi Automation Director の新しいインストールを完了します。
   別の Hitachi Command Suite 製品がクラスタ環境に既に存在する場合は、Automation Director をインストールする前に、以下のことを確認してください。
   - 管理サーバの IP アドレスとして、論理ホストの IP アドレスを指定します。

   他の Hitachi Command Suite 製品がクラスタ環境に存在しない場合は、Automation Director をインストールする前に、以下のことを確認してください。
   - 管理サーバの IP アドレスとして、アクティブノードの IP アドレスを指定します。

   メモ クラスタ構成で既にセットアップされた環境で Hitachi Automation Director をアップグレードする場合は、更新インストールを実行する前に、リソースグループに登録されるスクリプトのフェイルオーバーを防止する必要があります。クラスタ管理ソフトウェアで、リソースグループに登録されるスクリプトを右クリックして、[[property]-[policy]] タブから、再起動しないようにリソースを設定します。

4. 使用する製品のライセンスを登録します。アクティブノードの IP アドレスにアクセスします。
5. クラスタ内で Hitachi Command Suite 製品を既に構成している場合、次のステップへスキップします。Automation Director がクラスタ内の最初の Hitachi Command Suite 製品である場合は、以下を実行します。
   a. 空白のテキストファイルに以下の情報を追加します。

   ```
   mode=online
   virtualhost=logical-host-name
   onlinehost=active-node-host-name
   standbyhost=standby-node-host-name
   ```

   メモ アクティブノードで、mode として online を指定する必要があります。

   ファイルを cluster.conf という名前で*HCS-Common-Component-installation-folder*¥conf に保存します。
6. 次のコマンドを使用して、Hitachi Command Suite 製品を確実に停止します。
   HCS-Common-Component-installation-folder¥bin¥hcmds64srv /stop/server AutomationWebService
7. **setupcluster /exportpath**ExportPath コマンドを実行します。ここで、Exportpath は絶対または相対ディレクトリパスを指定します。

関連タスク

- 2.3.5 スタンバイノードで Hitachi Automation Director クラスタ化をセットアップする

2.3.5 スタンバイノードで Hitachi Automation Director クラスタ化をセットアップする

アクティブノードでクラスタ化インストールを設定した後、クラスタ構成のスタンバイノード上の管理サーバで Hitachi Automation Director のインストールを完了できます。

操作手順

1. クラスタ管理ソフトウェアで、Hitachi Automation Director のリソースを含んでいるグループをスタンバイノードに移動します。グループを右クリックして［Move］を選択してから、［Select Node］または［Move this service or application to another node］を選択します。
2. ほかの Hitachi Command Suite 製品でクラスタ環境が構築されている場合は、次のコマンドを使用して、Hitachi Command Suite 製品のサービスが登録されるクラスタグループをオフラインにして、フェイルオーバーを無効にします。

   - Hitachi Command Suite 製品のバージョン 8.1.2 以降がインストールされていない場合：
     integrated-installation-media¥HCS¥ClusterSetup
     ¥hcmds64clustersrvstate /soff /r *HCS-cluster-group-name*
   - Hitachi Command Suite 製品のバージョン 8.1.2 以降がインストールされている場合：
     HCS-Common-Component-installation-directory¥ClusterSetup
     ¥hcmds64clustersrvstate /soff /r *HCS-cluster-group-name*
     ここで、
     r - Hitachi Command Suite 製品のサービスが登録されるグループの名前を指定します。グループ名にスペースが含まれる場合は、グループ名を引用符（"）で囲む必要があります。たとえば、グループ名が HCS cluster の場合は、"HCS cluster"と指定します。
3. スタンバイノード上の Hitachi Automation Director の新しいインストールを完了します。
   スタンバイノードに Hitachi Automation Director をインストールする前に、以下の要件に注意してください。

   - アクティブノードと同じ場所に Hitachi Automation Director をインストールする必要があります。
   - 他の Hitachi Command Suite 製品が既に存在し、クラスタ環境でアクティブな場合、管理サーバの IP アドレスとして論理ホスト名（クラスタ管理 IP アドレスに割り当てられる仮想ホスト名）を指定します。クラスタ環境に他の Hitachi Command Suite 製品がない場合、スタンバイノードの IP アドレスまたはホスト名を指定します。

   メモ　クラスタ構成で既にセットアップされた環境で Hitachi Automation Director をアップグレードする場合は、更新インストールを実行する前に、リソースグループに登録されるスクリプトのフェイルオーバーを防止する必要があります。クラスタ管理ソフトウェアで、リソースグループに登録されるスクリプトを右クリックして、［[property]-[policy]］タブから、再起動しないようにリソースを設定します。

4. 使用する製品のライセンスを登録します。
5. クラスタ内で Hitachi Command Suite 製品を既に構成している場合、次のステップへスキップします。Hitachi Automation Director がクラスタ内の最初の Hitachi Command Suite 製品である場合は、以下を実行します。
   a. 空白のテキストファイルに以下の情報を追加します。

```
mode=standby
virtualhost=logical-host-name
onlinehost=active-node-host-name
standbyhost=standby-node-host-name
```

ファイルを cluster.conf という名前で*HCS-Common-Component-installation-folder*¥conf に保存します。

メモ スタンバイノードで、mode として standby を指定する必要があります。

6. 次のコマンドを使用して、Hitachi Command Suite 製品を確実に停止します。
   hcmds64srv /stop /server AutomationWebService
7. **setupcluster /exportpath** コマンドを実行します。ここで、exportpath は、絶対または相対ディレクトリパスを指定します。

2.3.6 サービスを登録しクラスタインストールの初期設定を行う

Hitachi Automation Director をクラスタ構成のアクティブノードおよびスタンバイノードにインストールした後、以下のステップの説明に従ってサービスとスクリプトを登録し、クラスタ化をオンラインにできます。

操作手順

1. クラスタ管理ソフトウェアで、Hitachi Automation Director のリソースを含んでいるグループをアクティブノードに移動します。グループを右クリックして［Move］を選択してから、［Select Node］または［Move this service or application to another node］を選択します。
2. 次のコマンドを使用して、クラスタ管理ソフトウェアグループで Hitachi Automation Director サービスを登録します。
   HCS-Common-Component-installation-directory¥ClusterSetup
   ¥hcmds64clustersrvupdate /sreg /r *HCS-cluster-group-name* /sd *drive-letter-of-shared-disk* /ap *resource-name-for-client-access-point*
   ここで、
   r - Hitachi Automation Director を含む Hitachi Command Suite 製品のサービスが登録されるグループの名前を指定します。グループ名にスペースが含まれる場合は、グループ名を引用符（"）で囲む必要があります。たとえば、グループ名が HAD cluster の場合は、"HAD cluster"と指定します。
   sd - クラスタ管理ソフトウェアに登録される共有ディスクのドライブ名を指定します。このオプションに対して複数のドライブ名を指定することはできません。Hitachi Command Suite 製品のデータベースが複数の共有ディスクに分割されている場合は、各共有ディスクについて hcmds64clustersrvupdate コマンドを実行します。
   ap - クラスタ管理ソフトウェアに登録されるクライアントアクセスポイント用リソースの名前を指定します。
3. クラスタソフトウェアに対して **stopcluster** コマンドを実行するためのスクリプトリソースとして、以下を登録します。
   - **HAD-installation-folder¥bin¥stopcluster /prepare**

   スクリプトリソース名とスクリプト名は任意です。リソースがオフラインのときだけ **stopcluster** コマンドが実行されるように、スクリプトを構成します。具体的な詳細については、使用するクラスタソフトウェアのマニュアルを参照してください。

メモ Hitachi Automation Director をアップグレードする場合は、スクリプトを登録する必要はありません。ただし、リソースグループに登録されるスクリプトのフェイルオーバーを有効にする必要があります。クラスタ管理ソフトウェアで、リソースグループに登録されるスクリプトを右クリックして選択し、［[property]-[policy]］タブから、再起動するようにリソースを設定します。

4. クラスタ管理ソフトウェアで、リソーススクリプトを右クリックして選択し、[[property]-[Dependencies]]タブから依存関係を設定します。さらに、[HAutomation Engine HCS-cluster-group-name]を、スクリプトがオンラインになる前にオンラインにならなければならないリソースに指定する必要があります。
5. アクティブノードで、次のコマンドを使用して Hitachi Automation Director を含む Hitachi Command Suite サービスが登録されるグループをオンラインにして、フェイルオーバーを有効にします。

   HCS-Common-Component-installation-folder¥ClusterSetup

   ¥hcmds64clustersrvstate /son /r *HCS-cluster-group-name*

   ここで、

   r - Hitachi Automation Director を含む Hitachi Command Suite 製品のサービスが登録されるグループの名前を指定します。グループ名にスペースが含まれる場合は、グループ名を引用符（"）で囲む必要があります。たとえば、グループ名が HAD cluster の場合は、"HAD cluster"と指定します。
6. クラスタソフトウェアで、リソースグループのステータスを［online］に変更します。

2.4 インストール後のタスク

Automation Director のインストール後は、以下のインストール後のタスクを完了してください。

1. ユーザーアカウントを管理するサーバが SSL 通信を使用する場合、**hcmds64prmset** コマンドを実行して、サーバのポート番号を設定します（必要に応じて）。
2. 登録済み URL を確認します（推奨）。
3. Automation Director 管理サーバへのアクセスを確認します。
4. ライセンスを登録します。
5. System アカウントのパスワードを変更します（推奨）。
6. System アカウントのメールアドレスを設定します。
7. RMI 通信を有効にします。

   メモ　このステップは、Device Manager v8.1.4 を使用する場合のみ必要です。
8. HCS および HAD サービスを停止し、再開します（必要に応じて）。

登録済み URL の確認

次のコマンドを使用して、登録済み URL を確認します。

HCS-Common-Component-installation-folder¥bin¥hcmds64chgurl /list

URL 内のホスト名をチェックします。非クラスタ環境の場合、ホスト名は物理ホスト名でなければなりません。クラスタ環境の場合、ホスト名は論理ホスト名でなければなりません。登録済み URL が正しくなかった場合には、次のコマンドを使用して URL を変更します。

HCS-Common-Component-installation-folder¥bin¥hcmds64chgurl /change

http://incorrect-IP-address-or-host-name:port-number

http://correct-IP-address-or-host-name:port-number

2.4.1 インストールを確認する

インストールが完了したら、インストールが成功したことを Web ブラウザから確認してください。

操作手順

1. Automation Director によってサポートされている Web ブラウザを開きます。
2. アドレスバーに、Automation Director の URL を次の形式で指定します。
   http://*HAD-server-address* :22015/Automation/
   ログインウィンドウが開くので、管理サーバにアクセスできることを確認します。

2.4.2 ライセンスを登録する

最初にログオンするときには、有効なライセンスキーを指定する必要があります。

操作手順

1. ログインウィンドウの［ライセンス］をクリックします。
2. ライセンスキーを入力するか、ライセンスファイルの場所を参照して、［保存］をクリックします。

2.4.3 System アカウントのパスワードを変更する

System アカウントは、すべての Hitachi Command Suite 製品のユーザー管理および実行権限を持つデフォルトのアカウントです。Automation Director を初めてインストールするときには、System アカウントのパスワードを変更することをお勧めします。

操作手順

1. 管理クライアントから、次の認証情報を使用してログオンします。
   - **ユーザー ID:** system
   - **パスワード（デフォルト値）:** manager
2. ［管理］タブで、［プロファイル］をクリックします。
3. ［パスワード変更］をクリックし、必要なパスワードを入力して［OK］をクリックします。

操作結果

デフォルトのパスワードが変更されます。

2.4.4 Hitachi Command Suite および Automation Director のサービスを停止および開始する

Hitachi Command Suite および Automation Director はコマンドプロンプトからサービスを実行します。Hitachi Command Suite は、スタートメニューからでも停止および開始できます。

メモ　HAD サービスは、スタートメニューからは開始できません。

(1) 「スタート」メニューからすべてのサービスを停止および開始する

次の手順により、すべての Hitachi Command Suite サービスを停止および開始します。

操作手順

1. ［Start］－［All Programs］－［Hitachi Command Suite］－［Manage Services］を選択します。

2. ［Start - HCS］または［Stop - HCS］をクリックします。

(2)　コマンドプロンプトからすべてのサービスを停止および開始する（Windows）

次の手順により、すべての Hitachi Command Suite および Automation Director のサービスを停止および開始します。

操作手順

1. コマンドプロンプトで、C:¥Program Files¥HiCommand¥Base64¥bin に移動します。
2. サービスを停止するには、次のコマンドを入力します。
   hcmds64srv.exe /stop
   サービスを開始するには、次のコマンドを入力します。
   hcmds64srv.exe /start

(3)　コマンドプロンプトからすべてのサービスを停止および開始する（Linux）

次の手順により、すべての Hitachi Command Suite および Automation Director のサービスを停止および開始します。

操作手順

1. コマンドプロンプトで、/opt/HiCommand/Base64/bin に移動します。
2. サービスを停止するには、次のコマンドを入力します。
   hcmds64srv -stop
   サービスを開始するには、次のコマンドを入力します。
   hcmds64srv -start

(4)　コマンドプロンプトから Automation Director サービスのみ停止および開始する（Windows）

操作手順

1. C:¥Program Files¥HiCommand¥Base64¥bin に移動します。
2. サービスを停止するには、次のコマンドを入力します。
   hcmds64srv.exe /stop /server AutomationWebService
   サービスを開始するには、次のコマンドを入力します。
   hcmds64srv.exe /start /server AutomationWebService

(5)　コマンドプロンプトから Automation Director サービスのみ停止および開始する（Linux）

操作手順

1. /opt/HiCommand/Base64/bin に移動します。
2. サービスを停止するには、次のコマンドを入力します。
   hcmds64srv -stop -server AutomationWebService
   サービスを開始するには、次のコマンドを入力します。
   hcmds64srv -start -server AutomationWebService

2.4.5 RMI 通信を有効にする

HAD サービスを使用する前に、Replication Manager の RMI 通信を構成する必要があります。このステップは、Replication を使用するかどうかにかかわらず必要です。Replication Manager の RMI 通信

を有効にしなかった場合、Device Manager 接続は正しく機能せず、［管理］タブにリストされる接続ステータスはエラーを示します。

3

Automation Director を構成する

この章では、Automation Director を構成する方法について説明します。

3.1 管理サーバのシステム設定を変更する

ここでは、Automation Director 管理サーバのシステム設定の変更に関して説明します。

3.1.1 Automation Director のポート番号を変更する

Automation Director に使用するポート番号は、インストール後に必要に応じて変更できます。

操作手順

1. Automation Director を停止します。
2. Automation Director のプロパティを編集します。
   a. *Installation-folder-for-Hitachi-Command-Suite*¥Automation¥conf ¥command_user.properties を開きます。
   b. command.http.port の値を必要に応じて変更します。
3. Automation Director を開始します。
4. 管理サーバと管理クライアント間の通信に使用されるポート（デフォルトでは、22015/TCP または 22016/TCP）を変更した場合は、Automation Director にアクセスするための URL を変更してください。

