

Hitachi Ops Center Automator

インストールガイド

4010-1J-035

対象製品

Hitachi Ops Center Automator 10.8.0

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ます、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

HITACHI は、株式会社 日立製作所の商標または登録商標です。

Active Directory は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Cisco は、Cisco Systems, Inc. またはその関連会社の米国およびその他の一定の国における登録商標または商標です。

Internet Explorer は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Linux は、Linus Torvalds 氏の日本およびその他の国における登録商標または商標です。

Microsoft は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Red Hat is a registered trademark of Red Hat, Inc. in the United States and other countries.

Red Hat は、米国およびその他の国における Red Hat, Inc. の登録商標です。

Red Hat Enterprise Linux is a registered trademark of Red Hat, Inc. in the United States and other countries.

Red Hat Enterprise Linux は、米国およびその他の国における Red Hat, Inc. の登録商標です。

UNIX は、The Open Group の商標です。

Windows は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Windows Server は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were developed at the National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign.

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/>

1. This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)

2. This product includes cryptographic software written by Eric Young (ey@cryptsoft.com)

3. This product includes software written by Tim Hudson (tjh@cryptsoft.com)

4. This product includes the OpenSSL Toolkit software used under OpenSSL License and Original SSLeay License. OpenSSL License and Original SSLeay License are as follow:

LICENSE ISSUES

=====

The OpenSSL toolkit stays under a double license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit.

See below for the actual license texts.

OpenSSL License

/* =====

* Copyright (c) 1998-2019 The OpenSSL Project. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. All advertising materials mentioning features or use of this
 * software must display the following acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"
 *
 * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 * endorse or promote products derived from this software without
 * prior written permission. For written permission, please contact
 * openssl-core@openssl.org.
 *
 * 5. Products derived from this software may not be called "OpenSSL"
 * nor may "OpenSSL" appear in their names without prior written
 * permission of the OpenSSL Project.
 *
 * 6. Redistributions of any form whatsoever must retain the following
 * acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"
 *
 * THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 * =====
 *
 * This product includes cryptographic software written by Eric Young
 * (eay@cryptsoft.com). This product includes software written by Tim

```

* Hudson (tjh@cryptsoft.com).
*
*/
Original SSLeay License
-----
/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
*
* This package is an SSL implementation written
* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscapes SSL.
*
* This library is free for commercial and non-commercial use as long as
* the following conditions are aheared to. The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code. The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
* must display the following acknowledgement:
* "This product includes cryptographic software written by
* Eric Young (eay@cryptsoft.com)"
* The word 'cryptographic' can be left out if the rouines from the library
* being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
* the apps directory (application code) you must include an acknowledgement:
* "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL

```

* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

*

* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]

*/

Oracle および Java は、オラクルおよびその関連会社の登録商標です。

This product includes software developed by IAIK of Graz University of Technology.

This product includes software developed by Daisuke Okajima and Kohsuke Kawaguchi (<http://relaxngcc.sf.net/>).

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 Andy Clark

Java is a registered trademark of Oracle and/or its affiliates.



その他記載の会社名，製品名などは，それぞれの会社の商標もしくは登録商標です。

発行

2021 年 10 月 4010-1J-035

著作権

All Rights Reserved. Copyright© 2021, Hitachi, Ltd.

目次

はじめに.....	11
対象読者.....	12
マニュアルの構成.....	12
マイクロソフト製品の表記について.....	12
関連マニュアル.....	13
このマニュアルで使用している記号.....	13
KB（キロバイト）などの単位表記について.....	14
このマニュアルでの表記.....	14
 1.概要.....	17
1.1 製品の概要.....	18
1.2 関連する Hitachi Ops Center 製品について.....	18
1.3 Ops Center Automator システム構成.....	18
1.4 Ops Center Automator のインストールと構成のワークフロー.....	19
1.5 Ops Center Automator での認証方法.....	20
 2.Ops Center Automator をインストールする.....	21
2.1 インストールの前提条件.....	22
2.1.1 サーバ時刻を変更する.....	22
2.1.2 名前解決設定を変更する.....	23
2.1.3 ポートの衝突を回避する.....	23
2.2 Ops Center Automator をインストールする（Windows）.....	24
2.3 クラスタ環境で Ops Center Automator をインストールする（Windows）.....	24
2.3.1 クラスタ環境での Ops Center Automator の使用について.....	25
2.3.2 クラスタインストールワークフロー.....	25
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確認する.....	27
2.3.4 アクティブノードで Ops Center Automator クラスタ化をセットアップする.....	27
2.3.5 スタンバイノードで Ops Center Automator クラスタ化をセットアップする.....	28
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.....	29
2.4 Ops Center Automator をインストールする（Linux）.....	30
2.5 データベースフォルダのウイルススキャンを抑止する.....	30
2.6 インストール後のタスク.....	31
2.6.1 登録済み URL を変更する（Windows）.....	31
2.6.2 登録済み URL を変更する（Linux）.....	31

2.6.3 インストールを確認する.....	32
2.6.4 ライセンスを登録する.....	32
2.6.5 共通コンポーネントおよび Ops Center Automator のサービスを停止および開始する.....	32
(1) 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)	32
(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 (Linux)	32
(3) コマンドプロンプトから Ops Center Automator サービスのみ停止および開始する (Windows)	33
(4) コマンドプロンプトから Ops Center Automator サービスのみ停止および開始する (Linux)	33
2.7 Common Services にシングルサインオンを構成する.....	33
2.7.1 Ops Center Automator を Common Services に登録する.....	33
2.7.2 setupcommonservice コマンド.....	34
3.Ops Center Automator を構成する.....	37
3.1 管理サーバのシステム設定を変更する.....	38
3.1.1 管理サーバと管理クライアントとの通信に使用されるポート番号を変更する.....	38
3.1.2 ポート番号を変更した場合に共通コンポーネントのプロパティを更新する.....	39
3.1.3 管理サーバのホスト名を変更する.....	41
3.1.4 管理サーバの IP アドレスを変更する.....	41
3.1.5 管理サーバの URL を変更する.....	42
3.2 セキュア通信を構成する.....	43
3.2.1 Common Services とのセキュアな通信サービスについて.....	43
3.2.2 Ops Center Automator のセキュリティ設定について.....	43
3.2.3 管理クライアントのセキュリティを構成する.....	44
(1) 管理クライアントのセキュア通信について.....	44
(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)	44
(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)	48
(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.....	52
3.2.4 Web サービス接続の証明書をインポートする.....	53
3.2.5 ESX クラスタサービスの VMware サーバ証明書をインストールする (Windows)	54
3.2.6 ESX クラスタサービスの VMware サーバ証明書をインストールする (Linux)	55
3.2.7 Ops Center Automator サーバと REST API サーバの間で SSL 通信を使用するための設定を指定する (認証局によるサーバ証明書を使っている場合)	56
3.2.8 サーバ証明書の有効期限を確認する.....	56
3.3 監査ログ.....	56
3.3.1 監査ログを設定する.....	56
3.3.2 監査ログを有効にする.....	58
3.3.3 auditlog.conf ファイルの設定.....	58
3.3.4 auditlog.conf ファイルのサンプル.....	60
3.3.5 監査ログに出力されるデータのフォーマット.....	60
3.4 別のホストへ Ops Center Automator を移動する.....	62
3.5 Common Services を別のホストに移動する.....	63
3.6 システム構成を変更する.....	63
3.7 パフォーマンスモードを設定する.....	70
3.8 メール通知を構成する.....	71
3.9 エージェントレス接続の対応 OS.....	73
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	74
3.11 エージェントレス接続の Windows 前提条件.....	77
3.12 エージェントレス接続の SSH 前提条件.....	79
3.12.1 パスワード認証.....	79

3.12.2 公開鍵認証.....	79
3.12.3 キーボードインタラクティブ認証.....	81
3.13 Configuration Manager サーバで Java ヒープメモリサイズを設定する.....	81
4.外部認証サーバでのユーザー管理.....	83
4.1 外部認証サーバでのユーザー管理.....	84
5.Ops Center Automator をアンインストールする.....	85
5.1 Ops Center Automator をアンインストールする（Windows）	86
5.2 クラスタ環境で Ops Center Automator をアンインストールする.....	86
5.3 Ops Center Automator をアンインストールする（Linux）	88
付録 A Ops Center Automator のファイルの場所とポート.....	89
A.1 Ops Center Automator のファイルの場所.....	90
A.2 ポート設定.....	91
付録 B Ops Center Automator のプロセス.....	95
B.1 プロセス一覧（Windows）	96
B.2 プロセス一覧（Linux）	96
付録 C トラブルシューティング.....	97
C.1 保守情報を収集する.....	98
C.2 ログファイルを収集する.....	98
索引.....	99



はじめに

このマニュアルでは、Hitachi Ops Center Automator 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
- このマニュアルでの表記

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

Ops Center Automator の概要について説明しています。

第2章 Ops Center Automator 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)/Oracle Linux での、Ops Center Automator のインストール方法について説明しています。