3.1.2 ユーザーアカウントを管理するサーバの情報を変更する

必要に応じて、ユーザーアカウントを管理するサーバの情報を変更できます。

メモ　ユーザーアカウントは、接続先の Device Manager がインストールされているホスト上の共通コンポーネントによって管理されます。

操作手順

1. （Windows）Device Manager の HBase 64 Storage Mgmt Web Service に対して SSL が設定されていない場合は、このコマンドを実行します。

```
Common-Component-installation-folder¥bin¥hcmds64prmset /host Device-Manager-IP-address-or-host-name /port HBase-64-Storage-Mgmt-Web-Service-of-Device-Manager-non-SSLportnumber
```

2. （Windows）Device Manager の HBase 64 Storage Mgmt Web Service に対して SSL が設定されている場合は、このコマンドを実行します。

```
Common-Component-installation-folder¥bin¥hcmds64prmset /host Device-Manager-host-name /sslport HBase-64-Storage-Mgmt-Web-Service-of-Device-Manager-SSL-port-number
```

3. （Linux）Device Manager の HBase 64 Storage Mgmt Web Service に対して SSL が設定されていない場合は、このコマンドを実行します。

```
Common-Component-installation-folder/bin/hcmds64prmset -host Device-Manager-IP-address-or-host-name -port HBase-64-Storage-Mgmt-Web-Service-of-Device-Manager-non-SSLportnumber
```

4. （Linux）Device Manager の HBase 64 Storage Mgmt Web Service に対して SSL が設定されている場合は、このコマンドを実行します。

```
Common-Component-installation-folder/bin/hcmds64prmset -host Device-Manager-host-name -sslport HBase-64-Storage-Mgmt-Web-Service-of-Device-Manager-SSL-port-number
```

3.1.3 タスク処理エンジンによって使用されるポート番号を変更する

必要に応じて、Hitachi Automation Director のタスク処理エンジンによって使用されるポート番号を変更できます。タスク処理エンジンは Automation Director の内部コンポーネントであり、タスクプロセスの実行を受け持ちます。これらの個別プロセスは、通信ポートの使用を必要とします。

メモ　ポート番号を変更する前に、Hitachi Automation Director　GUI の［タスク］タブの［状態］列をチェックして、実行中のタスクがないことを確認してください。実行中、待機中、長期実行中ステータスのタスク、または停止しているプロセスが、ポート番号の変更による影響を受けないことを確認します。

操作手順

1. **hcmds64srv /stop** コマンドを実行して、Automation Director を停止します。
2. テキストエディタで%windir%¥system32¥drivers¥etc¥services ファイルを開き、jp1ajs3cdinetd で定義されているポート番号の値を変更します。
3. *Automation-Director-installation-folder*¥system¥AJS3CD¥conf¥ディレクトリに ajscd_DNA.properties という名前のファイルを作成して、次のエントリを追加します。
   ajscd.port_number=*port-number-value-from-step-2*
4. **hcmds64srv /start** コマンドを実行して、Automation Director を開始します。

(1)　ポート番号を変更した場合の Hitachi Command Suite のプロパティ更新

Automation Director のポート番号を変更した場合は、次の表に示されている Hitachi Command Suite 共通プロパティを更新する必要があります。

ポート番号（デフォルト）	プロパティファイルのパス（HCS 共通コンポーネントインストール先ディレクトリ）	場所
22015/TCP	¥uCPSB¥httpsd¥conf¥user_httpsd.conf	Listen
		Listen [::]:
		#Listen 127.0.0.1:
22016/TCP	¥uCPSB¥httpsd¥conf¥user_httpsd.conf	VirtualHost タグの*host-name*:*port-number*
		Listen
		Listen [::]:
22031/TCP	¥uCPSB¥httpsd¥conf ¥user_hsso_httpsd.conf	Listen
22032/TCP	¥HDB¥CONF¥emb¥HiRDB.ini	PDNAMEPORT
	¥HDB¥CONF¥pdsys	pd_name_port
	¥database¥work¥def_pdsys	pd_name_port
22033/TCP	¥uCPSB¥CC¥web¥redirector ¥workers.properties	worker.HBase64StgMgmtSSOService.port
	¥uCPSB¥CC¥web¥containers ¥HBase64StgMgmtSSOService¥usrconf ¥usrconf.properties	webserver.connector.ajp13.port
22034/TCP	¥uCPSB¥CC¥web¥containers ¥HBase64StgMgmtSSOService¥usrconf ¥usrconf.properties	webserver.shutdown.port

3.1.4 管理サーバのホスト名または IP アドレスを変更する

ここでは、管理サーバのホスト名または IP アドレスの変更に関して説明します。

(1) 管理サーバのホスト名を変更する

管理サーバのホスト名は、Hitachi Automation Director のインストール後に変更できます。

管理サーバのホスト名は最大 128 文字で、大文字と小文字が区別されます。

操作手順

1. 新しい管理サーバホスト名と IP アドレスをメモしておいてください。
   Windows マシンでホスト名を確認する必要がある場合は、`ipconfig /ALL` コマンドを使用してホスト名を表示します。
2. Automation Director をソースホストにバックアップします。
3. *backup_folder*¥Automation¥base の下のすべてのフォルダを削除します。ただし、*backup_folder*¥Automation¥base の直下のファイルは残します。
4. *backup_folder*¥Automation¥base¥common_conf.txt ファイルを開き、内容を削除して、ファイルを保存します。
5. 管理サーバのホスト名を変更します。次に、サーバを再起動します。
6. `chgcommonbasehostname.bat` *`Revised host name`* を実行して、共通ベースのホスト名構成を変更します。
7. バックアップしたデータを管理サーバにリストアします。
8. Automation Director を停止します。
9. Hitachi Command Suite の共通コンポーネントのプロパティを編集します。
10. 他の Hitachi Command Suite 製品を実行している場合は、必要に応じてそれらの設定を変更します。
11. すべての Hitachi Command Suite サービスが実行していることを確認します。
12. `chgenginehostname.bat` *`Revised host name`* を実行して、オートメーションエンジンのホスト名構成を変更します。
13. 元のホスト名または IP アドレスを使用してブラウザから管理サーバにアクセスする場合は、Hitachi Command Suite の URL を更新します。
14. `hcmds64srv /start` コマンドを実行して Automation Director を起動し、新しい URL を使用して製品にアクセスできることを確認します。

操作結果

管理サーバのホスト名または IP アドレスが変更されました。

(2) 管理サーバのホスト名を変更した場合の Hitachi Command Suite のプロパティ更新

Automation Director 管理サーバのホスト名を変更した場合は、次の表に示されている Hitachi Command Suite 共通プロパティを更新する必要があります。

プロパティファイルのパス（HCS 共通コンポーネントインストール先ディレクトリ）	プロパティ	必要な変更
¥uCPSB¥httpsd¥conf¥user_httpsd.conf	ServerName	値を新しいホスト名に変更します。
	VirtualHost タグ	管理サーバと管理クライアント間の通信に TLS または SSL が使用され、ホ

プロパティファイルのパス（HCS 共通コンポーネントインストール先ディレクトリ）	プロパティ	必要な変更
		スト名が指定された場合、値をアスタリスク（*）に変更します。
	VirtualHost タグの Servername	管理サーバと管理クライアント間の通信に TLS または SSL が使用される場合は、値を新しいホスト名に変更します。
¥HDB¥CONF¥pdsys ¥database¥work¥def_pdsys	pdunit の-x オプション	値をループバックアドレス 127.0.0.1 に変更します。
¥HDB¥CONF¥pdutsys ¥database¥work¥def_pdutsys	pd_hostname	
¥HDB¥CONF¥emb¥HiRDB.ini	PDHOST	

(3) 管理サーバの IP アドレスを変更した場合の Hitachi Command Suite のプロパティ更新

Automation Director 管理サーバの IP アドレスを変更した場合は、次の表に示されている Hitachi Command Suite 共通プロパティを更新する必要があります。

プロパティファイルのパス（HCS 共通コンポーネントインストール先ディレクトリ）	プロパティ	必要な変更
¥uCPSB¥httpsd¥conf ¥user_httpsd.conf	ServerName	値を新しいホスト名または新しい IP アドレスに変更します。
¥HDB¥CONF¥pdsys ¥database¥work¥def_pdsys	pdunit の-x オプション	元の IP 値が指定された場合は、値をループバックアドレス 127.0.0.1 に変更します。
¥HDB¥CONF¥pdutsys ¥database¥work¥def_pdutsys	pd_hostname	
¥HDB¥CONF¥emb¥HiRDB.ini	PDHOST	

3.1.5 Automation Director の URL を変更する

ここでは、管理サーバの URL の変更に関して説明します。

(1) 管理サーバの URL を変更する

管理サーバのホスト名または IP アドレス、Automation Director のポート、または SSL 設定を変更した場合は、Hitachi Automation Director 管理サーバの URL を変更する必要があります。Automation Director が他の Hitachi Command Suite 製品と同じ管理サーバで実行している場合は、Hitachi Command Suite のすべての URL を 1 つのコマンドで変更できます。

メモ プロトコルとポート番号を含んだ完全な URL を使用する必要があります（たとえば、http://HostA:22015）。

操作手順

1. 次のコマンドを使用して、現在の URL を確認します。

   HCS-Common-Component-installation-folder¥bin¥hcmds64chgurl /list
2. Automation Director がスタンドアロンのサーバにインストールされている場合は、次のコマンドで Automation Director の URL だけを変更します。

   HCS-Common-Component-installation-folder¥bin¥hcmds64chgurl /change *new-URL* /type Automation
3. Automation Director が同じサーバにインストールされている場合は、次のコマンドを使用して、この管理サーバ上で実行している Hitachi Command Suite のすべての URL を変更します。

   HCS-Common-Component-installation-folder¥bin¥hcmds64chgurl /change *old-URL new-URL*
4. ショートカットファイルの URL を変更します。

   - Windows Server 2008 R2 の場合：
     ［Start］－［All Programs］－［Hitachi Command Suite］－［Automation Director］を選択して、［HAD］**Login** を右クリックします。［Properties］を選択して、［Web Document］タブで URL を変更します。
   - Windows Server 2012 および Windows Server 2012 R2 の場合：
     ［Start］－［All apps］－［Hitachi Command Suite］－［Automation Director］を選択して、［HAD］**Login** を右クリックします。［Properties］を選択して、［Web Document］タブで URL を変更します。

   URL には次の形式を使用します。
   Protocol://*Management-server-IP-address-or-host-name*:*port-number*/Automation/login.htm
   ここで、

   - *Protocol* は、非 SSL 通信の場合は http、SSL 通信の場合は https です。
   - *Management-server-IP-address-or-host-name* は、Hitachi Automation Director がインストールされている管理サーバの IP アドレスまたはホスト名です。
   - *port-number* は、user_httpsd.conf ファイルの Listen 行で設定されたポート番号です。
     SSL 以外の通信の場合は、SSL 以外の通信用のポート番号を指定します（デフォルト：22015）。
     SSL 通信の場合は、SSL 通信用のポート番号を指定します（デフォルト：22016）。
     user_httpsd.conf ファイルは、*HCS-Common-Component-installation-folder*¥uCPSB¥httpsd¥conf¥ディレクトリにあります。
5. 新しい URL を使用して Automation Director にアクセスできることを確認します。

3.2 セキュア通信を構成する

ここでは、Hitachi Automation Director のセキュア通信を構成する方法について説明します。

3.2.1 Automation Director のセキュリティ設定について

Automation Director に対してセキュア通信を使用することによって、セキュリティを高めることができます。セキュア通信では、Automation Director は Automation Director ネットワーク通信に Secure Sockets Layer（SSL）または Transport Layer Security（TLS）を使用することによって、セキュリティを高めることができます。SSL または TLS により、Automation Director での通信パートナー確認、パートナー識別のための認証強化、送受信される情報内の改ざんデータ検出を実現します。また、通信チャンネルが暗号化されるため、データが盗聴から保護されます。

Automation Director は、以下のタイプの通信について、SSL または TLS を使用したセキュア通信を使用できます。

- 管理サーバと管理クライアント間の通信
- 管理サーバと SMTP サーバ間の通信
- 管理サーバと外部認証サーバ（LDAP ディレクトリサーバ）間の通信
- 管理サーバと管理対象間の通信

また、特定の管理クライアントだけが管理サーバにアクセスできるように、アクセスを制限できます。

メモ　セキュリティを有効にして Automation Director を使用するときには、サーバ証明書の有効期限が切れていないことを確認してください。サーバ証明書の有効期限が切れている場合は、有効な証明書を Automation Director に登録しないとサーバに接続できません。

3.2.2 管理クライアントのセキュリティを構成する

ここでは、管理サーバと管理クライアント間のセキュア通信の設定について説明します。

(1)　管理クライアントのセキュア通信について

SSL を使用して Automation Director 管理サーバと管理クライアント間のセキュア通信を実現します。SSL を実装するには、まず管理サーバに SSL をセットアップし、次に管理クライアントに SSL をセットアップします。Web ベースのインタフェースクライアントに SSL をセットアップするプロセスは、CLI クライアントの場合とは異なります。

メモ

「Allocate Volumes and Create Datastore on VMware vSphere」または「Allocate Like Volumes and Create Datastore on VMware vSphere」サービステンプレートを使用する場合は、Hitachi Command Suite の設定を次のように更新することで、SSL 経由で TLSv1.0 を検証する必要があります。

1. *Hitachi Command Suite installation folder* /Base64/conf/init.conf ファイルの「ssl.protocol」プロパティに「TLSv1」を追加します。
2. Hitachi Command Suite を再起動します。

(2)　セキュアなクライアント通信のためにサーバ上で SSL をセットアップする（Windows）

管理サーバと管理クライアント間のセキュア通信を実装するには、管理サーバで SSL をセットアップする必要があります。

前提条件

サーバで SSL をセットアップする前に、以下の前提条件を確認してください。

- 管理クライアント上で実行している Web ブラウザのバージョンが Automation Director によってサポートされること。
- サーバ証明書の署名アルゴリズムが管理クライアントの Web ブラウザによってサポートされていること。
- 既存の秘密鍵、証明書署名要求、および自己署名証明書の場所が確認されていること（それらを再作成するときの場所を確認してください）。

使用する認証局について、以下の情報を確認してください。

- **hcmds64ssltool** コマンドを使用して作成した証明書署名要求が PEM format であり、秘密鍵のサイズが 2048 ビットであること。
- 認証局によって発行されるサーバ証明書が X.509 PEM format を使用し、署名アルゴリズムをサポートしていること。
- サーバ証明書アプリケーションプロセスが理解されていること。

秘密鍵と証明書署名要求に加えて、以下の手順によって自己署名証明書が作成されます。自己署名証明書は、テスト目的でのみ使用することを推奨します。

操作手順

1. Automation Director を開始します。
2. HCS 共通コンポーネントの秘密鍵（httpsdkey.pem）、証明書署名要求（httpsd.csr）、および自己署名証明書（httpsd.pem）を作成するには、次のコマンドを使用します。

   HCS-Common-Component-installation-folder¥bin¥hcmds64ssltool /key *HCS-Common-Component-installation-folder*¥uCPSB¥httpsd¥sslc¥bin¥demoCA¥httpsdkey.pem /csr *HCS-Common-Component-installation-folder*¥uCPSB¥httpsd¥sslc¥bin¥demoCA¥httpsd.csr /cert *HCS-Common-Component-installation-folder*¥uCPSB¥httpsd¥sslc¥bin¥demoCA¥httpsd.pem /certtext *HCS-Common-Component-installation-folder*¥uCPSB¥httpsd¥sslc¥bin¥demoCA¥httpsd.txt /validity 365

   このコマンドは、自己署名証明書の内容を httpsd.txt に出力します。自己署名証明書は、テスト目的でのみ使用することを推奨します。
   このコマンドを実行すると、署名アルゴリズムは SHA256 と RSA を使用して、validity オプションで指定された有効期限（365 日のタイムスパンに基づく）付きの自己署名証明書を作成します。
   署名アルゴリズムは、sigalg オプションを使用して指定できます。このオプションを省いた場合は、SHA256 と RSA が使用されます。また、SHA1 と RSA または MD5 と RSA を指定することもできます。

   メモ 出力先パスに同じ名前のファイルが存在する場合、hcmds64ssltool コマンドを実行すると、ファイルが上書きされます。ファイルを再作成するときには、ファイルを別の出力先に保存することを推奨します。

3. プロンプトが表示されたら、コロン（:）の後に以下の情報を入力します。
   - サーバ名（管理サーバホスト名） - たとえば、HAD_SC1。
   - 組織単位（セクション） - たとえば、Automation Director。
   - 組織名（会社） - たとえば、Hitachi。
   - 都市または地区名 - たとえば、Santa Clara。
   - 州または県名（フルネーム） - たとえば、California。
   - 国名（2 文字のコード） - たとえば、US。

   フィールドを空白のままにしておくには、ピリオド（.）を入力します。角括弧（[]）内に表示されるデフォルト値を選択するには、［Enter］を押します。
4. 証明書署名要求（httpsd.csr）を認証局に送信して、サーバ証明書を申請します。

   メモ 自己署名証明書を使用する場合、このステップは不要ですが、生産環境では署名付きサーバ証明書を使用することを推奨します。

認証局によって発行されたサーバ証明書は、通常、メールで送信されます。認証局によって送信されたメールとサーバ証明書を必ず保存してください。

5. Automation Director を停止します。
6. 秘密鍵（httpsdkey.pem）とサーバ証明書または自己署名証明書（httpsd.pem）を、次のディレクトリにコピーします。

   HCS-Common-Component-installation-folder¥uCPSB¥httpsd¥conf¥ssl¥server
7. 次の場所から user_httpsd.conf ファイルを開きます。

   HCS-Common-Component-installation-folder¥uCPSB¥httpsd¥conf
   ¥user_httpsd.conf
8. user_httpsd.conf ファイル内で、以下のようにします。
   a. ハッシュ[#]記号を削除することによって、以下の行を非コメント化します。

      #Listen 22016
      #<VirtualHost *:22016>
      から
      #</VirtualHost>
      ただし、#SSLCACertificateFile はコメントアウトしたままにしておく必要があります。
      以下に、user_httpsd.conf ファイルの編集例を示します。SSL ECC を使用している場合は、以下の行も非コメント化します。
      #SSLECCCertificateKeyFile
      #SSLECCCertificateFile

```
ServerName host-name
Listen 22015
Listen [::]:22015
#Listen 127.0.0.1:22015
SSLDisable
Listen 22016
#Listen [::]:22016
<VirtualHost *:22016>
ServerName host-name
SSLEnable
SSLProtocol TLSv1 TLSv11 TLSv12
SSLRequiredCiphers ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-
GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-ECDSA-AES128-
SHA256:ECDHE-ECDSA-AES256-SHA:ECDHE-ECDSA-AES128-SHA:AES256-GCM-
SHA384:AES256-SHA256:AES256-SHA:AES128-SHA256:AES128-SHA:DES-CBC3-
SHA
SSLRequireSSL
SSLCertificateKeyFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/httpsdkey.pem"
SSLCertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/httpsd.pem"
#SSLECCCertificateKeyFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/ecc-httpsdkey.pem"
#SSLECCCertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/ecc-httpsd.pem"
# SSLCACertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
cacert/anycert.pem"
</VirtualHost>
#HWSLogSSLVerbose On
```

   b. 必要に応じて、以下の行を編集します。

      最初の行の ServerName
      <VirtualHost>タグの ServerName

SSLCertificateKeyFile
SSLCertificateFile
SSLECCCertificateKeyFile（ECC を使用する場合）
SSLECCCertificateFile（ECC を使用する場合）
#SSLCACertificateFile
認証局から発行されたチェーンサーバ証明書を使用するときには、"#SSLCACertificateFile"行から番号記号（#）を削除し、（認証局によって作成された）チェーン証明書ファイルを絶対パスで指定します。

メモ
外部サーバから管理サーバへの非 SSL 通信をブロックするには、Listen 22015 行と Listen [::]:22015 行の先頭に番号記号（#）を追加してコメントアウトします。これらの行をコメントアウトした後、#Listen 127.0.0.1:22015 行の番号記号を削除します。
IPv6 環境の場合、#Listen [::]:22016 行の先頭の番号記号（#）を削除します。
管理サーバ内の非 SSL 通信をブロックするには、HBase 64 Storage Mgmt Web Service のポートを閉じます。

以下に、user_httpsd.conf ファイルの編集例を示します。番号は、デフォルトのポート番号を示しています。

```
ServerName host-name
Listen 22015
Listen [::]:22015
#Listen 127.0.0.1:22015
SSLDisable
Listen 22016
#Listen [::]:22016
<VirtualHost *:22016>
ServerName host-name
SSLEnable
SSLProtocol TLSv1 TLSv11 TLSv12
SSLRequiredCiphers ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-
GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-
ECDSA-AES256-SHA:ECDHE-ECDSA-AES128-SHA:AES256-GCM-SHA384:AES256-
SHA256:AES256-SHA:AES128-SHA256:AES128-SHA:DES-CBC3-SHA
SSLRequireSSL
SSLCertificateKeyFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/httpsdkey.pem"
SSLCertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/server-certificate-or-self-signed-certificate-file"
#SSLECCCertificateKeyFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/ecc-httpsdkey.pem"
#SSLECCCertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/ecc-httpsd.pem"
SSLCACertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
cacert/certificate-file-from-certificate-authority"
</VirtualHost>
#HWSLogSSLVerbose On
```

9. HCS 共通コンポーネントのサーバ証明書を Automation Director が参照するトラストストアにインポートします。
   証明書をトラストストアにインポートするには hcmds64keytool ユーティリティを使用します。
   HCS-Common-Component-installation-folder¥bin¥hcmds64keytool -import -alias *alias-name* -keystore *HCS-Common-Component-installation-folder*¥uCPSB¥jdk¥jre¥lib¥security¥jssecacerts -storepass *trust-store-password* -file *certificate-file*

10. Automation Director を開始します。
11. 次のように hcmds64chgurl を使用して、Automation Director の URL を更新します。
    - プロトコルを http:から https:に変更します。
    - セキュア通信に使用されるポート番号を変更します。

操作結果

これで、Automation Director サーバ上で SSL が実装されます。

(3) セキュアなクライアント通信のためにサーバ上で SSL をセットアップする（Linux）

管理サーバと管理クライアント間のセキュア通信を実装するには、管理サーバで SSL をセットアップする必要があります。

前提条件

サーバで SSL をセットアップする前に、以下の前提条件を確認してください。

- 管理クライアント上で実行している Web ブラウザのバージョンが Automation Director によってサポートされること。
- サーバ証明書の署名アルゴリズムが管理クライアントの Web ブラウザによってサポートされていること。
- 既存の秘密鍵、証明書署名要求、および自己署名証明書の場所が確認されていること（それらを再作成するときの場所を確認してください）。

使用する認証局について、以下の情報を確認してください。

- **hcmds64ssltool** コマンドを使用して作成した証明書署名要求が PEM format であり、秘密鍵のサイズが 2048 ビットであること。
- 認証局によって発行されるサーバ証明書が X.509 PEM format を使用し、署名アルゴリズムをサポートしていること。
- サーバ証明書アプリケーションプロセスが理解されていること。

秘密鍵と証明書署名要求に加えて、以下の手順によって自己署名証明書が作成されます。自己署名証明書は、テスト目的でのみ使用することを推奨します。

操作手順

1. Automation Director を開始します。
2. HCS 共通コンポーネントの秘密鍵（httpsdkey.pem）、証明書署名要求（httpsd.csr）、および自己署名証明書（httpsd.pem）を作成するには、次のコマンドを使用します。

   HCS-Common-Component-installation-folder/bin/hcmds64ssltool -key *HCS-Common-Component-installation-folder*/uCPSB/httpsd/sslc/bin/demoCA/httpsdkey.pem -csr *HCS-Common-Component-installation-folder*/uCPSB/httpsd/sslc/bin/demoCA/httpsd.csr -cert *HCS-Common-Component-installation-folder*/uCPSB/httpsd/sslc/bin/demoCA/httpsd.pem -certtext*HCS-Common-Component-installation-folder*/uCPSB/httpsd/sslc/bin/demoCA/httpsd.txt -validity 365

   このコマンドは、自己署名証明書の内容を httpsd.txt に出力します。自己署名証明書は、テスト目的でのみ使用することを推奨します。

このコマンドを実行すると、署名アルゴリズムは SHA256 と RSA を使用して、validity オプションで指定された有効期限（365 日のタイムスパンに基づく）付きの自己署名証明書を作成します。
署名アルゴリズムは、sigalg オプションを使用して指定できます。このオプションを省いた場合は、SHA256 と RSA が使用されます。また、SHA1 と RSA または MD5 と RSA を指定することもできます。

メモ 出力先パスに同じ名前のファイルが存在する場合、hcmds64ssltool コマンドを実行すると、ファイルが上書きされます。ファイルを再作成するときには、ファイルを別の出力先に保存することを推奨します。

3. プロンプトが表示されたら、コロン（:）の後に以下の情報を入力します。
   - サーバ名（管理サーバホスト名） - たとえば、HAD_SC1。
   - 組織単位（セクション） - たとえば、Automation Director。
   - 組織名（会社） - たとえば、Hitachi。
   - 都市または地区名 - たとえば、Santa Clara。
   - 州または県名（フルネーム） - たとえば、California。
   - 国名（2 文字のコード） - たとえば、US。

   フィールドを空白のままにしておくには、ピリオド（.）を入力します。角括弧（[]）内に表示されるデフォルト値を選択するには、［Enter］を押します。
4. 証明書署名要求（httpsd.csr）を認証局に送信して、サーバ証明書を申請します。

メモ 自己署名証明書を使用する場合、このステップは不要ですが、生産環境では署名付きサーバ証明書を使用することを推奨します。

   認証局によって発行されたサーバ証明書は、通常、メールで送信されます。認証局によって送信されたメールとサーバ証明書を必ず保存してください。
5. Automation Director を停止します。
6. 秘密鍵（httpsdkey.pem）とサーバ証明書または自己署名証明書（httpsd.pem）を、次のディレクトリにコピーします。

   HCS-Common-Component-installation-folder/uCPSB/httpsd/conf/ssl/server
7. 次の場所から user_httpsd.conf ファイルを開きます。

   HCS-Common-Component-installation-folder/uCPSB/httpsd/conf/user_httpsd.conf
8. user_httpsd.conf ファイル内で、以下のようにします。
   a. ハッシュ[#]記号を削除することによって、以下の行を非コメント化します。

      #Listen 22016
      #<VirtualHost *:22016>
      から
      #</VirtualHost>
      ただし、#SSLCACertificateFile はコメントアウトしたままにしておく必要があります。
      以下に、user_httpsd.conf ファイルの編集例を示します。SSL ECC を使用している場合は、以下の行も非コメント化します。
      #SSLECCCertificateKeyFile

#SSLECCCertificateFile

```
ServerName host-name
Listen 22015
Listen [::]:22015
#Listen 127.0.0.1:22015
SSLDisable
Listen 22016
#Listen [::]:22016
<VirtualHost *:22016>
ServerName host-name
SSLEnable
SSLProtocol TLSv1 TLSv11 TLSv12
SSLRequiredCiphers ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-
GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-ECDSA-AES128-
SHA256:ECDHE-ECDSA-AES256-SHA:ECDHE-ECDSA-AES128-SHA:AES256-GCM-
SHA384:AES256-SHA256:AES256-SHA:AES128-SHA256:AES128-SHA:DES-CBC3-
SHA
SSLRequireSSL
SSLCertificateKeyFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/httpsdkey.pem"
SSLCertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/httpsd.pem"
#SSLECCCertificateKeyFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/ecc-httpsdkey.pem"
#SSLECCCertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/ecc-httpsd.pem"
# SSLCACertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
cacert/anycert.pem"
</VirtualHost>
#HWSLogSSLVerbose On
```

b. 必要に応じて、以下の行を編集します。

最初の行の ServerName

<VirtualHost>タグの ServerName

SSLCertificateKeyFile

SSLCertificateFile

SSLECCCertificateKeyFile（ECC を使用する場合）

SSLECCCertificateFile（ECC を使用する場合）

#SSLCACertificateFile

認証局から発行されたチェーンサーバ証明書を使用するときには、"#SSLCACertificateFile"行から番号記号（#）を削除し、（認証局によって作成された）チェーン証明書ファイルを絶対パスで指定します。

メモ

外部サーバから管理サーバへの非 SSL 通信をブロックするには、Listen 22015 行と Listen [::]:22015 行の先頭に番号記号（#）を追加してコメントアウトします。これらの行をコメントアウトした後、#Listen 127.0.0.1:22015 行の番号記号を削除します。