第3章 Ops Center Automator を構成する

Ops Center Automator を構成する方法について説明しています。

第4章 外部認証サーバでのユーザー管理

外部認証サーバでユーザー認証を設定する方法について説明しています。

第5章 Ops Center Automator を削除する

Ops Center Automator を削除する方法について説明しています。

付録A Ops Center Automator のファイルの場所とポート

Ops Center Automator のインストール時に作成されるファイルの場所およびポートについて説明しています。

付録B Ops Center Automator のプロセス

Ops Center Automator のプロセスについて説明しています。

付録C トラブルシューティング

Ops Center Automator サーバでエラーが発生した場合の対処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	Internet Explorer®
Windows	次の製品を区別する必要がある場合の表記です。 <ul style="list-style-type: none">• Microsoft® Windows Server® 2012• Microsoft® Windows Server® 2012 R2

表記	製品名
	<ul style="list-style-type: none"> • Microsoft® Windows Server® 2016 • Microsoft® Windows Server® 2019
Windows Server 2012	次の製品を区別する必要がある場合の表記です。 <ul style="list-style-type: none"> • Microsoft® Windows Server® 2012 • Microsoft® Windows Server® 2012 R2
Windows Server 2016	Microsoft® Windows Server® 2016
Windows Server 2019	Microsoft® Windows Server® 2019

関連マニュアル

このマニュアルの関連マニュアルを次に示します。必要に応じてお読みください。

- *Hitachi Ops Center Automator ユーザーズガイド*, 4010-1J-034
- *Hitachi Ops Center Automator Service Builder ユーザーズガイド*, 4010-1J-037
- *Hitachi Ops Center Automator メッセージ*, 4010-1J-038
- *Hitachi Ops Center インストールガイド*, 4010-1J-101
- *Hitachi Ops Center API Configuration Manager REST API リファレンスガイド*, 4010-1J-031

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	<ul style="list-style-type: none"> • 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： <pre>pairedisplay -g group</pre> （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： <pre>pairedisplay -g oradb</pre>
<>（山括弧）	次のような場合に、変数を示します。 <ul style="list-style-type: none"> • 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： <pre>Status-<report-name><file-version>.csv</pre> • 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。

規則	説明
(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

1KB（キロバイト）、1MB（メガバイト）、1GB（ギガバイト）、1TB（テラバイト）は、それぞれ 1KiB（キビバイト）、1MiB（メビバイト）、1GiB（ギビバイト）、1TiB（テビバイト）と読み替えてください。

1KiB、1MiB、1GiB、1TiB は、それぞれ 1,024 バイト、1,024KiB、1,024MiB、1,024GiB です。

このマニュアルでの表記

このマニュアルでは、製品の名称を省略して表記しています。このマニュアルでの表記と、製品の正式名称または意味を次に示します。

表記	製品名
Common Services	Hitachi Ops Center Common Services
Configuration Manager	Hitachi Ops Center API Configuration Manager
HUS VM	Hitachi Unified Storage VM
Linux	Ops Center Automator がサポートしている Red Hat Enterprise Linux [®] および Oracle Linux [®] の総称です。
Ops Center Automator (Automator)	Hitachi Ops Center Automator
Ops Center Portal	Hitachi Ops Center Portal
Ops Center Viewpoint	Hitachi Ops Center Viewpoint
Oracle Linux	Oracle Linux [®]
Red Hat Enterprise Linux (RHEL)	Red Hat Enterprise Linux [®]
Service Builder	Ops Center Automator Service Builder
Virtual Storage Platform	Hitachi Virtual Storage Platform
	Hitachi Virtual Storage Platform VP9500
VMware	VMware [®]
VMware ESX	VMware vSphere [®] ESXi [™]
VMware vCenter	VMware vCenter Server [™]
VMware vCenter Server	
VMware vSphere	VMware vSphere [®]
VSP 5000 シリーズ	Hitachi Virtual Storage Platform 5100
	Hitachi Virtual Storage Platform 5200
	Hitachi Virtual Storage Platform 5500
	Hitachi Virtual Storage Platform 5600

表記	製品名
	Hitachi Virtual Storage Platform 5100H
	Hitachi Virtual Storage Platform 5200H
	Hitachi Virtual Storage Platform 5500H
	Hitachi Virtual Storage Platform 5600H
VSP E シリーズ	Hitachi Virtual Storage Platform E590
	Hitachi Virtual Storage Platform E790
	Hitachi Virtual Storage Platform E990
	Hitachi Virtual Storage Platform E590H
	Hitachi Virtual Storage Platform E790H
VSP Fx00 モデル	Hitachi Virtual Storage Platform F350
	Hitachi Virtual Storage Platform F370
	Hitachi Virtual Storage Platform F400
	Hitachi Virtual Storage Platform F600
	Hitachi Virtual Storage Platform F700
	Hitachi Virtual Storage Platform F800
	Hitachi Virtual Storage Platform F900
VSP F1500	Hitachi Virtual Storage Platform F1500
VSP F シリーズ	VSP Fx00 モデル
	VSP F1500
VSP Gx00 モデル	Hitachi Virtual Storage Platform G100
	Hitachi Virtual Storage Platform G130
	Hitachi Virtual Storage Platform G150
	Hitachi Virtual Storage Platform G200
	Hitachi Virtual Storage Platform G350
	Hitachi Virtual Storage Platform G370
	Hitachi Virtual Storage Platform G400
	Hitachi Virtual Storage Platform G600
	Hitachi Virtual Storage Platform G700
	Hitachi Virtual Storage Platform G800
	Hitachi Virtual Storage Platform G900
VSP G1000	Hitachi Virtual Storage Platform G1000
VSP G1500	Hitachi Virtual Storage Platform G1500
VSP G シリーズ	VSP Gx00 モデル
	VSP G1000
	VSP G1500
VSP ファミリー	Virtual Storage Platform
	VSP 5000 シリーズ
	VSP E シリーズ
	VSP F シリーズ

表記	製品名
	VSP G シリーズ

概要

この章では、以下について説明します。

- 1.1 製品の概要
- 1.2 関連する Hitachi Ops Center 製品について
- 1.3 Ops Center Automator システム構成
- 1.4 Ops Center Automator のインストールと構成のワークフロー
- 1.5 Ops Center Automator での認証方法

1.1 製品の概要

Ops Center Automator 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、Ops Center Automator は既存の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。

Ops Center Automator 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 Ops Center 製品について

Hitachi Ops Center 製品は、下記製品で構成されます。

- Hitachi Ops Center Automator
- Hitachi Ops Center Viewpoint

Hitachi Ops Center 製品は以下のコンポーネントを内包しています。

- Hitachi Ops Center Common Services
- Hitachi Ops Center API Configuration Manager

Hitachi Ops Center 製品は共通の設定でユーザーとセキュリティを管理できます。

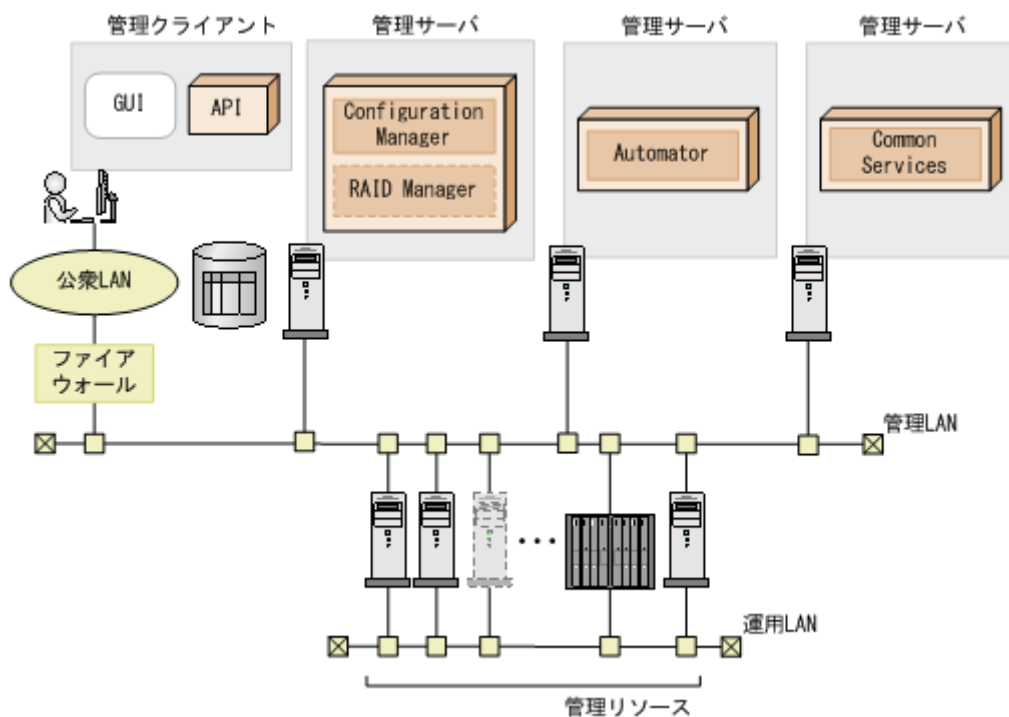
1.3 Ops Center Automator システム構成

Ops Center Automator は、Common Services および Configuration Manager を使用します。

Common Services は、ユーザー情報を一元管理して Ops Center Automator へリンク & ラUNCH するための Ops Center Portal を提供します。Configuration Manager は、ストレージシステムの情報取得や構成変更を行うための、REST の原則に従った Web API を提供します。

Ops Center Automator、Configuration Manager および Common Services は同じ管理サーバにインストールするか、または、別の管理サーバにインストールすることもできます。ただし、Common Services は Linux 版のみサポートされているため、Ops Center Automator を Windows サーバにインストールする場合には、Common Services をインストールするための Linux サーバが別途必要となります。