IPv6 環境の場合、#Listen [::]:22016 行の先頭の番号記号（#）を削除します。

管理サーバ内の非 SSL 通信をブロックするには、HBase 64 Storage Mgmt Web Service のポートを閉じます。

以下に、user_httpsd.conf ファイルの編集例を示します。番号は、デフォルトのポート番号を示しています。

```
ServerName host-name
```

```
Listen 22015
Listen [::]:22015
#Listen 127.0.0.1:22015
SSLDisable
Listen 22016
#Listen [::]:22016
<VirtualHost *:22016>
ServerName host-name
SSLEnable
SSLProtocol TLSv1 TLSv11 TLSv12
SSLRequiredCiphers ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-
GCM-SHA256:ECDHE-ECDSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-
ECDSA-AES256-SHA:ECDHE-ECDSA-AES128-SHA:AES256-GCM-SHA384:AES256-
SHA256:AES256-SHA:AES128-SHA256:AES128-SHA:DES-CBC3-SHA
SSLRequireSSL
SSLCertificateKeyFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/httpsdkey.pem"
SSLCertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/server-certificate-or-self-signed-certificate-file"
#SSLECCCertificateKeyFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/ecc-httpsdkey.pem"
#SSLECCCertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
server/ecc-httpsd.pem"
SSLCACertificateFile
"HCS-Common-Component-installation-directory/uCPSB/httpsd/conf/ssl/
cacert/certificate-file-from-certificate-authority"
</VirtualHost>
#HWSLogSSLVerbose On
```

9. HCS 共通コンポーネントのサーバ証明書を Automation Director が参照するトラストストアにインポートします。

   証明書をトラストストアにインポートするには keytool ユーティリティを使用します。

   HCS-Common-Component-installation-directory/uCPSB/jre/jdk/bin/keytool -import -alias *alias-name* -keystore *HCS-Common-Component-installation-directory*/uCPSB/jdk/jre/lib/security/jssecacerts -storepass *trust-store-password* -file *certificate-file*

10. Automation Director を開始します。
11. 次のように hcmds64chgurl を使用して、Automation Director の URL を更新します。

    - プロトコルを http:から https:に変更します。
    - セキュア通信に使用されるポート番号を変更します。

操作結果

これで、Automation Director サーバ上で SSL が実装されます。

(4) Web ベースの管理クライアントで SSL をセットアップする

管理サーバと管理クライアント間のセキュア通信を実装するには、Automation Director の Web ベースのユーザーインタフェースにアクセスするすべての Automation Director 管理クライアント上で SSL をセットアップする必要があります。まず、管理サーバに SSL をセットアップし、次に管理クライアントに SSL をセットアップします。このクライアントから管理サーバに最初にアクセスするときのみ、この手順に従う必要があります。

前提条件

使用される署名アルゴリズムが SHA256 と RSA の場合、使用される Web ブラウザは SHA256 と RSA 署名を持つサーバ証明書をサポートする必要があります。

操作手順

1. 管理 Web クライアントから、次の URL を使用して、SSL 接続で管理サーバにアクセスします。
   `https://HAD-management-server-name:port-number-for-SSL-communication/Automation/`
2. SSL 証明書をインストールします。

操作結果

SSL 証明書が管理クライアントに登録され、SSL を使用して管理サーバと通信できるようになります。

3.2.3 外部認証サーバのセキュア通信を設定する

Windows 環境で Automation Director 管理サーバと LDAP ディレクトリサーバ間のセキュア通信を実装するには、StartTLS プロトコルを使用します。StartTLS を実装するには、`exauth.properties` ファイルでプロパティを更新し、LDAP ディレクトリサーバ証明書を管理サーバにインポートする必要があります。

メモ Linux 環境で IPV6 アドレスを指定する場合は、アドレスを角括弧[ ]で囲む必要があります。

3.2.4 プライマリ HCS サーバへの認証接続のポート番号を変更する（Windows）

ポート番号を変更するには：

hcmds64prmset コマンドを実行して、認証接続のポート番号を次のように変更します。

```
HCS-Common-Component-installation-folder¥bin¥hcmds64prmset /hostname
<the hostname of a primary server> /sslport <SSL port number>
```

方法：

- 「hostname」として、クレデンシャルの共通名（CN）と同じ名前を指定します。
- 共通コンポーネントの SSL ポート番号（`sslport`）を指定します。デフォルトは 22016 です。

3.2.5 プライマリ HCS サーバへの認証接続のポート番号を変更する（Linux）

ポート番号を変更するには：

hcmds64prmset コマンドを実行して、認証接続のポート番号を次のように変更します。

```
HCS-Common-Component-installation-folder/bin/hcmds64prmset -hostname
<the host name of a primary server> -sslport <SSL port number>
```

方法：

- 「hostname」として、クレデンシャルの共通名（CN）と同じ名前を指定します。
- 共通コンポーネントの SSL ポート番号（`sslport`）を指定します。デフォルトは 22016 です。

3.2.6 VMware vCenter 証明書をインポートする

VMware サービスまたは VMware vCenter サーバ証明書を使用するときには、証明書を Hitachi Command Suite 共通コンポーネントのトラストストアにインポートする必要があります。

以下の証明書もインポートする必要があります。

- 認証局
- 中間認証局
- ルート認証局

場合によっては、有名な認証局証明書が既にインポートされている可能性があります。この場合、この手順は不要です。

Windows の場合、**hcmds64keytool** コマンドを使用します。Unix の場合、標準 **keytool** を使用します。Java で証明書をインポートするには、トラストストアのパスワードが 6 文字以上であることを確認してください。また、新しい別名が既存の別名と衝突しないことを確認してください。

Windows の場合：

```
HCS-Common-Component-installation-folder¥bin¥hcmds64keytool -import -
alias <alias name> -keystore <Hitachi Command Suite The installation
folder>¥uCPSB
¥jdk¥jre¥lib¥security¥jssecacerts -storepass <trust store password> -file
<certificate file>
```

Unix の場合：

```
HCS-Common-Component-installation-folder/uCPSB/jre/jdk/bin/keytool  -
import -alias <alias name> -keystore HCS-Common-Component-installation-
folder/uCPSB/jdk/jre/lib/security/jssecacerts -storepass <trust store
password> -file <certificate file>
```

追加のガイドライン

- VMware vCenter のセキュリティ設定の方法については、VMware のマニュアルを参照してください。
- vCenter サーバ証明書を取得するには、VMware のマニュアルでサーバ証明書へのアクセスについて参照してください。

3.2.7 Device Manager Agent のトラストストアにサーバ証明書をインポートする

Clone（Shadow Image）、Snapshot（Thin Image）、および Copy Topology サービスを使用するときには、Hitachi Device Manager サーバ証明書を Device Manager Agent のトラストストアにインポートする必要があります。

詳細については、『*Hitachi Command Suite システム構成ガイド*』の「5.5.22 Device Manager エージェントのトラストストアーへのサーバ証明 書のインポート」を参照してください。

3.2.8 Device Manager サーバ証明書をインポートする

Add Host 機能が有効になっている場合、各 Device Manager のサーバ証明書を取得し、自己署名サーバ証明書または認証局証明書を Automation Director が参照するトラストストアにインポートする必要があります。

Device Manager サーバ証明書を取得する

作成するには、『Hitachi Command Suite システム構成ガイド』の「5.3 SSL サーバの構築（Device Manager サーバ）」を参照してください。

メモ　Allocate Volumes for Symmetric Cluster Server from 2-Storage Systems サービスを使用する場合、Add Host 機能の有効／無効状態にかかわらず、これが必要です。

自己署名サーバ証明書または認証局証明書をインポートする

認証局証明書を使用するときには、中間認証局およびルート認証局の証明書もインポートする必要があります。場合によっては、有名な認証局証明書が既にインポートされている可能性があります。この場合、この手順は不要です。

メモ　Automation Director サーバ上の HCS 共通コンポーネントトラストストアは、`jssecacerts` です。

以下のガイドラインに従ってください。

- 複数の Device Manager 構成を実行している場合は、各 Device Manager のサーバ証明書を取得する必要があります。
- 自己署名証明書を使用するときには、各 Device Manager サーバの自己署名証明書をトラストストアにインポートします。
- 認証局証明書を使用するときには、サーバ証明書を発行する各認証局の証明書をトラストストアにインポートします。

関連参照

- 3.2.9 Hitachi Command Suite 共通コンポーネントのトラストストアに各 Device Manager のサーバ証明書をインポートする

3.2.9 Hitachi Command Suite 共通コンポーネントのトラストストアに各 Device Manager のサーバ証明書をインポートする

サーバ証明書は、各 Device Manager サーバから入手した後、Automation Director が参照するトラストストアにインポートする必要があります。

1. Device Manager トラストストアファイルをダウンロードします。
   Device Manager が自己署名証明書を使用している場合は、次のいずれかの URL を使用して、Web ブラウザからトラストストアをダウンロードします。Device Manager が認証局の証明書を既に使用している場合、このステップは不要です。
   SSL の場合はポート番号を 2443 に、非 SSL の場合は 2001 に、デフォルトで設定します。
   詳細については、『Hitachi Command Suite システム構成ガイド』の「5.5 SSL クライアントの構築」を参照してください。
   `https://<IP address or host name of Device Manager server>:<SSL port number of Device Manager server>/service/HiCommandCerts`
   `https://<IP address or host name of Device Manager server>:<Non-SSL port number of Device Manager server>/service/HiCommandCerts`
2. 各 Device Manager の証明書をエクスポートします。

Device Manager が自己署名証明書を使用している場合は、**hcmds64keytool** を使用して、Device Manager サーバ証明書を、ダウンロードしたトラストストアからエクスポートします。Device Manager が認証局の証明書を既に使用している場合、このステップは不要です。
ダウンロードしたトラストストアをトラストストアファイルとして指定します。
詳細については、『Hitachi Command Suite システム構成ガイド』の「5.5 SSL クライアントの構築」を参照してください。
Windows の場合：

```
HCS-installation-folder¥Base64¥bin¥hcmds64keytool -export keystore
<trust-store-file> -alias <alias-name> -file <certificate-file>
```

Linux の場合：

```
HCS-installation-folder/Base64/uCPSB/jdk/bin/keytool -export keystore
<trust-store-file> -alias <alias-name> -file <certificate-file>
```

3. 各 Device Manager の証明書を Hitachi Command Suite 共通コンポーネントのトラストストアにインポートします。
自己署名証明書のエクスポートしたサーバ証明書、またはトラストストアにある認証局の証明書をインポートします。
Windows の場合、**hcmds64keytool** を使用します。Unix の場合、Java の標準 **keytool** を使用して、証明書をインポートします。Java で証明書をインポートするには、トラストストアのパスワードが 6 文字以上であることを確認してください。また、新しい別名が既存の別名と衝突しないことを確認してください。
Device Manager が認証局の証明書、中間認証局および（他の認証局もルートする）ルート認証局の証明書を使用している場合、認証局をインポートする必要があります。場合によっては、有名な認証局証明書が既にインポートされている可能性があります。この場合、この手順は不要です。
Windows の場合：

```
HCS-Common-Component-installation-Folder¥bin¥hcmds64keytool -import -
alias <alias-name> -keystore HCS-Common-Component-installation-
directory¥uCPSB¥jdk¥jre¥lib¥security¥jssecacerts -storepass <trust-
store-password> -file <certificate-file>
```

Unix の場合：

```
HCS-Common-Component-installation-Folder/jdk/bin/hcmds64keytool -
import -alias <alias-name> -keystore HCS-Common-Component-
installation-directory/uCPSB/jdk/jre/lib/security/jssecacerts -
storepass <trust-store-password> -file <certificate-file>
```

3.2.10 サーバ証明書の有効期限を確認する

SSL 証明書の有効期限をチェックすることで、証明書の有効期限が切れていないかどうかを確認できます。管理サーバ証明書の有効期限が切れておらず、管理対象サーバとのセキュア通信を維持できることを確認する必要があります。

Hitachi Command Suite 共通コンポーネントのサーバ証明書の有効期限をチェックするには、次のコマンドを実行します。

Windows の場合：

```
<Hitachi Command Suite Installation
Folder>¥Base64¥uCPSB¥jdk¥jre¥bin¥keytool -printcert -v -file
<File name of certificate >
```

Linux の場合：

```
<Hitachi Command Suite Installation
Directory>/Base64/uCPSB/jdk/bin/keytool -printcert -v -file
<File name of certificate>
```

メモ　自己署名サーバ証明書の有効期限は、サーバ間の接続時にはチェックされません。HAD サーバと Device Manager サーバの接続時に証明書の有効期限をチェックする必要がある場合は、認証局によって発行された証明書を使用してください。その場合、サーバの証明書だけでなく、認証局と中間認証局の証明書もインポートします。その後、認証局を HCS 共通コンポーネントトラストストアにルートします。

3.3 Replication Manager の RMI 通信を有効にする

このセクションでは、管理サーバ上の Replication Manager の RMI 通信を有効にします。

前提条件

Administrator 権限を持つユーザー（Windows の場合）または root ユーザー（Linux の場合）として、Device Manager サーバにログインします。

Replication Manager の RMI 通信を有効にするには：

操作手順

1. Hitachi Command Suite 製品のサービスを停止します。
2. Replication Manager の `base.properties` ファイルの `base.rmi.enabled` プロパティとして、`true` を指定します。`base.properties` ファイルは、次の場所に格納されています。
   Windows：
   `Installation-folder-for-Hitachi-Command-Suite``¥ReplicationManager¥conf`
   Linux：
   `installation-directory-for-Hitachi-Command-Suite/``ReplicationManager/conf`
   Replication Manager の `base.properties` ファイルと `base.rmi.enabled` プロパティの詳細については、『*Hitachi Command Suite Replication Manager システム構成ガイド*』を参照してください。
3. Device Manager サーバの `rpmlib.properties` ファイルの `rpmlib.rpm.port` プロパティを設定します。
   Replication Manager の `base.properties` ファイルの `base.rmi.port` プロパティに対して設定されているポート番号を入力します。`base.rmi.port` プロパティの値（デフォルト：25200）を変更していない場合は、この操作は不要です。
   `base.properties` ファイルは、次の場所に格納されています。
   Windows：
   `Installation-folder-for-Hitachi-Command-Suite``¥ReplicationManager¥conf`
   Linux：
   `installation-directory-for-Hitachi-Command-Suite/``ReplicationManager/conf`
   Replication Manager の `base.properties` ファイルと base.rmi.enabled プロパティの詳細については、『*Hitachi Command Suite Replication Manager システム構成ガイド*』を参照してください。
4. Hitachi Command Suite 製品のサービスを開始します。

3.4 別のホストへ Hitachi Automation Director インストールを移動する

必要に応じて、Hitachi Automation Director のインストールを別のホストに再配置できます。

メモ　再配置元のホスト名または IP アドレスと再配置先のホスト名または IP アドレスが異なる場合は、管理サーバのホスト名を変更する必要があります。

前提条件

以下の設定が再配置元のホストと再配置先のホストで同じであることを確認します。

- ホスト名と IP アドレス。
- 文字コードタイプ。
- Hitachi Automation Director によって使用されるオペレーティングシステムユーザーのアカウント。
- HCS 製品環境（構成、バージョン、およびリビジョン）。
- Automation Director のインストールパス。