Ops Center Automator システム構成を次の図に示します



1.4 Ops Center Automator のインストールと構成のワークフロ

次の図は、Ops Center Automator 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。



このマニュアルには、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、および自動化の詳細については、『Hitachi Ops Center Automator ユーザーズガイド』を参照してください。

1.5 Ops Center Automator での認証方法

Ops Center Automator を運用するために、次の認証方法を使用できます。

- ・ 外部認証
- ・ ローカルユーザー認証

これらの認証方法は Common Services で管理され、Ops Center 製品間でのシングルサインオンを可能にします。



メモ Common Services では、外部の認証サーバと連携することで、ユーザー認証を ID プロバイダーで一元的行うこともできます。

Ops Center Automator 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®, および非クラスタ環境における Red Hat Enterprise Linux(RHEL)/Oracle Linux での、Ops Center Automator 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
- 2.2 Ops Center Automator をインストールする (Windows)
- 2.3 クラスタ環境で Ops Center Automator をインストールする (Windows)
- 2.4 Ops Center Automator をインストールする (Linux)
- 2.5 データベースフォルダのウイルススキャンを抑止する
- 2.6 インストール後のタスク
- 2.7 Common Services にシングルサインオンを構成する

2.1 インストールの前提条件

Ops Center Automator 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。

- 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、**Ops Center Automator** のリリースノートを参照してください。
- **Ops Center Automator** 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製品によって使用されておらず、競合していないことを確認します。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。
- 関連マシンの IP アドレスとホスト名の名前を解決します。
- このマニュアル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。
- このマニュアル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、**Linux** の **root** 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。
- サーバのシステム時刻が正しいことを確認します。**Ops Center** 製品が別のサーバにインストールされている場合は、**Ops Center Automator** サーバと当該サーバの時刻を同期させます。
- **RHEL/Oracle Linux** の場合、必要に応じて **Ops Center Automator** のファイアウォール例外を、手動で追加します。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。
- **RHEL/Oracle Linux** の場合、**/tmp** ディレクトリが **noexec** オプションを使用せずにマウントされていることを確認します。

関連参照

- [2.1.1 サーバ時刻を変更する](#)
- [2.1.2 名前解決設定を変更する](#)
- [付録 A.2 ポート設定](#)

2.1.1 サーバ時刻を変更する



重要 Ops Center Automator サーバの OS の時刻設定が、Ops Center 製品の管理サーバと同期している必要があります。



メモ Common Services と Ops Center Automator が異なるホスト上で稼働している場合、Common Services がインストールされているホストと、Ops Center Automator がインストールされているホストにタイムラグがあると、Ops Center Portal から Ops Center Automator を起動できません。Common Services のホストと Ops Center Automator のホストの時刻を同期させる必要があります。NTP を使用して両方のホストの時刻を同期させてください。

Ops Center Automator 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 OS 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、Ops Center Automator をインストールする前にリセットしてください。共通コンポーネントおよび Ops Center Automator のサービスが実行しているときに Ops Center Automator サーバの時刻を変更した場合、Ops Center Automator 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 **Ops Center Automator** 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 OS 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、**Ops Center Automator** 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. 共通コンポーネントおよび **Ops Center Automator** のサービスを停止します。
 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。

Ops Center Automator 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

Ops Center Automator にアクセスするためにブラウザを実行するマシンで、**Ops Center Automator** サーバ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、ホスト名が IP アドレス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

Ops Center Automator を新しくインストールする前に、管理サーバ上で **Ops Center Automator** 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** または **ss** コマンドを使用します。

ポート番号 **22170**～**22173** が他の製品によって使用されていないことを確認する必要があります。使用されている場合、新規インストールまたはアップグレードインストールが失敗するためです。

関連タスク

- [3.1.1 管理サーバと管理クライアントとの通信に使用されるポート番号を変更する](#)

関連参照

- ・ [付録 A.2 ポート設定](#)

2.2 Ops Center Automator をインストールする (Windows)

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 Ops Center Automator をインストールする方法を説明します。

ソフトウェアをアップグレードする場合は、**backupsystem** コマンドを使用して、既存のシステム構成とデータを必ずバックアップしてください。このコマンドの実行方法については、『*Hitachi Ops Center Automator ユーザーズガイド*』を参照してください。



メモ Ops Center Automator を、共通コンポーネントを使用するほかの製品 (Hitachi Command Suite 製品など) とともにインストールすることはできません。

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. インストールメディアを DVD ドライブに挿入します。
3. 以下のコマンドを実行して、インストールウィザードを起動します。
`<DVD ドライブ>:¥HAD_SERVER¥setup.exe`
4. 画面の指示に従って、必要な情報を指定します。
次のメッセージが表示された場合、Ops Center Automator のリリースノートを参照してください。

このサーバには、既に Hitachi Command Suite 製品がインストールされています。このインストールを続行すると、Hitachi Command Suite 製品のアップグレード (上書き (更新または修復) インストールを含む) ができなくなる可能性があります。一旦、インストールを中止しますか？

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。

[インストール完了] ウィンドウが開きます。

5. [完了] をクリックします。

操作結果

これで、Ops Center Automator がインストールされます。

関連参照

- ・ [2.6 インストール後のタスク](#)

2.3 クラスタ環境で Ops Center Automator をインストールする (Windows)

Windows クラスタ環境に Ops Center Automator をインストールします。



メモ Ops Center Automator は、Windows クラスタ環境だけをサポートします。Ops Center Automator は、Linux 環境でのクラスタリングをサポートしていません。

2.3.1 クラスタ環境での Ops Center Automator の使用について

Ops Center Automator 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。



メモ Ops Center Automator は、マルチサブネット構成のクラスタへのインストールはサポートしていません。

クラスタ環境で Ops Center Automator を使用するときには、次のように、1 台の Ops Center Automator 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。

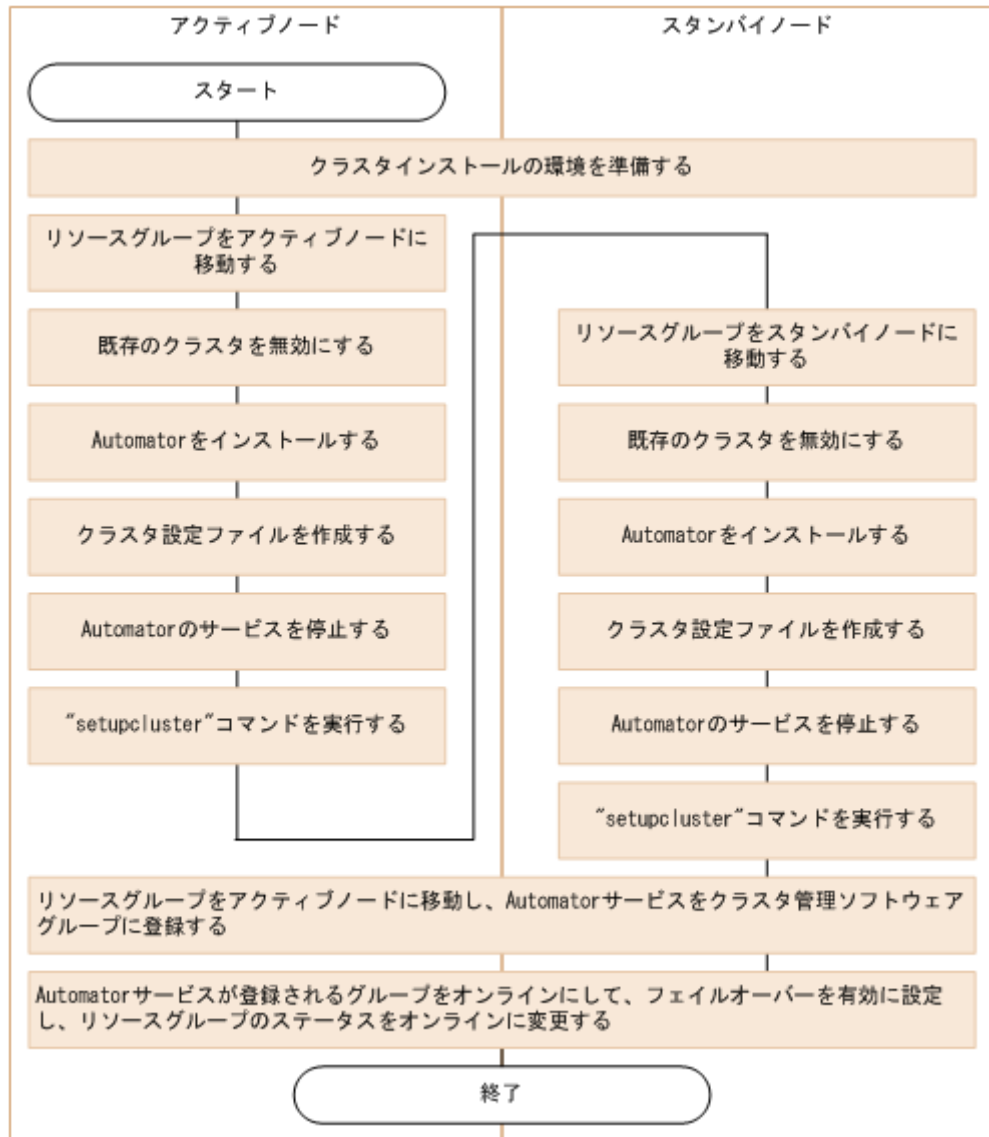


メモ アクティブノードがスタンバイノードにフェイルオーバーした場合、実行中のタスクは失敗するので、スタンバイノード上でタスクを再び実行する必要があります。

2.3.2 クラスタインストールワークフロー

Ops Center Automator 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、アクティブノードとスタンバイノードの両方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。