Hitachi Automation Director の***タスク***タブの「状態」列が「実行中」、「応答待ち中」、「異常検出」、「長期実行中」、または「停止中」を示す処理中のタスクがないことも確認する必要があります。

操作手順

1. Administrator 権限を使用して管理サーバにログインします。
2. Automation Director をソースホストにバックアップします。
   a. **`hcmds64srv /stop`** コマンドを実行して、現在のサービスを停止します。
   b. **`backupsystem`** コマンドを実行して、バックアップを実行します。
3. アーカイブされたバックアップファイルを再配置先のホストに移動します。
4. 再配置先のホストの管理サーバにログオンします。
5. 再配置先のホストで、Hitachi Automation Director のリストアを実行します。
   a. **`hcmds64srv /stop`** コマンドを実行して、サービスを停止します。
   b. **`restoresystem`** コマンドを実行して、バックアップをリストアします。
   c. リストア先の環境に合わせて、以下の構成ファイルの適切な設定を変更します。
      - 外部認証サーバ統合構成ファイル（`exauth.properties`）
      - セキュリティ定義ファイル（security.conf）
      - 監査ログ定義ファイル（auditlog.conf）
      - ポート番号変更設定（user_httpsd.conf）
      - SSL 環境構築手順（user_httpsd.conf）

   これらの構成ファイルは、次のディレクトリにあります。
   - *`Backup destination folder ¥HBase¥base¥conf`*
   - *`Backup destination folder ¥HBase¥base¥httpsd.conf`*
6. ポート番号が変更された場合、新しいポート番号を反映するように、必要な設定を変更します。
7. **`hcmds64srv /start`** コマンドを実行して、サービスを再開します。

関連タスク

- (1) 管理サーバのホスト名を変更する

3.5 外部ネットワーク構成のない Automation Director を実行する

プライベートな内部ネットワークを使用する環境で Automation Director を実行するには、Authenticode シグネチャ機能を無効にする必要があります。Authenticode は、ネットワーク経由でダウンロードまたは転送されるソフトウェアの整合性を確認します。Windows は、デフォルトでは Authenticode シグネチャ機能が有効な状態で実行します。

外部ネットワークに接続しない環境では、システムがサービス部品を実行できるようになるまでに 20 秒以上かかることがあります。

Authenticode シグネチャを無効にするには、Microsoft.NET framework を再構成する必要があります。

操作手順

1. テキストエディタを使用して、以下のファイルを開きます。
   a. **<system-drive>:¥Windows¥Microsoft.NET¥Framework ¥v2.0.50727¥aspnet.config**
   b. **<system-drive>:¥Windows¥Microsoft.NET¥Framework¥v2.0.50727¥CONFIG ¥machine.config**
2. 以下に示されているように、**'generatePublisherEvidence enabled'** を false に設定して、ファイルを保存します。

```
-----------------------------------------
<configuration>
<runtime>
<generatePublisherEvidence enabled="false"/>
</runtime>
</configuration>
-----------------------------------------
```

3.6 プロパティファイル（config_user.properties）でシステム構成を変更する

conf_user.properties ファイルは、ログやタスクなど、Hitachi Automation Director のさまざまな設定を構成するための定義ファイルです。プロパティファイルを変更したときには、Hitachi Automation Director エンジン Web サービスを再起動してください。

このファイルから以下の構成プロパティを変更できます。

- ログファイル構成（保存するログの数を指定します）
- タスクおよび履歴構成（保存するタスクとタスク履歴の数を指定します）
- リモートコマンド実行に関する構成（SSH/telnet ポート番号）
- メール通知の構成情報
- Service Builder に関する構成情報
- 接続タイムアウト値の設定

形式

specification-key-name=setting

インストール先フォルダ

HAD-installation-folder¥conf

説明

プロパティファイルを編集するときには、次のことに注意してください。

- #で始まる行は、コメントとして扱われます。
- 空白行は無視されます。
- エンコードは ISO 8859-1 です。
- 内容は大文字と小文字が区別されます。
- 文字列の中で¥を指定するには、¥¥と入力する必要があります。
- 設定として無効な値を入力した場合はデフォルト値に設定され、メッセージ **KNAE02022-W** が統合トレースログとパブリックログに出力されます。
- 1 つのファイル内で同じ指定キーが複数回入力された場合、最後に指定したキーが有効になります。

プロパティファイルでの設定

分類	キー名	設定	値	デフォルト値
ログ [1]	logger.syslogle vel	イベントログまたは syslog 出力の閾値を指定します。	• 0 : メッセージ ID の出力レベルが 0 の場合のみ出力します。 • 10 : メッセージ ID の出力レベルが 0 または 10 の場合に出力します。	0
	logger.message. server.MaxBacku pIndex	サーバのログバックアップファイルの最大数を指定します。	1～16	7
	logger.message. server.MaxFileS ize	サーバの最大ログファイルサイズ（KB 単位）を指定します。	4～2097151	1024
	logger.message. command.MaxBack upIndex	コマンドのログバックアップファイルの最大数を指定します。	1～16	7

分類	キー名	設定	値	デフォルト値
	logger.message.command.MaxFileSize	コマンドの最大ログファイルサイズ（KB 単位）を指定します。	4～2097151	1024
	logger.TA.MaxFileSize	タスクの最大ログファイルサイズ（KB 単位）を指定します。	4～2097151	10240
タスク管理	tasklist.autoarchive.taskRemainingPeriod	終了したタスクをタスクリストに残しておく期間（日数）を指定します。	1～90	7
	tasklist.autoarchive.executeTime	自動アーカイブタスクを実行する時刻を指定します。	00:00:00～23:59:59	04:00:00
	tasklist.autoarchive.maxTasks	タスクリストに表示するタスクの最大数を指定します。	100～5000	5000
	tasklist.autodelete.maxHistories	保持する履歴エントリの最大数を指定します。	100～30000	30000
	task.details.jobnet.status.visible	[タスク詳細] ダイアログボックスのステップリストにタスク処理エンジンのステータスを表示するか、またはステップのステータスを表示するかを指定します。 true : タスク処理エンジンのステータスを表示します false : ステップのステータスを表示します。	true/false	false
サービス管理	packagemanager.extraPresets.maxFiles	追加の事前設定フォルダに配置できるサービステンプレート 1 つあたりの事前設定プロパティ定義ファイルの最大数を指定します。	5～100	5
反復	foreach.max_value	繰り返し実行部品によって実行できる同時タスクの最大数を指定します。	1～99	3
リモート接続ポート番号	ssh.port.number	操作対象機器の SSH ポート番号を指定します。	0～65535	22
	telnet.port.number	操作対象機器の Telnet ポート番号を指定します。	0～65535	
一般コマンド、リモートコマンド、ファイル転送、端末接続	plugin.stdoutSize.wmi	標準出力および標準エラーの合計サイズがプロパティ値を超えると、部品エラーが発生します。 注：プロパティ値の単位はキロバイト（KB）です。		

分類	キー名	設定	値	デフォルト値
		次の条件が当てはまる場合、部品操作時にこのプロパティが適用されます。 - 接続先のホストが Windows - 実行対象の部品が汎用コマンド実行部品またはコンテンツ部品 Windows では、改行数が 65535 以上でも、部品は実行を続けることができます。この機能の特徴を生かすには、プロパティ値を適切に設定する必要があります。たとえば、このプロパティが 100 KB に設定（デフォルト値）されている場合は、部品は改行の最大数 65535 以上を処理できません。部品は、最大 100 KB に達すると実行を停止します。		
	plugin.stdoutSize.ssh	標準出力および標準エラーの合計サイズがプロパティ値を超えると、部品エラーが発生します。 注：プロパティ値の単位はキロバイト（KB）です。 次の 2 つの主要な条件が当てはまる場合、部品操作時にこのプロパティが適用されます。 [条件（1）（注：次の対象の条件を満たす必要があります。）] - 接続先のホストが Linux または UNIX。 - 実行対象の部品が汎用コマンド実行部品またはコンテンツ部品。 [条件（2）（注：次のプロトコル条件と部品の条件を満たす必要があります。）] - 接続プロトコルが SSH。 - 実行対象の部品がターミナル接続部品またはターミナルコマンド実行部品。		
	plugin.stdoutSize.telnet	標準出力および標準エラーの合計サイズがプロパティ値を超えると、部品エラーが発生します。 注：プロパティ値の単位はキロバイト（KB）です。 次の条件が当てはまる場合、部品操作時にこのプロパティが適用されます。		

分類	キー名	設定	値	デフォルト値
		- 接続プロトコルが SSH。 - 対象の部品がターミナル接続部品またはターミナルコマンド実行部品。		
	plugin.remoteFileAccess.retry.times	コンテンツ部品またはファイル転送部品によって内部実行されるファイル操作コマンドの再試行回数を指定します。再試行間隔は 100ms に固定されています。 一時的なファイルアクセスエラーが発生した場合、コマンドを再試行すると操作が成功することがあります。ただし、ファイルアクセスエラーが回復しなかった場合、部品が終了するまで、再試行に余分な時間がかかります。ディスクに問題がない場合でもファイルアクセスエラーが発生する環境では、このプロパティを指定してください。	0～100	0
	ssh.privateKeyFile	SSH 接続に公開鍵認証が使用される場合、秘密鍵ファイルの絶対パスを指定します。	0～255	
	plugin.localMode	ローカル実行モードを有効にするか無効にするかを指定します。 true：有効 false：無効	true/false	true
リモートファイル操作の再試行	plugin.remoteFileAccess.retry.times	コンテンツ部品およびファイル転送部品によって内部実行されるファイルを操作するコマンドの再試行回数を指定します。再試行間隔は 100ms に固定されています。 一時的なファイルアクセスエラーが発生した場合でも、再試行によって成功することがあります。ただし、ファイルアクセスエラーが回復しなかった場合、部品が終了するまで、再試行に余分な時間がかかります。ディスクなどに問題がない場合でもファイルアクセスエラーが発生する環境では、このプロパティを設定してください。	0～100	0
端末接続	plugin.terminal.prompt.account	ユーザー ID 待機状態の検出に使用される正規表現を指定します。 標準出力および標準エラー出力が指定された正規表現に一	1～1024	login|Login Name|Username|UserName

分類	キー名	設定	値	デフォルト値
		致した場合、ターミナル接続部品（プロトコルとして Telnet が指定される）は、ユーザー ID が入力されなければならないと判断して、ユーザー ID を入力します。		
	plugin.terminal.prompt.password	パスワード待機状態の検出に使用される正規表現を指定します。 標準出力および標準エラー出力が指定された正規表現に一致した場合、ターミナル接続部品（プロトコルとして Telnet が指定される）は、パスワードが入力されなければならないと判断して、パスワードを入力します。	1～1024	password| Password| PassWord
	telnet.connect.wait	操作対象機器との SSH 接続が確立された後、標準出力が戻るまでの待ち時間（秒数）を指定します。	1～600	60
リモートコマンド	plugin.remoteCommand.executionDirectory.wmi	対象ホストが Windows を実行している場合に実行するコンテンツ部品を含む、実行ディレクトリのパスを指定します。実行ディレクトリは、事前に作成しておく必要があります。 コンテンツ部品の［実行モード］が［スクリプト］の場合、指定された値とスクリプトファイル名の合計文字列長は最大 140 文字です。長さが 140 文字を超えた場合、スクリプトの転送は失敗します。さらに、スクリプトファイル名は 90 文字以内で指定しなければならないため、この指定値は 50 文字以内でなければなりません。	0～256	
	plugin.remoteCommand.executionDirectory.ssh	操作対象ホストの OS が UNIX の場合にコンテンツ部品を実行する実行ディレクトリのパスを指定します。実行ディレクトリは、事前に作成しておく必要があります。	0～128	
	plugin.remoteCommand.workDirectory.ssh	操作対象ホストの OS が UNIX の場合、ファイル転送部品またはコンテンツ部品の実行時に使用される作業フォルダを指定します。フォルダ	1～128	/tmp/Hitachi_AO

分類	キー名	設定	値	デフォルト値
		またはシンボリックリンクを絶対パスとして入力します(1～128 文字)。さらに、シンボリックリンクはパスのレイヤとして含めることができます。		
リモートホスト接続の再試行	ssh.connect.retry.times	操作対象機器への SSH 接続が失敗した場合の再試行回数を指定します。	0～100	3
	ssh.connect.retry.interval	操作対象機器への SSH 接続が失敗した場合の再試行間隔（秒数）を指定します。	1～600	10
	wmi.connect.retry.times	操作対象機器への WMI 接続が失敗した場合の再試行回数を指定します。	0～100	3
	wmi.connect.retry.interval	操作対象機器への WMI 接続が失敗した場合の再試行間隔（秒数）を指定します。	1～600	10
	telnet.connect.retry.times	操作対象機器への Telnet 接続が失敗した場合の再試行回数を指定します。	0～100	3
	telnet.connect.retry.interval	操作対象機器への Telnet 接続が失敗した場合の再試行間隔（秒数）を指定します。	1～600	10
メール通知の再試行	mail.notify.retry.times	メールを送信する通知機能が失敗した場合の再試行回数を指定します。	0～100	3
	mail.notify.retry.interval	メールを送信する通知機能が失敗した場合の再試行間隔（秒数）を指定します。	1～600	10
	mail.plugin.retry.times	メール通知部品でのメール送信が失敗した場合の再試行回数を指定します。	0～100	3
	mail.plugin.retry.interval	メール通知部品でのメール送信が失敗した場合の再試行間隔（秒数）を指定します。	1～600	10
監査ログ	logger.Audit.command.useLoginUserID	コマンドが実行されるときの監査ログのサブジェクト識別情報に、ユーザー ID として HAD ログインユーザー ID を出力するかどうかを指定します。	true/false	false
ウィンドウの更新	client.events.refreshinterval	イベントの更新間隔（秒数）を指定します。	0～65535	5

分類	キー名	設定	値	デフォルト値
エディタ	client.editor.upload.maxfilesize	[Service Builder Edit] ウィンドウで、Automation Director の操作に使用される端末からサーバにアップロードできる最大ファイルサイズ（MB 単位）を指定します。	1～10	3
	client.editor.canvas.maxwidth	[フロー] ビューの幅の最大サイズ（px 単位）を指定します。	3600～10000	3600
	client.editor.canvas.maxhigh	[フロー] ビューの高さの最大サイズ（px 単位）を指定します。	2400～30000	2400
	server.editor.step.perTemplate.maxnum	サービステンプレートあたりの最大ステップ数を指定します。	320～40000	320
	server.editor.step.perLayer.maxnum	レイヤあたりの最大ステップ数を指定します。	80～10000	80
	server.editor.publicProperty.perTemplate.maxnum	サービステンプレートあたりのサービスプロパティの最大数を指定します。	100～2000	100
	server.editor.propertyGroup.perTemplate.maxnum	サービステンプレートあたりのプロパティグループの最大数を指定します。	5～1000	500
デバッガ	tasklist.debugger.autodelete.taskRemainingPeriod	サービステンプレートあたりのプロパティグループの最大数を指定します。	1～90	7
	client.debugger.tasklog.maxfilesize	[タスクログ] タブに表示されるタスクログのサイズ（KB）を指定します。	4～10240	1024
	logger.debugger.TA.MaxFileSize	デバッグタスクの最大ログファイルサイズ（KB）を指定します。	4～2097151	10240
Task Monitor	client.monitor.tasklog.maxfilesize	[タスクログ] ダイアログボックスに表示されるタスクログのサイズ（KB）を指定します。	4～10240	1024
	client.monitor.tasklog.refresh.interval	[タスクログ] ダイアログボックスの自動更新間隔（秒数）を指定します。	30～300	30