初めて Ops Center Automator 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を確認してください。

ソフトウェアをアップグレードする場合は、**backupsystem** コマンドを使用して、既存のシステム構成とデータを必ずバックアップしてください。このコマンドの実行方法については、『*Hitachi Ops Center Automator ユーザーズガイド*』を参照してください。



メモ 既にクラスタ構成でインストールされている Ops Center Automator のアップグレードを行うときには、アップグレード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 Ops Center Automator 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 Ops Center Automator 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確認する

クラスタ環境で Ops Center Automator 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 Ops Center Automator 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、以下の項目を確認します。

- クラスタ管理ソフトウェアを使用して、Ops Center Automator 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「ロール」など異なる名前と呼ばれることがあります。

2.3.4 アクティブノードで Ops Center Automator 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、Ops Center Automator の新規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. アクティブノード上で Ops Center Automator をインストールします。
インストールの前に、次の要件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。

インストール時に次のメッセージが表示された場合、Ops Center Automator のリリースノートを参照してください。

このサーバには、既に Hitachi Command Suite 製品がインストールされています。このインストールを続行すると、Hitachi Command Suite 製品のアップグレード(上書き(更新または修復)インストールを含む)ができなくなる可能性があります。一旦、インストールを中止しますか？

3. 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。
4. 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=<論理ホスト名>
onlinehost=<アクティブノードのホスト名>
standbyhost=<スタンバイノードのホスト名>
```



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<共通コンポーネントのインストールフォルダ>
¥conf に保存します。

5. 次のコマンドを使用して、Ops Center Automator のサービスを確実に停止します。
<共通コンポーネントのインストールフォルダ>¥bin¥hcms64srv /stop /server
AutomationWebService
6. **setupcluster /exportpath** コマンドを実行します。exportpath には、絶対または相対フ
ォルダパスを指定します。

関連タスク

- ・ [2.3.5 スタンバイノードで Ops Center Automator 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 Ops Center Automator クラスタ化をセッ 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 Ops Center Automator 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、Ops Center Automator 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 [移動] を選択してから、[ノードを選
択] または [このサービスまたはアプリケーションを別のノードに移動] を選択します。
2. スタンバイノード上で Ops Center Automator をインストールします。
インストールの前に、以下の要件を確認してください。

- ・ アクティブノードと同じ場所に Ops Center Automator をインストールする必要があります。
- ・ 管理サーバの IP アドレスとして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。

インストール時に次のメッセージが表示された場合、Ops Center Automator のリリースノート
を参照してください。

このサーバには、既に Hitachi Command Suite 製品がインストールされています。このイン
ストールを続行すると、Hitachi Command Suite 製品のアップグレード(上書き(更新ま
たは修復)インストールを含む)ができなくなる可能性があります。一旦、インストールを中止
しますか?

3. 使用する製品のライセンスを登録します。
4. 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=<論理ホスト名>  
onlinehost=<アクティブノードのホスト名>  
standbyhost=<スタンバイノードのホスト名>
```

ファイルを `cluster.conf` という名前で<共通コンポーネントのインストールフォルダ>
¥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。

5. 次のコマンドを使用して、Ops Center Automator のサービスを確実に停止します。
＜共通コンポーネントのインストールフォルダ＞¥bin¥hcnds64srv /stop /server
AutomationWebService
6. **setupcluster /exportpath** コマンドを実行します。exportpath には、絶対または相対フォルダパスを指定します。

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

Ops Center Automator 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、Ops Center Automator 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 [移動] を選択してから、[ノードを選択] または [このサービスまたはアプリケーションを別のノードに移動] 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 Ops Center Automator サービスを登録します。
＜共通コンポーネントのインストールフォルダ＞¥ClusterSetup
¥hcnds64clustersrvupdate /sreg /r <グループ名> /sd <共有ディスクのドライブレター名> /ap <クライアントアクセスポイント用リソース名>
 - /r
Ops Center Automator 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 Automator cluster の場合は、"Automator 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。Ops Center Automator 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 hcnds64clustersrvupdate 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. アクティブノードで、次のコマンドを使用して、Ops Center Automator の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。
＜共通コンポーネントのインストールフォルダ＞¥ClusterSetup
¥hcnds64clustersrvstate /son /r <グループ名>
r オプションには、Ops Center Automator 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 Automator cluster の場合は、"Automator cluster" と指定します。
4. 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 Ops Center Automator をインストールする (Linux)

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 Ops Center Automator をインストールする方法を説明します。

ソフトウェアをアップグレードする場合は、**backupsystem** コマンドを使用して、既存のシステム構成とデータを必ずバックアップしてください。このコマンドの実行方法については、『*Hitachi Ops Center Automator ユーザーズガイド*』を参照してください。



メモ Ops Center Automator を、共通コンポーネントを使用するほかの製品とともにインストールすることはできません。

操作手順

1. `install.sh` を実行して、Ops Center Automator をインストールします。

次のメッセージが表示された場合、Ops Center Automator のリリースノートを参照してください。

```
Hitachi Command Suite products are already installed on this server.  
If you continue the installation, you might not able to upgrade  
(including overwrite (modify or repair) installation) of Hitachi  
Command Suite products. Do you want to abort the installation once?
```

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

関連参照

- [2.6 インストール後のタスク](#)

2.5 データベースフォルダのウイルススキャンを抑止する

インストール時にウイルス検出プログラムが起動してデータベースフォルダをスキャンすると、インストールに時間が掛かる、または、失敗することがあります。

操作手順

1. インストール時のデータベースフォルダのウイルススキャンを抑止するために、以下のフォルダまたはディレクトリを、ウイルススキャンの対象外としてウイルススキャンプログラムに登録します。



メモ 以下のフォルダやディレクトリはデフォルトのパスであり、インストール時に変更できません。

- Windows の場合 :
`system-drive¥Program Files¥hitachi¥Automation`
`system-drive¥Program Files¥hitachi¥database`
`system-drive¥Program Files¥hitachi¥Base64¥HDB`
- Linux の場合 :
`/opt/hitachi/Automation`

```
/var/opt/hitachi/Automation  
/var/opt/hitachi/Base64/HDB  
/var/opt/hitachi/database
```

操作結果

登録されたフォルダやディレクトリは、インストール時のウイルススキャンの対象になりません。

2.6 インストール後のタスク

Ops Center Automator 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. 登録済み URL を変更します。詳細は、「[2.6.1 登録済み URL を変更する \(Windows\)](#)」または「[2.6.2 登録済み URL を変更する \(Linux\)](#)」を参照してください。
2. Ops Center Automator 管理サーバへのアクセスを確認します。
3. **setupcommonservice** コマンドを実行して、Common Services をセットアップします。このタスクは必ず実施してください。
setupcommonservice コマンドの詳細は、「[2.7.2 setupcommonservice コマンド](#)」を参照してください。



メモ クラスタ構成では、アクティブノードでのみ **setupcommonservice** を実行できます。

4. ライセンスを登録します。

2.6.1 登録済み URL を変更する (Windows)

Ops Center Automator のインストール後に、登録済み URL を変更します。

操作手順

1. 次のコマンドを使用して、登録済み URL を確認します。
＜共通コンポーネントのインストールフォルダ＞¥bin¥hcnds64chgurl /list
2. URL 内のホスト名を確認します。非クラスタ環境では、ホスト名は物理ホスト名でなければなりません。クラスタ環境では、ホスト名は論理ホスト名でなければなりません。
3. 次のコマンドを使用して、Ops Center Automator の登録済み URL を変更します。
＜共通コンポーネントのインストールフォルダ＞¥bin¥hcnds64chgurl /change
https://＜Ops-Center-Automator 管理サーバの IP アドレスまたはホスト名＞:22016 /type Automation

2.6.2 登録済み URL を変更する (Linux)

Ops Center Automator のインストール後に、登録済み URL を変更します。

操作手順

1. 次のコマンドを使用して、登録済み URL 内のホスト名を確認します。
＜共通コンポーネントのインストールディレクトリ＞/bin/hcmds64chgurl -list
2. 次のコマンドを使用して、Ops Center Automator の登録済み URL を変更します。

```
<共通コンポーネントのインストールディレクトリ>/bin/hcmds64chgurl -change  
https://<Ops-Center-Automator 管理サーバの IP アドレスまたはホスト名>:22016  
-type Automation
```

2.6.3 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. Ops Center Automator 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、Ops Center Automator の URL を次の形式で指定します。
`https://<Ops-Center-Automator 管理サーバの IP アドレスまたはホスト名>:22016/
Automation/`

操作結果

管理サーバにアクセスできることを確認するログインウィンドウが開きます。

2.6.4 ライセンスを登録する

最初にログインするときには、有効なライセンスキーを指定する必要があります。



メモ Ops Center Automator のライセンスについては、サポートサービスにお問い合わせください。

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、[ファイルを選択] をクリックして、ライセンスファイルを参照します。
3. [保存] をクリックします。

2.6.5 共通コンポーネントおよび Ops Center Automator のサービスを停止および開始する

共通コンポーネントおよび Ops Center Automator はコマンドプロンプトからサービスを実行できます。

(1) 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、共通コンポーネントおよび Ops Center Automator 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、<共通コンポーネントのインストールフォルダ>%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`

(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 (Linux)

次の手順により、共通コンポーネントおよび Ops Center Automator 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、<共通コンポーネントのインストールディレクトリ>/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。
`hcnds64srv -stop`
サービスを開始するには、次のコマンドを入力します。
`hcnds64srv -start`

(3) コマンドプロンプトから Ops Center Automator サービスのみ停止および開始する (Windows)

操作手順

1. <共通コンポーネントのインストールフォルダ>%bin に移動します。
2. サービスを停止または開始します。
 - ・ サービスを停止するには、次のコマンドを入力します。
`hcnds64srv /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。
`hcnds64srv /start /server AutomationWebService`

(4) コマンドプロンプトから Ops Center Automator サービスのみ停止および開始する (Linux)

操作手順

1. <共通コンポーネントのインストールディレクトリ>/bin に移動します。
2. サービスを停止または開始します。
 - ・ サービスを停止するには、次のコマンドを入力します。
`hcnds64srv -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。
`hcnds64srv -start -server AutomationWebService`

2.7 Common Services にシングルサインオンを構成する

Ops Center Portal のシングルサインオン機能を利用するには、Ops Center Automator を Common Services に登録する必要があります。Common Services のホスト名、IP アドレス、またはサーバポート番号を変更した場合は、Ops Center Automator を再登録する必要があります。

2.7.1 Ops Center Automator を Common Services に登録する

Ops Center Automator サーバ上でコマンドを実行して、Ops Center Automator を Common Services に登録する必要があります。

操作手順

1. **auto** オプションを指定して **setupcommonservice** コマンドを実行し、Ops Center Automator を Common Services に登録します。
setupcommonservice コマンドの詳細については、「[2.7.2 setupcommonservice コマンド](#)」を参照してください

2.7.2 setupcommonservice コマンド

setupcommonservice コマンドは、Common Services と連携するための設定コマンドです。
setupcommonservice コマンドは、Ops Center Automator を Common Services にアプリケーションとして登録し、Ops Center Automator を Common Services を使用する認証サーバとして設定します。

機能

setupcommonservice コマンドは、Ops Center Automator の URL を Common Services に登録します。登録される URL は、**hcmds64chgurl** コマンドに登録された URL を使用します。

hcmds64chgurl に登録されている URL がブラウザで解決できることをあらかじめ確認してから、**setupcommonservice** コマンドを実行してください。

このコマンドには、Common Services と Ops Center Automator との間にセキュアな接続が必要です。

構文

構文 (Windows) :

```
setupcommonservice {[/csUri CommonServiceUri | /csUri CommonServiceUri /  
csUsername CommonServiceUsername] [/appName ApplicationName]  
[/appDescription ApplicationDescription] [ /auto ] | /help }
```

構文 (Linux) :

```
setupcommonservice {[-csUri CommonServiceUri | -csUri CommonServiceUri -  
csUsername CommonServiceUsername] [-appName ApplicationName]  
[-appDescription ApplicationDescription] [ -auto ] | -help }
```



メモ 対話モードでのパスワードの入力を求められます。

オプション

オプション	説明
csUri	Common Services の URL を指定します。(例 : https://common.service/portal)
csUsername	Common Services で管理される、opscenter-security-administrator ロールを持つユーザーを指定します。ユーザー名には、1 バイトの英数字を使用できます。これには、(!# \$% & '() * +. = @ ^ _)が含まれます。長さは 1～255 文字です。ユーザー名の大文字と小文字は区別されます。 このオプションを指定してコマンドを実行する場合は、パスワードの入力を求められます。
appName	Common Services で表示される Ops Center Automator の名前を指定します。名前は、1～128 文字で指定できます。 新規登録時に appName を省略すると、Ops Center Automator のホスト名または IP アドレスが名前として設定されます。更新時に appName を省略すると、名前は変更されません。
appDescription	Common Services で表示される、Ops Center Automator の説明を指定します。説明は、0～512 文字で指定できます。

オプション	説明
auto	Ops Center Automator のサービスおよびデータベースを自動で開始および停止します。

関連概念

- [3.2.1 Common Services とのセキュアな通信サービスについて](#)

Ops Center Automator を構成する

この章では、Ops Center Automator 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
- 3.2 セキュア通信を構成する
- 3.3 監査ログ
- 3.4 別のホストへ Ops Center Automator を移動する
- 3.5 Common Services を別のホストに移動する
- 3.6 システム構成を変更する
- 3.7 パフォーマンスモードを設定する
- 3.8 メール通知を構成する
- 3.9 エージェントレス接続の対応 OS
- 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
- 3.11 エージェントレス接続の Windows 前提条件
- 3.12 エージェントレス接続の SSH 前提条件
- 3.13 Configuration Manager サーバで Java ヒープメモリサイズを設定する

3.1 管理サーバのシステム設定を変更する

ここでは、Ops Center Automator 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 管理サーバと管理クライアントとの通信に使用されるポート番号を変更する

Ops Center Automator 管理サーバと管理クライアント（Web ブラウザ）間の通信に使用されるポート番号を変更するには、定義ファイルの編集と、ファイアウォールの例外登録が必要になります。クラスタシステムの場合、実行系サーバと待機系サーバで同じ手順を実施してください。



メモ Ops Center Automator に使用される他のポートの情報については、ポート設定の参考トピックを参照してください。

Ops Center Automator 管理サーバと管理クライアント間のポート番号を変更するには：

操作手順

1. Ops Center Automator を停止します。
2. 定義ファイルのキーを編集してポート番号の設定を変更します。
 - HTTPS の場合、手順 3 に進みます。
 - HTTP の場合、次のように定義ファイルのキーを編集してポート番号の設定を変更します。
 - a. `user_httpsd.conf` ファイルの `Listen` キーの行を変更します。

Windows の場合：
<共通コンポーネントのインストールフォルダ>%uCPsB11%httpsd%conf
%user_httpsd.conf

Linux の場合：
<共通コンポーネントのインストールディレクトリ>/uCPsB11/httpsd/conf/
user_httpsd.conf

次の行で、22015 に替わる新しいポート番号を指定します。

```
Listen [::]:22015
Listen 22015
#Listen 127.0.0.1:22015
```
 - b. `command_user.properties` ファイルの `command.http.port` の行を変更します。

クラスタシステムの場合、この定義ファイルは別のフォルダに含まれています。

Windows（非クラスタ環境）の場合：
<Ops-Center-Automator のインストールフォルダ>%conf

Windows（クラスタ環境）の場合：
<共有フォルダ名>%Automation%conf

Linux の場合：
/opt/hitachi/Automation/conf
 - c. `config_user.properties` ファイルの `server.http.port` の行を変更します。

クラスタシステムの場合、この定義ファイルは別のフォルダに含まれています。

Windows（非クラスタ環境）の場合：
<Ops-Center-Automator のインストールフォルダ>%conf

Windows（クラスタ環境）の場合：
<共有フォルダ名>%Automation%conf

Linux の場合：
/opt/hitachi/Automation/conf

- d. 手順 4 に進みます。
3. HTTPS の場合、次のように定義ファイルのキーを編集してポート番号の設定を変更します。
 - a. user_httpsd.conf ファイルを開きます。
 Windows の場合 :
 <共通コンポーネントのインストールフォルダ>%uCP SB11%httpsd%conf%user_httpsd.conf
 Linux の場合 :
 <共通コンポーネントのインストールディレクトリ>/uCP SB11/httpsd/conf/user_httpsd.conf
 - b. 次の行で 22016 に替わる新しいポート番号を指定して、Listen キーの行を変更します。

```
Listen [::]:22016
```

```
Listen 22016
```

```
VirtualHost *22016
```
4. ファイアウォールの例外登録をします。
 - OS が Windows の場合は、**hcmds64fwcancel** コマンドを実行してファイアウォールの例外登録をします。
 - OS が Linux の場合は、OS の仕様に従って例外登録をします。手順については、OS のマニュアルを参照してください。
5. Ops Center Automator を開始します。
6. **hcmds64chgurl** コマンドを実行して、Ops Center Automator の URL を更新します。
7. **setupcommonservice** コマンドを実行して、Common Services を設定します。

関連概念

- [2.6.5 共通コンポーネントおよび Ops Center Automator のサービスを停止および開始する](#)

関連参照

- [2.7.2 setupcommonservice コマンド](#)
- [3.1.2 ポート番号を変更した場合に共通コンポーネントのプロパティを更新する](#)
- [付録 A.2 ポート設定](#)

3.1.2 ポート番号を変更した場合に共通コンポーネントのプロパティを更新する

Ops Center Automator のポート番号を変更する場合は、共通コンポーネントのプロパティを更新する必要があります。プロパティを更新後、Ops Center Automator サーバを再起動する必要があります。