分類	キー名	設定	値	デフォルト値
	client.monitor.status.interval	task monitor の自動更新間隔（秒数）を指定します。	30～300	30
LongRunningTask チェック間隔閾値	server.longRunning.check.interval	LongRunningTask チェック間隔閾値（分数）	0～20160	2880
LongRunning 監視間隔	server.longRunning.monitor.interval	LongRunning 監視間隔（秒数）	1～3600	60
Web クライアント	plugin.http.read.timeout	HTTP/HTTPS 接続が確立されるときのタイムアウト値（秒数）を指定します。0 を指定した場合、タイムアウトは発生しません。	0～3600	60
	plugin.http.read.timeout	HTTP/HTTPS 接続が確立されるときのタイムアウト値（秒数）を指定します。0 を指定した場合、タイムアウトは発生しません。	0～86400	600

[1] タスクのログ出力閾値は、サービス共有プロパティで設定できます。

例

logger.sysloglevel = 0

logger.message.server.MaxBackupIndex = 7

logger.message.server.MaxFileSize = 1024

logger.message.command.MaxBackupIndex = 7

logger.message.command.MaxFileSize = 1024

logger.TA.MaxFileSize = 1024

tasklist.autoarchive.taskRemainingPeriod = 7

tasklist.autoarchive.executeTime = 04:00:00

tasklist.autoarchive.maxTasks = 5000

tasklist.autodelete.maxHistories = 30000

mail.notify.retry.times = 3

mail.notify.retry.interval = 10

mail.plugin.retry.times = 3

mail.plugin.retry.interval = 10

client.events.refreshinterval = 5

3.7 コマンドプロパティファイル（command_user.properties）により HAD サーバとの通信用ポート番号を変更する

これは、コマンドを実行するときに使用される http ポートを設定する定義ファイルです。Automation Director と Web ブラウザ間の通信に使用するポート番号を変更した場合は、コマンドを実行するときに使用する http ポートも同じ番号に変更する必要があります。これは、定義ファイルを更新するために必要です。

形式

specification-key-name=setting

インストール先フォルダ

HAD-installation-folder¥conf

説明

1 行に 1 つの指定キーと設定を指定します。コマンドプロパティファイルを編集するときには、次のことに注意してください。

- #で始まる行は、コメントとして扱われます。
- 空白行は無視されます。
- エンコードは ISO8859-1 です。
- エントリでは大文字と小文字が区別されます。
- 文字列の中で¥を指定するには、¥¥と入力する必要があります。
- 設定として無効な値を入力した場合はデフォルト値に設定され、メッセージ KNAE02022-W が統合トレースログとパブリックログに出力されます。
- 1 つのファイル内で同じ指定キーが複数回入力された場合は、最後に入力したキーが有効になります。

設定

キー名	設定	値	デフォルト値
command.http.port	コマンドを実行するときに使用される http ポートを指定します。	1～65535	22015

定義の例

```
command.http.port = 22015
```

3.8 メール通知定義を変更する

これは、障害発生時またはタスク内での異常検出時の、メール通知の定義ファイルです。

形式

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<mail xmlns="http://www.hitachi.com/products/it/software/xml/automation/
conf/mailDefinition">
<title>email-title</title>
<body>email-body</body> </mail>
```

インストール先フォルダ

HAD-installation-folder¥conf

説明

メール通知の定義ファイルは XML 形式です。編集する箇所は、email-title および email-body セクションです。

ファイルを編集するときには、次のことに注意してください。

- メール通知の定義ファイルがない場合や整形式 XML でない場合、読み取りエラーが発生します。この場合、メールはデフォルトの件名と本文で送信されます。
- <mail>、<title>、および<body>の外部でタグを指定した場合、タグが整形式 XML であっても、タグとその内容は無視されます。
- <title>または<body>タグが省略された場合には、値として空の文字列が指定されます。
- <mail>タグを省略することはできません。省略した場合、形式は無効であり、読み取りエラーが発生します。
- エントリでは大文字と小文字が区別されます。

メール通知定義ファイルの設定

設定	XML 要素	文字列長	デフォルト値
メール通知で使用されるメールの件名	<title>	0～9,999 バイトの文字列	[HCS Automation] $TASK_NAME$は $TASK_STATUS$に変更されました。
メール通知で使用されるメールの本文	<body>	0～9,999 バイトの文字列	サービスグループ名：$SERVICE_GROUP_NAME$ タスク名：$TASK_NAME$ ユーザー名：$USER_NAME$ タスク詳細：$TASK_DETAIL_URL$

XML エンティティ参照

メールに表示する文字	入力する文字
&	&amp;
<	&lt;
>	&gt;
"	&quot;
'	&apos;

メール通知定義ファイルの埋め込み文字

埋め込まれる文字	項目	備考
$SERVICE_GROUP_NAME$	サービスグループ名	リソースグループ名を表す文字列に設定します。
$TASK_NAME$	タスク名	タスクのプロパティの形式に従って設定します。
$TASK_ID$	タスク ID	
$TASK_KIND$	タスクの種類	
$SERVICE_NAME$	サービス名	
$TASK_TAGS$	タスクのタグ	
$TASK_STATUS$	タスクのステータス	
$EXECUTION_DATE$	操作の実行日時	
$PLANNED_START_DATE$	開始予定日時	
$START_DATE$	実際の開始日時	
END_DATE	終了日時	
$USER_NAME$	操作を実行するユーザー	
$TASK_DETAIL_URL$	［タスク詳細］ウィンドウの URL	http で始まる URL に設定します。

3.9 セキュリティ定義ファイル（security.conf）でパスワードポリシーを変更する

これは、ユーザーパスワードの条件とロックに関する設定の定義ファイルです。このファイルで HCS パスワードポリシーを変更し、必要に応じてセキュリティ設定をカスタマイズできます。

形式

specification-key-name=setting

インストール先フォルダ

Common-Component-installation-folder¥conf¥sec

説明

1 行に 1 つの指定キーと設定を指定します。セキュリティ定義ファイルのデフォルトの状態は、次のとおりです。

```
# This is the minimum length of the password
# (minimum: 1 -256 characters)
password.min.length=4

# This is the minimum number of uppercase characters included in the
password
# (minimum: 0-256 characters, character type: A-Z)
password.min.uppercase=0

# This is the minimum number of lowercase characters included in the
password
# (minimum: 0-256 characters, character type: a-z)
password.min.lowercase=0

# This is the minimum number of numeric characters included in the
password
# (minimum: 0-256 characters, character type: 0-9)
password.min.numeric=0

# This is the minimum number of symbolic characters included in the
password
# (minimum: 0-256 characters, character type: !# $ % & ' ( ) * + - .= @ ¥
^ _ |)
password.min.symbol=0

# This specifies whether the user ID can be used for the password
# (true = cannot use the user ID, false = can use the user ID)
password.check.userID=false

# This is the minimum number of login failures before an account is
locked
# (minimum: 0-10 times)
account.lock.num=0
```

設定

キー名	設定	設定可能な値	デフォルト値
password.min.length	パスワードの最小文字数を指定します。	1～256	4
password.min.uppercase	パスワードに含むべき大文字の最小数を指定します。0 を指定した場合、大文字の数に関する制約はありません。	0～256	0
password.min.lowercase	パスワードに含むべき小文字の最小数を指定します。0 を指定した場合、小	0～256	0

キー名	設定	設定可能な値	デフォルト値
	文字の数に関する制約はありません。		
password.min.numeric	パスワードに含むべき数字の最小数を指定します。0 を指定した場合、文字の数に関する制約はありません。	0～256	0
password.min.symbol	パスワードに含むべき記号の最小数を指定します。0 を指定した場合、記号の数に関する制約はありません。	0～256	0
password.check.userID	ユーザー ID と同じパスワードを防止するかどうかを指定します。	• true：防止します • false：防止しません	false
account.lock.num	アカウントが自動的にロックされるまでのログインの連続失敗回数を指定します。0 を指定した場合、ログインの試みが失敗してもアカウントは自動的にロックされません。	0～10	0

3.10 操作対象機器との接続に使用される情報を構成する

始める前に：

- 次のパスにあるすべてのファイルは、接続先プロパティファイルとみなされます。
  パス： <HAD-installation-folder>¥Automation¥conf¥plugin¥destinations
- ファイル名には、次の形式を使用する必要があります。
  <Host name>.properties, <IPv4 address>.properties, <IPv6 address>.properties

  メモ IPv6 アドレス内のコロン「:」はファイル名には使用できないため、ダッシュ（-）に置き換えます。例：2001::234:abcd -> 2001--234-abcd.properties.

- デフォルトでは、#sample.properties ファイルは次の場所に保存されます： <HAD-installation-folder>¥Automation¥conf¥plugin¥destinations

プロパティファイルを編集するときには、次のことに注意してください。

- #で始まる行は、コメントとして扱われます。
- 空白行は無視されます。
- エンコードは ISO 8859-1 です。

- 内容は大文字と小文字が区別されます。
- 文字列内でスラッシュ（¥）を指定するには、二重スラッシュ（¥¥）を使用する必要があります。
- 接続先プロパティファイルで無効な値を指定した場合、接続先プロパティファイルを参照する部品で実行エラーが発生します。
- 1 つのファイル内で同じ指定キーを複数回入力した場合は、最後に指定したキーが有効になります。

対象機器に接続するには、以下の構成情報を使用してください。

対象機器がクラスタ環境の一部である場合の構成ガイドライン

次のことを守ってください。

- 操作対象機器の OS が Windows Server 2012 または Windows Server 2012 R2 クラスタ環境である場合、作業ディレクトリ（`wmi.workDirectory.sharedName` および `wmi.workDirectory.sharedPath`）を設定する必要があります。設定しなかった場合、部品は接続エラーになります。
- コンテンツ部品を含んだスクリプトを実行する場合は、実行ディレクトリ（`common.executionDirectory`）も指定する必要があります。設定しなかった場合、スクリプトファイルの転送は失敗します。

キー名	設定	指定可能な値	最小値	最大値
terminal.charset	通信に使用される文字セットを指定します。	EUC-JP eucjp ibm-943C ISO-8859-1 MS932 PCK Shift_JIS UTF-8 windows-31j	1	64
telnet.port	ターミナル接続部品での Telnet 接続に使用されるポート番号を指定します。この設定は、プロパティファイル（config_user.properties）の telnet.port.number 設定に優先します。	0～65535	0	65535
ssh.port	次のどれかの部品を使用して、SSH 接続に使用されるポート番号を指定します：汎用コマンド実行部品、ファイル転送部品、ターミナル接続部品、コンテンツ部品。この設定は、プ	0～65535	0	65535

	ロパティファイル（config_user.properties）の ssh.port.number 設定に優先します。			
telnet.prompt.account	ターミナル接続部品を使用して対象機器との接続を確立する際に出力されるユーザー ID の入力を求める文字列の検出に使用する、正規表現パターンを指定します。1～1,024 文字を使用できます。たとえば、「Username:」と指定します。	正規表現パターンで使用できる文字列。	1 文字	1024 文字
telnet.prompt.password	ターミナル接続部品を使用して対象機器との接続を確立する際に出力されるパスワードの入力を求める文字列の検出に使用する、正規表現パターンを指定します。1～1,024 文字を使用できます。たとえば、「Password:」と指定します。	正規表現パターンで使用できる文字列。	1 文字	1024 文字
telnet.noStdout.port.list	ターミナル接続部品を使用して接続が確立された後に標準出力を返さないサービスのポート番号を指定します。1～1,024 文字を使用できます。複数のポート番号を指定するには、区切り文字としてコンマを使用します。	0～65535 とコンマ (,)	1 文字	1024 文字
wmi.workDirectory.sharedName	これは、操作対象機器の OS が Windows の場合に有効なプロパティです。操作対象上でコマンドを実行するときに転送されるファイルが置かれる共有フォルダの名前を指定します。フォルダは	1 バイトの英数字、「-」、「_」、および「.」。	0 文字	80 文字

	wmi.workDirectory.sharedPath と同じである必要があります。このプロパティを使用する場合、操作対象の管理共有設定は不要です。0〜80 文字の文字列を指定します。			
wmi.workDirectory.sharedPath	これは、操作対象機器の OS が Windows の場合に有効なプロパティです。操作対象上でコマンドを実行するときに転送されるファイルが置かれる共有フォルダの絶対パスを指定します。汎用コマンド実行部品を実行する場合、実行ディレクトリは、このプロパティが示すパス下の"¥Hitachi¥CMALib¥HAD¥home"になります。フォルダは、wmi.workDirectory.sharedName と同じである必要があります。このプロパティを使用する場合、操作対象の管理共有設定は不要です。0〜80 文字の文字列を指定します。	1 バイトの英数字、「:」、「¥」、「-」、「_」、および「.」。	0 文字	80 文字
ssh.workDirectory	これは、操作対象機器の OS が Linux/Unix の場合に有効なプロパティです。ファイル転送部品またはコンテンツ部品で転送用ファイルが置かれるディレクトリの絶対パスを指定します。このプロパティで指定されたパスも、親ディレクトリのパスも、ファイル転送部品の接続先および受信先として	1 バイトの英数字、「/」、「-」、「_」、および「.」。	0 文字	128 文字

	指定することはできません。作業フォルダには、接続するユーザーの読み取り権限、書き込み権限、および実行権限が必要です。ファイル転送部品またはコンテンツ部品が実行されるときに、このプロパティで指定されたパスが存在しなかった場合、部品の実行時に作成されます。ディレクトリを作成できない場合、部品の実行は異常終了します。ディレクトリが新しく作成されるかどうかにかかわらず、指定されたディレクトリのアクセス権限を 777 に変更してください。優先されるのは、config_user.properties で定義された plugin.remoteCommand.workDirectory.ssh の値です。0～128 文字の文字列を指定します。			
common.executionDirectory	操作対象に対してコンテンツ部品を実行するときの実行ディレクトリを指定します。部品定義で定義された実行ディレクトリの値が設定されていなかった場合、このプロパティの値が適用されます。優先されるのは、config_user.properties で定義された plugin.remoteCommand.executionDirectory.wmi および plugin.remoteCommand.executionDirectory.ssh の値です。0～		0 文字	128 文字

	128 文字の文字列を指定します。"			

3.11 エージェントレス接続先の前提条件（Windows）

ここでは、エージェントレス接続先の前提条件について説明します。

サポートされるユーザー

以下のユーザーは、エージェントレス接続先で使用できます。

- ビルトイン Administrator
- administrators グループに属するユーザー(1)(2)
- Active Directory のビルトイン Administrator
- Active Directory の Domain Admin グループに属するユーザー(1)(2)

(1) UAC（ユーザーアクセス制御）昇格は、コマンド実行時に実行できません。

(2) 接続先の OS が以下の条件の 1 つに該当する場合は、レジストリ設定を実行する必要があります。

- Windows Server 2008 で UAC 機能が有効である
- Windows Server 2008 R2 で UAC 機能が有効である
- OS は Windows Server 2012 である
- OS は Windows Server 2012 R2 である

レジストリエディタを使用して、次のレジストリのキーのエントリを設定します。

メモ　OS の再起動は不要です。

項目	値
レジストリキー	HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows¥CurrentVersion¥Policies¥System
レジストリエントリ	LocalAccountTokenFilterPolicy
レジストリエントリとして設定される値	1 (DWORD)

オプションとして、コマンドプロンプトに次のコマンドを入力できます。

```
reg add HKEY_LOCAL_MACHINE¥SOFTWARE¥Microsoft¥Windows¥CurrentVersion
¥Policies¥System /v LocalAccountTokenFilterPolicy /t REG_DWORD /d 0x1 /f
```

管理共有設定

管理共有を使用して、レジストリエディタで次のレジストリのキーのエントリを設定し、OS を再起動します。

項目	値
レジストリキー	HKEY_LOCAL_MACHINE¥SYSTEM¥CurrentControlSet¥Services¥Lanmanserver¥parameters
レジストリエントリ	AutoShareServer
レジストリエントリとして設定される値	1 (DWORD)

コマンドプロンプトに次のコマンドを入力します。

```
reg add HKEY_LOCAL_MACHINE¥SYSTEM¥CurrentControlSet¥Services¥Lanmanserver
¥parameters /v AutoShareServer /t REG_DWORD /d 1
```

3.12 エージェントレス接続先の前提条件（SSH）

ここでは、以下の部品で SSH プロトコルを使用する場合の前提条件について説明します。

- コンテンツ部品
- 汎用コマンド実行部品
- ファイル転送部品
- ターミナル接続部品
- ターミナルコマンド実行部品
- ターミナル切断部品

メモ　SSH はバージョン 2 をサポートする必要があります。

3.12.1 パスワード認証

SSH サーバに対するパスワード認証を、次のように設定する必要があります。

1. リモート操作対象ホストに root としてログインします。
2. sshd_config ファイルを開きます。
   HP-UX の場合：/opt/ssh/etc/sshd_config
   その他の OS の場合：/etc/ssh/sshd_config
3. PubkeyAuthentication の値を yes に設定します。PubkeyAuthentication の行がコメントアウトされている場合は、コメントアウトのハッシュ記号（#）を削除します。
4. 次のコマンドを実行して、sshd サービスを再開します。
   RHEL／CentOS／SUSE Linux／Oracle Linux（RHEL 6.4 など）の場合： /etc/rc.d/init.d/sshd restart
   Solaris（Solaris 10 など）の場合：/usr/sbin/svcadm restart ssh
   AIX（AIX 6.1 など）の場合：kill -HUP [**Process ID of sshd**]
   HP-UX（HP-UX 11i V3 の例）：/sbin/init.d/secsh stop; /sbin/init.d/secsh start

メモ　これらのコマンドは、OS のバージョンによって変わることがあります。追加情報については、OS のマニュアルを参照してください。

3.12.2 公開鍵認証

ここでは、SSH サーバに接続する公開鍵を認証する方法について説明します。

SSH サーバのセットアップ

公開鍵認証を使用するには、SSH サーバに対する公開鍵認証を設定する必要があります。

1. リモート操作対象ホストに root としてログインします。
2. `sshd_config` を開きます。
   HP-UX：/opt/ssh/etc/sshd_config
   HP-UX 以外：/etc/ssh/sshd_config
3. `PubkeyAuthentication` の値を yes に設定します。`PubkeyAuthentication` の行がコメントアウトされている場合は、コメントアウトのハッシュ記号（#）を削除します。
4. 次のコマンドを実行して、`sshd` サービスを再開します。
   RHEL／CentOS／SUSE Linux／Oracle Linux（RHEL 6.4 など）の場合：`/etc/rc.d/init.d/sshd restart`
   Solaris（Solaris 10 など）の場合：`/usr/sbin/svcadm restart ssh`
   AIX（AIX 6.1 など）の場合：`kill -HUP [Process ID of sshd]`
   HP-UX（HP-UX 11i V3 の例）：`/sbin/init.d/secsh stop; /sbin/init.d/secsh start`

メモ　これらのコマンドは、OS のバージョンによって変わることがあります。追加情報については、OS のマニュアルを参照してください。

鍵の作成（初回）

公開鍵と秘密鍵を作成します。鍵は、HAD がインストールされる OS 上で作成することを強く推奨します。

メモ　秘密鍵を別の OS に移動すると、秘密鍵が漏えいしてセキュリティリスクを負う恐れがあります。ただし、別の OS 上で作成された鍵を使用することは可能です。

参考として、以下の手順では RHEL6.4（Linux）上で鍵を作成します。

1. **`ssh-keygen`** コマンドを実行します。
   RSA 鍵を作成する場合：`ssh-keygen -t rsa`
   DSA 鍵を作成する場合：`ssh-keygen -t dsa`
2. 秘密鍵の場所と名前を決めます。
   マルチバイト文字を含まないパスとファイル名を指定します。デフォルトでは、`~/.ssh/id_rsa` が設定されます（RSA 鍵を作成する場合）。秘密鍵は、選択されたパスに対して指定されたファイル名として設定されます。公開鍵は、秘密鍵と同じディレクトリに、秘密鍵の名前に「.pub」ファイル拡張子を付けたファイルとして設定されます。
3. パスフレーズを入力します。

パスフレーズを入力して、Return キーを押すように求められます。次に、パスフレーズの再入力を求められます。秘密鍵のパスフレーズを設定しない場合は、パスフレーズを入力せずに Return キーを押します。

HAD への秘密鍵の配置

HAD がインストールされる OS 上に秘密鍵を配置します。任意の場所に配置し、パスをプロパティファイル（config_user.properties）の ssh.privateKeyFile に設定します。

リモート操作対象ホストへの公開鍵の配置

1. **cat** コマンドの出力をリダイレクトし、生成された公開鍵ファイルの内容を、認証に使用される公開鍵ファイル（authorized_keys）に追加します。（例：cat id_rsa.pub >> authorized_keys）
2. **chmod** コマンドを実行して、authorized_keys の属性を 600 に変更します（書き込みおよび読み取り権限を所有者にのみ与えます）。属性が 600 でない場合、部品実行時に認証が失敗することがあります。
   デフォルトでは、authorized_keys の配置場所は、~/.ssh の直下になっています。~/.ssh に関しては、属性を 700 に変更します（書き込み、読み取り、および実行権限を所有者にのみ与えます）。

shared property の構成

1. HAD アプリケーションにログインします。
2. [管理] > [サービス共有プロパティ]を選択します。
3. 秘密鍵のパスフレーズを開きます（SSH 公開鍵認証の場合）。
4. 値としてパスフレーズを入力します。
   値は、秘密鍵のパスフレーズです（SSH 公開鍵認証の場合）。

3.12.3 キーボードインタラクティブ認証

キーボードインタラクティブ認証を使用するには、認証を SSH サーバに設定する必要があります。

1. リモート操作対象ホストに root としてログインします。
2. sshd_config を開きます。
   HP-UX：/opt/ssh/etc/sshd_config
   HP-UX 以外：/etc/ssh/sshd_config
3. 次のようにキーボードインタラクティブ認証を設定します。
   RHEL/CentOS/SUSE、Linux/Oracle Linux、Linux/AIX/HP-UX の場合
   - ChallengeResponseAuthentication の値を yes に設定します。
   （ChallengeResponseAuthentication の行がコメントアウトされている場合は、コメントアウトのハッシュ記号（#）を削除します。）
   - UsePAM の値を yes に設定します。（lUsePAM の行がコメントアウトされている場合は、コメントアウトのハッシュ記号（#）を削除します。

   Solaris10 の場合
   - PAMAuthenticationViaKBDInt の値を yes に設定します。
   （PAMAuthenticationViaKBDInt の行がコメントアウトされている場合は、コメントアウトのハッシュ記号（#）を削除します。）

   Solaris11 の場合

- KbdInteractiveAuthentication の値を yes に設定します。
（KbdInteractiveAuthentication の行がコメントアウトされている場合は、コメントアウトのハッシュ記号（#）を削除します。）

4. AIX の場合、以下の設定を行います。

メモ　AIX OS 以外の場合は、設定を変更する必要はありません。

- /etc/pam.conf を開き、以下を追加します。
Authentication ブロックの内側
sshd auth required /usr/lib/security/pam_aix を追加します。

Account Management ブロックの内側
sshd account required /usr/lib/security/pam_aix を追加します。

Password Management ブロックの内側
sshd auth required /usr/lib/security/pam_aix.を追加します。

Password Management ブロックの内側
sshd password required /usr/lib/security/pam_aix を追加します。

Session Management ブロックの内側
sshd session required /usr/lib/security/pam_aix を追加します。

- /etc/ssh/sshd_config を開いて、次の行を変更します。
UsePAM = no を UsePAM = yes に変更します。（UsePAM の行がコメントアウトされている場合は、コメントアウトのハッシュ記号（#）を削除します。）

- /etc/security/login.cfg を開いて、次の行を変更します。
auth_type = STD_AUTH を auth_type = PAM_AUTH に変更します。（auth_type の行がコメントアウトされている場合は、コメントアウトのハッシュ記号（#）を削除します。）

5. 次のコマンドを実行して、sshd サービスを再開します。サポートされる各 OS についてコマンド例を示します。
RHEL/CentOS/SUSE Linux/Oracle Linux（RHEL 6.4 など）の場合：
/etc/rc.d/init.d/sshd restart

Solaris（Solaris 10 など）の場合：
/usr/sbin/svcadm restart ssh

AIX（AIX 6.1 など）の場合：
kill -HUP [**Process ID of sshd**]

HP-UX（HP-UX 11i V3 など）の場合：
/sbin/init.d/secsh stop; /sbin/init.d/secsh start

メモ　これらのコマンドは、オペレーティングシステムのバージョンによって変わる場合があります。詳細については、該当する OS のマニュアルを参照してください。

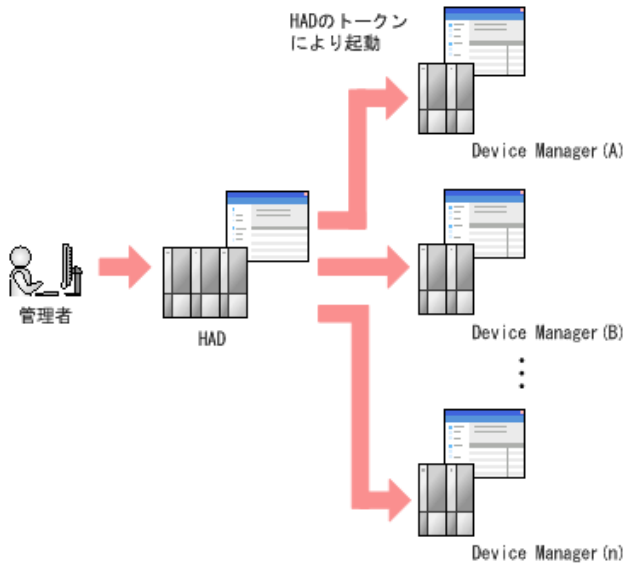
3.13 1 つの HAD サーバから複数の Device Manager を使用する

Hitachi Automation Director では、1 つの HAD サーバから複数の Device Manager を使用することができます。この機能は、1 つのトークンだけを（主に）使用する複数の共通コンポーネント認証サーバ間の相互認証を使用することによって可能になります。

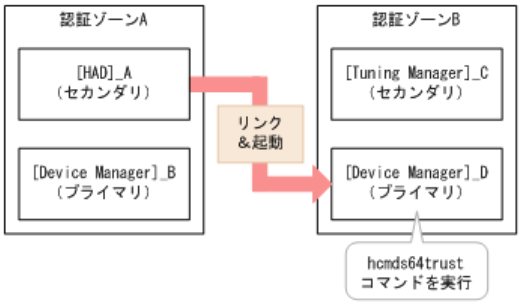
相互認証とは、クライアント／サーバ接続経由でアプリケーショントラフィックを送信する前にクライアントがサーバに身元を証明しなければならず、サーバがクライアントに身元を証明しなければならないセキュリティ機能です。

メモ　相互認証は、システムアカウントや共通コンポーネントの内部アカウント（セットアップやその他の内部機能に使用される）などのビルトインアカウントでは行うことができません。

次の図は、1 つの HAD サーバから複数の Device Manager を操作する例を示しています。



次の図は、下記のガイドラインに基づく相互認証を示しています。



ガイドライン

1. サーバ ID を変更するには、**hcmds64chgtsid** コマンドを使用します。サーバ ID がデフォルトのホスト名の場合、このステップは不要です。
2. [Device Manager]_D で **hcmds64trust** コマンドを実行し、[Device Manager]_B の接続先情報を登録します。
3. 認証ゾーン A と認証ゾーン B で相互認証を行うユーザーの設定を、次のように選択します。

   - 共通ユーザー管理に登録されたユーザーを使用する場合は、認証ゾーン A と認証ゾーン B の共通ユーザー管理に同じユーザーを登録し、権限を付与します。
   - 共通ユーザー管理に登録されていない外部認証グループのユーザーを使用する場合は、グループ DN（外部認証グループのユーザーは認証サーバに属します）を認証ゾーン A と認証ゾーン B の共通ユーザー管理に登録し、権限を付与します

4

Hitachi Automation Director を削除する

この章では、Hitachi Automation Director を削除する方法について説明します。

4.1 Hitachi Automation Director を削除する（Windows）

Windows 環境で Hitachi Automation Director を削除するには、次のセクションに記載されている手順に従います。

前提条件

- Hitachi Automation Director の**タスク**タブの状態列が待機中、応答待ち中、実行中、長期実行中、異常検出のいずれかの状態になっているタスクがある場合には、タスクが停止するか実行を終了するまで待ちます。
- すべてのサービスダイアログボックスを閉じます。
- Windows のサービスまたは開いているコマンドプロンプトを閉じます。
- サーバ上のセキュリティ監視、ウイルス検出、またはプロセス監視ソフトウェアを無効にします。

注意　他の Hitachi Command Suite 製品が同じホストにインストールされている場合は、共有フォルダ（¥Base¥database）を削除しないでください。このフォルダを削除すると、他の Hitachi Command Suite 製品が停止します。

操作手順

1. Windows に管理者としてログオンします。
2. 次のコマンドを実行して、すべてのサービスを停止します。
   Common-Component-installation-folder¥bin¥hcmds64srv /stop
3. ［Control Panel］を開き、［Programs and Features］または［Add or Remove Programs］を選択します。
4. ［Automation Director］を選択して［Remove］をクリックするか、プログラムを選択し、右クリックして［Uninstall］を選択します。
5. ［Setup］ウィンドウで［Uninstallation］をクリックして、ソフトウェア削除プロセスを開始します。
   削除プロセスによって、*HAD-installation-folder*¥Automation フォルダが削除されます。

操作結果

Automation Director がホストから削除されます。

4.2 Hitachi Automation Director を削除する（Linux）

Linux 環境で Hitachi Automation Director を削除するには、次の手順に従います。

操作手順

1. root ディレクトリ（cd /root など）に移動します。
2. 次のコマンドを実行します：*<Installation directory of HAD>*/ADUninstall/uninstall.sh

4.3 クラスタ環境で Hitachi Automation Director を削除する

Hitachi Automation Director を別のサーバに移行するか、運用を中止する場合には、クラスタ環境のサーバから Hitachi Automation Director ソフトウェアを削除できます。

メモ　Hitachi Automation Director を削除した場合、プロパティファイル、ログファイル、その他の製品関連のファイルが削除されます。

操作手順

1. クラスタ管理ソフトウェアで、Hitachi Automation Director サービスが登録されているグループをスタンバイノードからアクティブノードに移動します。グループを右クリックして［Move］を選択し、［**Select Node**］または［**Move this service or application to another node**］を選択します。
2. 次のコマンドを使用して、Hitachi Automation Director を含む Hitachi Command Suite サービスが登録されているグループをオフラインにして、フェイルオーバーを無効にします。

   Hitachi-Command-Suite-Common-Component-installation-directory