操作手順

1. 次の表に示されている共通コンポーネントのプロパティを更新します。

ポート番号 (デフォルト)	プロパティファイルのパス (<共通コンポーネントのインストールフォルダ>配下)	更新場所
22015/TCP	%uCP SB11%httpsd%conf%user_httpsd.conf	Listen
		Listen [::]:
		#Listen 127.0.0.1:
22016/TCP	%uCP SB11%httpsd%conf%user_httpsd.conf	VirtualHost タグの host-name :port-number

ポート番号 (デフォルト)	プロパティファイルのパス (<共通コンポーネントのインストールフォルダ>配下)	更新場所
		Listen
		Listen [::]:
22031/TCP	%CPSB11%\httpsd\conf %user_hssso_httpsd.conf	Listen
22032/TCP	%HDB%\CONF%\emb%\HiRDB.ini	PDNAMEPORT
	%HDB%\CONF%\pdsys	pd_name_port
	%database%\work%\def_pdsys	pd_name_port
22035/TCP	%CPSB11%\CC%\server%\usrconf%\ejb %HBase64StgMgmtSSOService %usrconf.properties	webserver.connector.nio_http.port
	%CPSB11%\httpsd\conf フォルダ配下の次のファイル <ul style="list-style-type: none"> reverse_proxy.conf reverse_proxy_before.conf reverse_proxy_after.conf hssso_reverse_proxy.conf 	webserver.connector.nio_http.port
22036/TCP	%CPSB11%\CC%\server%\usrconf%\ejb %HBase64StgMgmtSSOService %usrconf.properties	ejbserver.rmi.naming.port
22037/TCP	%CPSB11%\CC%\server%\usrconf%\ejb %HBase64StgMgmtSSOService %usrconf.properties	ejbserver.http.port
22038/TCP	%CPSB11%\CC%\server%\usrconf%\ejb %HBase64StgMgmtSSOService %usrconf.properties	ejbserver.rmi.remote.listener.port
22170/TCP	%CPSB11%\CC%\server%\userconf%\ejb %AutomationWebService %usrconf.properties	webserver.connector.nio_http.port
	%CPSB11%\httpsd\conf フォルダ配下の次のファイル <ul style="list-style-type: none"> reverse_proxy.conf reverse_proxy_before.conf reverse_proxy_after.conf hssso_reverse_proxy.conf 	webserver.connector.nio_http.port
22171/TCP	%CPSB11%\CC%\server%\userconf%\ejb %AutomationWebService %usrconf.properties	ejbserver.rmi.naming.port
22172/TCP	%CPSB11%\CC%\server%\userconf%\ejb %AutomationWebService %usrconf.properties	ejbserver.http.port

ポート番号（デフォルト）	プロパティファイルのパス（＜共通コンポーネントのインストールフォルダ＞配下）	更新場所
22173/TCP	¥uCPSB11¥CC¥server¥userconf¥ejb ¥AutomationWebService ¥usrconf.properties	ejbserver.rmi.remote.listener.port

- Ops Center Automator サービスを再起動します。

3.1.3 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、Ops Center Automator 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

- 新しい管理サーバのホスト名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。
- `hcnds64srv /stop` コマンドを実行して、共通コンポーネントおよび Ops Center Automator のサービスを停止します。
- `user_httpsd.conf` ファイルを編集します。

`user_httpsd.conf` ファイルは、次の場所に格納されています。

Windows の場合：

＜共通コンポーネントのインストールフォルダ＞¥uCPSB11¥httpsd¥conf

Linux の場合：

＜共通コンポーネントのインストールディレクトリ＞/uCPSB11/httpsd/conf

ServerName 行の値を新しいホスト名に変更します。

```
ServerName <管理サーバのホスト名>
```

SSL 設定が有効の場合、サーバ証明書を再取得し、VirtualHost ディレクティブの ServerName 行の値を新しいホスト名に変更します。

```
<VirtualHost *:22016>
```

```
ServerName <管理サーバのホスト名>
```

- `command_user.properties` ファイルの `command_hostname` 行の値を新しいホスト名に変更します。
`command_user.properties` ファイルは、次の場所に格納されています。
Windows（非クラスタ環境）の場合：
＜Ops-Center-Automator のインストールフォルダ＞¥conf
Windows（クラスタ環境）の場合：
＜共有フォルダ＞¥Automation¥conf
Linux の場合：
＜Ops-Center-Automator のインストールディレクトリ＞/conf
- 管理サーバのホスト名を変更します。変更後、サーバを再起動します。
- Ops Center Automator の URL にホスト名を使用している場合、`hcnds64chgurl` コマンドを実行して、URL を更新します。

3.1.4 管理サーバの IP アドレスを変更する

管理サーバの IP アドレスは、Ops Center Automator のインストール後に変更できます。

操作手順

1. [タスク] タブで、タスクの状態を確認します。
稼働中のタスク（実行中、応答待ち中、長期実行中、異常検出、または停止中）がある場合、タスクを停止する、またはタスクの状態が実行完了（完了、失敗、またはキャンセル）に変わるまで待機します。
2. `hcmds64srv /stop` コマンドを実行して、共通コンポーネントおよび Ops Center Automator のサービスを停止します。
3. 管理サーバの IP アドレスを変更します。
4. `hcmds64srv /start` コマンドを実行して、共通コンポーネントおよび Ops Center Automator のサービスを開始します。
5. Ops Center Automator の URL に IP アドレスを使用している場合、`hcmds64chgurl` コマンドを実行して、URL を更新します。

3.1.5 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、Ops Center Automator のポート、または SSL 設定を変更した場合は、Ops Center Automator 管理サーバの URL を変更する必要があります。



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。
`<共通コンポーネントのインストールフォルダ>%bin%hcmds64chgurl /list`
2. Ops Center Automator 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 Ops Center Automator の URL だけを変更します。
`<共通コンポーネントのインストールフォルダ>%bin%hcmds64chgurl /change <変更後の URL > /type Automation`
URL には次の形式を使用します。
`<プロトコル>://<管理サーバの IP アドレスまたはホスト名>:<ポート番号>`
 - ・ <プロトコル>は、非 SSL 通信の場合は `http`、SSL 通信の場合は `https` です。
 - ・ <管理サーバの IP アドレスまたはホスト名>は、Ops Center Automator 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。
 - ・ <ポート番号>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、<共通コンポーネントのインストールフォルダ>
`%uCPsB11%httpsd%conf` にあります。
3. 新しい URL を使用して Ops Center Automator にアクセスできることを確認します。
4. `setupcommonservice` コマンドを実行して、Common Services に変更を適用します。

3.2 セキュア通信を構成する

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



メモ Common Services との通信にまだ SSL をセットアップしていない場合は、「[3.2.1 Common Services とのセキュアな通信サービスについて](#)」を参照してください。

3.2.1 Common Services とのセキュアな通信サービスについて

Ops Center Automator および Common Services は、SSL 接続で通信する必要があります。Common Services を利用するには、セキュアな通信を構成しなければなりません。詳細については、「[\(2\) 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 \(Windows\)](#)」または「[\(3\) 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 \(Linux\)](#)」を参照してください。

デフォルトでは、SSL の検証が有効になっていません。SSL の検証を有効にするには、<Ops-Center-Automator のインストールディレクトリ>/conf/config_user.properties ファイルで、sso.https.certification パラメータを「true」に設定してください。

Common Services が、Ops Center Automator と同じサーバ上にある場合は、**cssslsetup** コマンドを利用できます。

cssslsetup コマンドを利用することで、共通の秘密鍵とサーバ証明書を使用して、同じ管理サーバにインストールされている Ops Center 製品に、SSL 通信を構成できます。

cssslsetup コマンドの利用方法と対応範囲については、『Hitachi Ops Center インストールガイド』を参照してください。

3.2.2 Ops Center Automator のセキュリティ設定について

Ops Center Automator 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、Ops Center Automator は Ops Center Automator 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、Ops Center Automator 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。

Ops Center Automator 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 Ops Center Automator 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 Ops Center Automator に登録しないとサーバに接続できません。

3.2.3 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 Ops Center Automator 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。

(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。



メモ 新規インストール後、SSL 設定が有効になります。オプションなしで `hcmds64ssltool` コマンドを実行するときと同じ証明書が使用されます。アップグレードインストールの場合、現在の SSL 設定を保持します。

`hcmds64ssltool` コマンドは、2 種類の秘密鍵、RSA 暗号と ECC（楕円曲線暗号）に対応する証明書署名要求および自己署名証明書を作成します。証明書署名要求は、PEM 形式で作成されます。このコマンドは自己署名証明書の作成にも使用できますが、自己署名証明書は、テスト目的にだけ使用する必要があります。

前提条件

Administrator 権限を持つユーザーとしてログインします。

次の情報を収集します。

- 認証局が指定する証明書署名要求の要件
- 管理クライアントで実行している Web ブラウザのバージョン
Web ブラウザは、X.509 PEM 形式を使用しており、管理クライアント (GUI) で使用されているサーバ証明書の署名アルゴリズムをサポートしている必要があります。
- 既存の秘密鍵、証明書署名要求、および自己署名証明書の保存先フォルダ（再作成する場合）
出力先パスに同じ名前のファイルが既に存在する場合、ファイルを上書きしません。したがって、秘密鍵、証明書署名要求、および自己署名証明書を再作成する場合、既存の保存先フォルダ以外のフォルダに出力するか、既存のファイルを削除する必要があります。

操作手順

1. 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。
`<共通コンポーネントのインストールフォルダ>%bin%hcmds64ssltool [/key <秘密鍵ファイル>] [/csr <証明書発行要求ファイル>] [/cert <自己署名証明書ファイル>] [/certtext <自己署名証明書の内容ファイル>] [/validity <有効日数>] [/sigalg <RSA 暗号用のサーバ証明書の署名アルゴリズム>] [/eccsigalg <ECC 用のサーバ証明書の署名アルゴリズム>] [/ecckeysize <ECC 用の秘密鍵のキーサイズ>] [/ext <X.509 証明書の拡張情報>]`
 - /key

作成された秘密鍵ファイルの出力先を絶対パスで指定します。このオプションを省略すると、ファイルは、RSA 暗号の場合は `httpsdkey.pem`、ECC の場合は `ecc-httpsdkey.pem` というファイル名で、デフォルトの出力先パス※に出力されます。

- `/csr`

作成された証明書発行要求ファイルの出力先を絶対パスで指定します。このオプションを省略すると、ファイルは、RSA 暗号の場合は `httpsd.csr`、ECC の場合は `ecc-httpsd.csr` というファイル名で、デフォルトの出力先パス※に出力されます。

- `/cert`

作成された自己署名証明書の出力先を絶対パスで指定します。このオプションを省略すると、ファイルは、RSA 暗号の場合は `httpsd.pem`、ECC の場合は `ecc-httpsd.pem` というファイル名で、デフォルトの出力先パス※に出力されます。

- `/certtext`

作成された自己署名証明書の内容ファイルの出力先を絶対パスで指定します。このオプションを省略すると、ファイルは、RSA 暗号の場合は `httpsd.txt`、ECC の場合は `ecc-httpsd.txt` というファイル名で、デフォルトの出力先パス※に出力されます。

- `/validity`

日数で自己署名証明書の有効期限を指定します。このオプションを省略すると、デフォルトの 3,650 日が使用されます。

- `/sigalg`

RSA 暗号用のサーバ証明書の署名アルゴリズムを `SHA256withRSA` または `SHA1withRSA` で指定します。このオプションを省略すると、デフォルトの `SHA256withRSA` が使用されます。

- `/eccsigalg`

ECC 用のサーバ証明書の署名アルゴリズムを `SHA512withECDSA`、`SHA384withECDSA`、`SHA256withECDSA`、または `SHA1withECDSA` で指定します。このオプションを省略すると、デフォルトの `SHA384withECDSA` が使用されます。

- `/ecckeysize`

ECC 用のサーバ証明書の秘密鍵のサイズを 256 または 384 ビットで指定します。このオプションを省略すると、デフォルトの 384 が使用されます。

- `/ext`

X.509 証明書の拡張情報を指定します。自己署名証明書および証明書署名要求に SAN (Subject Alternative Name) を設定する場合は、このオプションを指定します。指定方法は、Java の `keytool` コマンドの `ext` オプションに基づきます。Ops Center Automator で指定できる拡張情報は SAN だけであることに注意してください。ext オプションを複数回指定した場合は、最初の指定が有効になります。

以下に、拡張情報を指定する例を示します。

- `www.example.com` をホスト名として指定する場合：

```
hcmds64ssltool /ext san=dns:www.example.com
```

- `www.example.com` と `www.example.net` を複数のホスト名として指定する場合：

```
hcmds64ssltool /ext san=dns:www.example.com, dns:www.example.net
```

このコマンドは、RSA ファイルおよび ECC ファイルを指定した出力先パスに出力します。

RSA ファイルは、指定したファイル名で、ECC ファイルは、指定したファイル名の先頭に「ecc-」が付いて出力されます。

注※ key、csr、cert、または certtext オプションを省略した場合のデフォルトの出力先は、次のとおりです。

<共通コンポーネントのインストールフォルダ>%uCP%SB11%httpsd%conf%ssl%server

2. 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 (管理サーバのホスト名) - 例: Automator_SC1
- ・ 組織単位 (セクション) - 例: Ops Center Automator
- ・ 組織名 (会社) - 例: Hitachi
- ・ 都市または地区名 - 例: Yokohama
- ・ 州または県名 (フルネーム) - 例: Kanagawa
- ・ 国名 (2 文字のコード) - 例: JP

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。

3. 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、本番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。

4. Ops Center Automator を停止します。

5. 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、次のフォルダにコピーします。

<共通コンポーネントのインストールフォルダ>%uCPsB11¥httpsd¥conf¥ssl¥server

6. 次の場所から user_httpsd.conf ファイルを開きます。

<共通コンポーネントのインストールフォルダ>%uCPsB11¥httpsd¥conf¥user_httpsd.conf

7. user_httpsd.conf ファイル内で、以下のようになります。

a. 番号記号 (#) 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

から

#HWSLogSSLVerbose On

ただし、#SSLCACertificateFile と #Header set Strict-Transport-Security max-age=31536000 はコメントアウトしたままにしておく必要があります。

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

以下に、user_httpsd.conf ファイルの編集例を示します。

```
ServerName <管理サーバのホスト名>
Listen [::]:22015
Listen 22015
#Listen 127.0.0.1:22015
SSLEngine Off
#Listen [::]:22016
Listen 22016
<VirtualHost *:22016>
ServerName <管理サーバのホスト名>
SSLEngine On
SSLProtocol TLSv1.2
# SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-
GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-
SHA256:AES256-GCM-SHA384:AES128-GCM-SHA256
SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-
SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256
SSLCertificateKeyFile
"<共通コンポーネントのインストールフォルダ>%uCPsB11¥httpsd¥conf¥ssl¥server
¥httpsdkey.pem"
SSLCertificateFile
```

```
"<共通コンポーネントのインストールフォルダ>%uCPsB11%httpsd%conf%ssl%server
%httpsd.pem"
SSLCertificateKeyFile
"<共通コンポーネントのインストールフォルダ>%uCPsB11%httpsd%conf%ssl%server
%ecc-httpsdkey.pem"
SSLCertificateFile
"<共通コンポーネントのインストールフォルダ>%uCPsB11%httpsd%conf%ssl%server
%ecc-httpsd.pem"
# SSLCACertificateFile
"<共通コンポーネントのインストールフォルダ>%uCPsB11%httpsd%conf%ssl%cacert
%anycert.pem"
# Header set Strict-Transport-Security max-age=31536000
</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

#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。

ディレクティブを編集する場合、以下について注意してください：

- 同じディレクティブを 2 回指定しないでください。ただし、**SSLCertificateKeyFile** および **SSLCertificateFile** ディレクティブは、RSA 暗号用と ECC 用で 2 回、指定できます。
- ディレクティブの途中で改行を入れないでください。
- 以下に示すディレクティブでパスを指定する場合、シンボリックリンクまたはジャンクションポイントを指定しないでください。
- 以下に示すディレクティブで証明書および秘密鍵ファイルを指定する場合、**PEM** 形式のファイルを指定してください。
- httpsd.conf ファイルおよび hssso_httpsd.conf ファイルを編集しないでください。
- 次の行の番号記号 (#) は削除しないでください。

```
# Header set Strict-Transport-Security max-age=31536000
```

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 <管理サーバのホスト名>
Listen [::]:22015
Listen 22015
#Listen 127.0.0.1:22015
SSLEngine Off
#Listen [::]:22016
Listen 22016
<VirtualHost *:22016>
ServerName <管理サーバのホスト名>
SSLEngine On
SSLProtocol TLSv1.2
```



```
# SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-
SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:AES256-
GCM-SHA384:AES128-GCM-SHA256
SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-
SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256
SSLCertificateKeyFile
"<共通コンポーネントのインストールフォルダ>%uCP SB11%httpsd%conf%ssl%server
%httpsdkey.pem"
SSLCertificateFile
"<共通コンポーネントのインストールフォルダ>%uCP SB11%httpsd%conf%ssl%server
%server-certificate-or-self-signed-certificate-file"
SSLCertificateKeyFile
"<共通コンポーネントのインストールフォルダ>%uCP SB11%httpsd%conf%ssl%server
%ecc-httpsdkey.pem"
SSLCertificateFile
"<共通コンポーネントのインストールフォルダ>%uCP SB11%httpsd%conf%ssl%server
%ecc-httpsd.pem"
SSLCACertificateFile
"<共通コンポーネントのインストールフォルダ>%uCP SB11%httpsd%conf%ssl%cacert
%certificate-file-from-certificate-authority"
# Header set Strict-Transport-Security max-age=31536000
</VirtualHost>
HWSLogSSLVerbose On
```

8. Ops Center Automator を開始します。
9. `hcmds64chgurl` コマンドを使用して、次のように Ops Center Automator 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、Ops Center Automator 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。



メモ 新規インストール後、SSL 設定が有効になります。オプションなしで `hcmds64ssltool` コマンドを実行するときと同じ証明書が使用されます。アップグレードインストールの場合、現在の SSL 設定を保持します。

`hcmds64ssltool` コマンドは、2 種類の秘密鍵、RSA 暗号と ECC（楕円曲線暗号）に対応する証明書署名要求および自己署名証明書を作成します。証明書署名要求は、PEM 形式で作成されます。このコマンドは自己署名証明書の作成にも使用できますが、自己署名証明書は、テスト目的にだけ使用することをお勧めします。