   ¥ClusterSetup¥hcmds64clustersrvstate /soff /r *HCS-cluster-group-name*

   ここで、

   r - Hitachi Automation Director を含む Hitachi Command Suite 製品のサービスが登録されるグループの名前を指定します。グループ名にスペースが含まれる場合は、グループ名を引用符（"）で囲む必要があります。たとえば、グループ名が HAD cluster の場合は、"HAD cluster"と指定します。
3. 次のコマンドを使用して、Hitachi Automation Director を含む Hitachi Command Suite サービスを削除します。

   メモ　サービスを削除する前に、クラスタ管理ソフトウェアから customer script を削除します。

   Hitachi-Command-Suite-Common-Component-installation-directory
   ¥ClusterSetup¥hcmds64clustersrvupdate /sdel /r *HCS-cluster-group-name*

   ここで、

   r - Hitachi Automation Director を含む Hitachi Command Suite 製品のサービスが登録されるグループの名前を指定します。グループ名にスペースが含まれる場合は、グループ名を引用符（"）で囲む必要があります。たとえば、グループ名が HAD cluster の場合は、"HAD cluster"と指定します。

   メモ
   - r オプションで指定されたグループに登録されているすべての Hitachi Automation Director と Hitachi Command Suite 製品のサービスが削除されます。ただし、Hitachi File Services Manager のサービスは削除されません。
   - Hitachi Command Suite 製品を引き続き使用する場合は、Hitachi Automation Director を削除した後で再登録できます。Hitachi Automation Director サービスを削除しても、問題はありません。サービスリソース名を変更していた場合、サービスが再登録されるときに、すべてのリソース名が再初期化されます。したがって、削除するサービスのリソース名を記録しておき、それらのサービスの再登録後に名前を変更する必要があります。
4. ユーザースクリプト（stopcluster /prepare コマンドを発行するスクリプト）をクラスタソフトウェアから削除します。
5. 次のコマンドを使用して、Hitachi Command Suite 製品を停止します。

HCS-Common-Component-installation-folder¥bin¥hcmds64srv /stop

6. アーカイブノードから Hitachi Automation Director を削除します。
7. アクティブノードで、不要になったファイルとフォルダ（クラスタ環境でのインストール時に作成されたファイルとフォルダなど）を削除します。
8. クラスタ管理ソフトウェアで、Hitachi Automation Director services group をスタンバイノードに移動します。グループを右クリックして［Move］を選択してから、［Select Node］または［Move this service or application to another node］を選択します。
9. スタンバイノードから Hitachi Automation Director を削除します。
10. クラスタインストールの削除を実行した後、Automation フォルダを削除して、他の HCS サービスを使用しない場合は、スタンバイノードから Base フォルダも削除します。
11. 以下のリソースが他のアプリケーションによって使用されていない場合は、クラスタ管理ソフトウェアを使用して、それらをオフラインにしてから削除します。
    - IP アドレス
    - 共有ディスク
12. スタンバイノードで、不要になったファイルとフォルダ（クラスタ環境でのインストール時に作成されたファイルとフォルダなど）を削除します。
13. 他の Hitachi Command Suite 製品を引き続き使用する場合は、次のコマンドを使用して、Hitachi Command Suite サービスをクラスタ管理ソフトウェアグループに登録します。

    HCS-Common-Component-installation-folder¥ClusterSetup
    ¥hcmds64clustersrvupdate /sreg /r *HCS-cluster-group-name* /sd *drive-letter-of-shared-disk* /ap *resource-name-for-client-access-point*

    ここで、

    r - Hitachi Command Suite 製品のサービスを登録するグループの名前を指定します。グループ名にスペースが含まれる場合は、グループ名を引用符（"）で囲む必要があります。たとえば、グループ名が HAD cluster の場合は、"HAD cluster"と指定します。

    sd - クラスタ管理ソフトウェアに登録される共有ディスクのドライブ名を指定します。このオプションに対して複数のドライブ名を指定することはできません。Hitachi Command Suite 製品のデータベースが複数の共有ディスクに分割されている場合は、各共有ディスクについて hcmds64clustersrvupdate コマンドを実行します。

    ap - クラスタ管理ソフトウェアに登録されるクライアントアクセスポイント用リソースの名前を指定します。
14. 他の Hitachi Command Suite 製品を引き続き使用する場合は、次のコマンドを使用して、Hitachi Command Suite サービスが登録されるグループをオンラインにして、フェイルオーバーを有効にします。

    HCS-Common-Component-installation-folder¥ClusterSetup
    ¥hcmds64clustersrvstate /son /r *HCS-cluster-group-name*

    ここで、

    r - Hitachi Command Suite 製品のサービスが登録されるグループの名前を指定します。グループ名にスペースが含まれる場合は、グループ名を引用符（"）で囲む必要があります。たとえば、グループ名が HAD cluster の場合は、"HAD cluster"と指定します。
15. クラスタ管理ソフトウェアで、Hitachi Command Suite のリソースを含んでいるグループをアクティブノードに移動します。グループを右クリックして［Move］を選択してから、［Select Node］または［Move this service or application to another node］を選択します。

4.4 認証データを削除する

削除が正常に完了したにもかかわらず KNAE04574-E 警告ダイアログボックスが表示された場合、認証データの削除は失敗しています。認証データを削除するには、ユーザーアカウントを管理する

サーバ（Device Manager がインストールされている接続先のホスト）上で hcmds64intg コマンドを実行します。

hcmds64intg コマンドを実行して、Windows ホストから認証データを削除するには：

操作手順

1. 次のコマンドを実行して、インストールされている Hitachi Command Suite 製品のすべてのサービスを開始します。
   `Common-Component-installation-folder¥bin¥hcmds64srv /start`
2. 次のコマンドを実行して、認証データを削除します。
   Common-Component-installation-folder`¥bin¥hcmds64intg /delete / type`*component-name* `/ user`*user-ID* `/pass`*password*

   - `/type`
     削除するコンポーネントの名前を指定します。`Automation` を指定できます。
   - `/user`
     Admin（ユーザー管理）権限を持つユーザーのユーザー ID を指定します。user オプションを指定せずにコマンドを実行した場合、ユーザー ID の指定を求められます。
   - `/pass`
     Admin（ユーザー管理）権限を持つユーザーのパスワードを指定します。pass オプションを指定せずにコマンドを実行した場合、パスワードの指定を求められます。

メモ　認証データを削除せずに、別の Hitachi Command Suite 製品の GUI ウィンドウを表示した場合、Automation サーバを削除した後でも、次のような問題が発生することがあります。

- Automation サーバのユーザー管理情報が表示される。
- ダッシュボードにある Automation サーバを起動するためのボタンが有効になる。ボタンをクリックすると、リンクエラーが表示される。

A

Hitachi Automation Director のファイルの場所とポート

この付録には、Hitachi Automation Director インストールの一部として作成されるすべてのフォルダが含まれています。

A.1 Automation Director のファイルの場所

インストール先フォルダ

次の表は、Hitachi Automation Director をインストールしたときに作成されるフォルダを示しています。「フォルダの詳細」列にはデフォルトのパスが示されていますが、インストール時に変更できます。

Windows フォルダの詳細	Windows フォルダの場所
インストール先フォルダ	*system-drive*¥Program Files¥HiCommand¥Automation
コマンドファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥bin
構成ファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥conf
サービステンプレートのフォルダ	*system-drive*¥Program Files¥HiCommand¥Automation¥contents
データファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥data
ヘルプファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥docroot
事前設定プロパティ定義ファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥extra_presets
インストールおよびアンインストール時の一時作業フォルダ	*system-drive*¥Program Files¥HiCommand¥Automation¥inst
ライブラリファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥lib
ログファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥logs
オープンソースソフトウェアのソースファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥ossSource
システムファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥system
内部コマンドによって使用される作業用ファイル	*system-drive*¥Program Files¥HiCommand¥Automation¥webapps
作業用フォルダ	*system-drive*¥Program Files¥HiCommand¥Automation¥work
共通コンポーネント	*system-drive*¥Program Files¥HiCommand¥Base64

Linux フォルダの詳細	Linux ディレクトリの場所
インストール先フォルダ	/opt/HiCommand/Automation
コマンドファイル	/opt/HiCommand/Automation/bin
構成ファイル	/opt/HiCommand/Automation/conf
サービステンプレートのフォルダ	/var/opt/HiCommand/Automation/contents

Linux フォルダの詳細	Linux ディレクトリの場所
データファイル	/var/opt/HiCommand/Automation/data
ヘルプファイル	/opt/HiCommand/Automation/docroot
事前設定プロパティ定義ファイル	/var/opt/HiCommand/Automation/extra_presets
インストールおよびアンインストール時の一時作業フォルダ	/opt/HiCommand/Automation/inst
ライブラリファイル	/opt/HiCommand/Automation/lib
ログファイル	/var/opt/HiCommand/Automation/logs
オープンソースソフトウェアのソースファイル	/opt/HiCommand/Automation/ossSource
システムファイル	/opt/HiCommand/Automation/system
内部コマンドによって使用される作業用ファイル	/var/opt/HiCommand/Automation/work
共通コンポーネント	/opt/HiCommand/Base64

A.2 ポート設定

Hitachi Automation Director は、以下のポート設定を使用します。

外部接続ポート

ポート番号	ファイアウォール	説明
22/tcp	HAD ←→ 操作対象	SSH に使用されます。 cjstartweb は、このポートを使用します。
23/tcp	HAD ←→ 操作対象	Telnet に使用されます。 cjstartweb は、このポートを使用します。
445/tcp または udp	HAD ←→ 操作対象	共有管理に使用されます。 cjstartweb は、このポートを使用します。
135/tcp および 139/tcp	HAD ←→ 操作対象	共有管理に使用されます。 cjstartweb は、このポートを使用します。
22015/tcp	ブラウザ → HAD	HBase Storage Mgmt Web Service へのアクセスに使用。非 SSL（非セキュア）通信では、初期設定が必要です。 このポート番号は変更できます。 httpsd は、このポートを使用します。
22016/tcp	ブラウザ → HAD	HBase Storage Mgmt Web Service へのアクセスに使用。SSL（セキュア）通信では、設定が必要です。 このポート番号は変更できます。 httpsd は、このポートを使用します。

ポート番号	ファイアウォール	説明
25/tcp	HAD → SMTP サーバ	メール送信に使用されます。 このポート番号は変更できます。 cjstartweb は、このポートを使用します。
88/tcp または udp	HAD → Kerberos サーバ	cjstartweb は、このポートを使用します。
359/tcp	HAD → LDAP ディレクトリサーバ	ldap/tls に使用されます。 cjstartweb は、このポートを使用します。
636/tcp	HAD → LDAP ディレクトリサーバ	LDAP に使用されます。 このポート番号は変更できます。 cjstartweb は、このポートを使用します。
1812/udp	HAD → Radius サーバ	cjstartweb は、このポートを使用します。

内部接続ポート

メモ　これらのポートは予約済みであり、内部ポート接続にのみ使用されます。

ポート番号	ファイアウォール	説明
20245/tcp	タスク処理エンジン ←→ タスク処理エンジン	マネージャのジョブステータス通知に使用されます。 jp1ajs2report は、このポートを使用します。
20250/tcp	HAD → タスク処理エンジン	タスク処理エンジンは、このポートを使用します。 ajscdinetd は、このポートを使用します。 HAD は、常にこのポートを使用します。
23031/tcp	HAD → HAD	以下のサービスへのアクセスに使用されます。 - HBase Storage Mgmt Web SSO Service - HSSO 専用 Web サーバ cjstartweb は、このポートを使用します。
23160/tcp	Jobnet コネクタ実行ホスト ←→ 接続実行時の Jobnet 実行ホスト	スケジューラサービス間の通信に使用されます。 jp1ajs2gw は、このポートを使用します。
23800/tcp	タスク処理エンジン ←→ タスク処理エンジン	クラスタ構成でのタスク処理エンジンの埋め込みデータベースで使用されます。 EmbeddedEdition_JF1 は、このポートを使用します。

B

hcmds64keytool ユーティリティを使用する

hcmds64keytool ユーティリティは、次のようにさまざまな方法で使用できます。

- 証明書をトラストストアにインポートする。
- トラストストアから証明書を削除する。
- Device Manager サーバの自己署名証明書をエクスポートする。
- トラストストアの一意の名前、トラストストアファイル名、およびパスワードを指定する。
- トラストストアにインポートされた証明書をチェックする。

メモ　この操作は、証明書が正しくインポートされたことを確認するのに役立ちます。

詳細については、『*Hitachi Command Suite システム構成ガイド*』を参照してください。

索引

ふ

ほ

ま

ら

わ