前提条件

root ユーザーとしてログインします。

次の情報を収集します。

- ・ 認証局が指定する証明書署名要求の要件
- ・ 管理クライアントで実行している Web ブラウザのバージョン
Web ブラウザは、X.509 PEM 形式を使用しており、管理クライアント (GUI) で使用されているサーバ証明書の署名アルゴリズムをサポートしている必要があります。

- 既存の秘密鍵、証明書署名要求、および自己署名証明書の保存先ディレクトリ（再作成する場合）
出力先パスに同じ名前のファイルが既に存在する場合、ファイルを上書きしません。したがって、秘密鍵、証明書署名要求、および自己署名証明書を再作成する場合、既存の保存先ディレクトリ以外のディレクトリに出力するか、既存のファイルを削除する必要があります。

操作手順

1. 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。

```
<共通コンポーネントのインストールディレクトリ>/bin/hcmds64ssltool [-key <秘密鍵ファイル>] [-csr <証明書発行要求ファイル>] [-cert <自己署名証明書ファイル>] [-certtext <自己署名証明書の内容ファイル>] [-validity <有効日数>] [-sigalg <RSA 暗号用のサーバ証明書の署名アルゴリズム>] [-eccsigalg <ECC 用のサーバ証明書の署名アルゴリズム>] [-ecckeysize <ECC 用の秘密鍵のキーサイズ>] [-ext <X.509 証明書の拡張情報>]
```

- -key
作成された秘密鍵ファイルの出力先を絶対パスで指定します。このオプションを省略すると、ファイルは、RSA 暗号の場合は **httpsdkey.pem**、ECC の場合は **ecc-httpsdkey.pem** というファイル名で、デフォルトの出力先パス※に出力されます。
- -csr
作成された証明書発行要求ファイルの出力先を絶対パスで指定します。このオプションを省略すると、ファイルは、RSA 暗号の場合は **httpsd.csr**、ECC の場合は **ecc-httpsd.csr** というファイル名で、デフォルトの出力先パス※に出力されます。
- -cert
作成された自己署名証明書の出力先を絶対パスで指定します。このオプションを省略すると、ファイルは、RSA 暗号の場合は **httpsd.pem**、ECC の場合は **ecc-httpsd.pem** というファイル名で、デフォルトの出力先パス※に出力されます。
- -certtext
作成された自己署名証明書の内容ファイルの出力先を絶対パスで指定します。このオプションを省略すると、ファイルは、RSA 暗号の場合は **httpsd.txt**、ECC の場合は **ecc-httpsd.txt** というファイル名で、デフォルトの出力先パス※に出力されます。
- -validity
日数で自己署名証明書の有効期限を指定します。このオプションを省略すると、デフォルトの 3,650 日が使用されます。
- -sigalg
RSA 暗号用のサーバ証明書の署名アルゴリズムを **SHA256withRSA** または **SHA1withRSA** で指定します。このオプションを省略すると、デフォルトの **SHA256withRSA** が使用されます。
- -eccsigalg
ECC 用のサーバ証明書の署名アルゴリズムを **SHA512withECDSA**、**SHA384withECDSA**、**SHA256withECDSA**、または **SHA1withECDSA** で指定します。このオプションを省略すると、デフォルトの **SHA384withECDSA** が使用されます。
- -ecckeysize
ECC 用のサーバ証明書の秘密鍵のサイズを 256 または 384 ビットで指定します。このオプションを省略すると、デフォルトの 384 が使用されます。
- -ext

X.509 証明書の拡張情報を指定します。自己署名証明書および証明書署名要求に SAN (Subject Alternative Name) を設定する場合は、このオプションを指定します。指定方法は、Java の **keytool** コマンドの **ext** オプションに基づきます。Ops Center Automator で指定できる拡張情報は SAN だけであることに注意してください。ext オプションを複数回指定した場合は、最初の指定が有効になります。

以下に、拡張情報を指定する例を示します。

- **www.example.com** をホスト名として指定する場合：
`hcmads64ssltool -ext san=dns:www.example.com`
- **www.example.com** と **www.example.net** を複数のホスト名として指定する場合：
`hcmads64ssltool -ext san=dns:www.example.com, dns:www.example.net`

このコマンドは、RSA ファイルおよび ECC ファイルを指定した出力先パスに出力します。RSA ファイルは、指定したファイル名で、ECC ファイルは、指定したファイル名の先頭に「ecc-」が付いて出力されます。

注※ **key**、**csr**、**cert**、または **certtext** オプションを省略した場合のデフォルトの出力先は、次のとおりです。

<共通コンポーネントのインストールディレクトリ>/uCPSB11/httpsd/conf/ssl/server

2. 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 (管理サーバのホスト名) - 例: Automator-SC1
- ・ 組織単位 (セクション) - 例: Ops Center Automator
- ・ 組織名 (会社) - 例: Hitachi
- ・ 都市または地区名 - 例: Yokohama
- ・ 州または県名 (フルネーム) - 例: Kanagawa
- ・ 国名 (2 文字のコード) - 例: JP

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。

3. 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、本番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。

4. Ops Center Automator を停止します。

5. 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。

<共通コンポーネントのインストールディレクトリ>/uCPSB11/httpsd/conf/ssl/server

6. 次の場所から user_httpsd.conf ファイルを開きます。

<共通コンポーネントのインストールディレクトリ>/uCPSB11/httpsd/conf/user_httpsd.conf

7. user_httpsd.conf ファイル内で、以下のようになります。

- a. 番号記号 (#) 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
から
#HWSLogSSLVerbose On
```

ただし、`#SSLCACertificateFile` と `#Header set Strict-Transport-Security max-age=31536000` はコメントアウトしたままにしておく必要があります。

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

以下に、`user_httpsd.conf` ファイルの編集例を示します。

```
ServerName <管理サーバのホスト名>
Listen [::]:22015
Listen 22015
#Listen 127.0.0.1:22015
SSLEngine Off
#Listen [::]:22016
Listen 22016
<VirtualHost *:22016>
ServerName <管理サーバのホスト名>
SSLEngine On
SSLProtocol TLSv1.2
# SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-
GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-
SHA256:AES256-GCM-SHA384:AES128-GCM-SHA256
SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-
SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256
SSLCertificateKeyFile
"<共通コンポーネントのインストールディレクトリ>/uCPSB11/httpsd/conf/ssl/
server/httpsdkey.pem"
SSLCertificateFile
"<共通コンポーネントのインストールディレクトリ>/uCPSB11/httpsd/conf/ssl/
server/httpsd.pem"
SSLCertificateKeyFile
"<共通コンポーネントのインストールディレクトリ>/uCPSB11/httpsd/conf/ssl/
server/ecc-httpsdkey.pem"
SSLCertificateFile
"<共通コンポーネントのインストールディレクトリ>/uCPSB11/httpsd/conf/ssl/
server/ecc-httpsd.pem"
# SSLCACertificateFile
"<共通コンポーネントのインストールディレクトリ>/uCPSB11/httpsd/conf/ssl/
cacert/anycert.pem"
# Header set Strict-Transport-Security max-age=31536000
</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`

`#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。

ディレクティブを編集する場合、以下について注意してください：

- 同じディレクティブを 2 回指定しないでください。ただし、`SSLCertificateKeyFile` および `SSLCertificateFile` ディレクティブは、RSA 暗号用と ECC 用で 2 回、指定できます。
- ディレクティブの途中で改行を入れないでください。

- 以下に示すディレクティブでパスを指定する場合、シンボリックリンクまたはジャンクションポイントを指定しないでください。
- 以下に示すディレクティブで証明書および秘密鍵ファイルを指定する場合、PEM 形式のファイルを指定してください。
- httpsd.conf ファイルおよび hssso_httpsd.conf ファイルを編集しないでください。
- 次の行の番号記号 (#) は削除しないでください。

```
# Header set Strict-Transport-Security max-age=31536000
```

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 <管理サーバのホスト名>
Listen [::]:22015
Listen 22015
#Listen 127.0.0.1:22015
SSLEngine Off
#Listen [::]:22016
Listen 22016
<VirtualHost *:22016>
ServerName <管理サーバのホスト名>
SSLEngine On
SSLProtocol TLSv1.2
# SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256:AES256-GCM-SHA384:AES128-GCM-SHA256
SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-GCM-SHA256
SSLCertificateKeyFile
"<共通コンポーネントのインストールディレクトリ>/uCP SB11/httpsd/conf/ssl/server/httpsdkey.pem"
SSLCertificateFile
"<共通コンポーネントのインストールディレクトリ>/uCP SB11/httpsd/conf/ssl/server/server-certificate-or-self-signed-certificate-file"
SSLCertificateKeyFile
"<共通コンポーネントのインストールディレクトリ>/uCP SB11/httpsd/conf/ssl/server/ecc-httpsdkey.pem"
SSLCertificateFile
"<共通コンポーネントのインストールディレクトリ>/uCP SB11/httpsd/conf/ssl/server/ecc-httpsd.pem"
SSLCACertificateFile
"<共通コンポーネントのインストールディレクトリ>/uCP SB11/httpsd/conf/ssl/cacert/certificate-file-from-certificate-authority"
# Header set Strict-Transport-Security max-age=31536000
</VirtualHost>
HWSLogSSLVerbose On
```

8. Ops Center Automator を開始します。
9. **hcmds64chgur1** コマンドを使用して、次のように Ops Center Automator 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、Ops Center Automator 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、Ops Center Automator 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 Ops Center Automator 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://<Ops-Center-Automator 管理サーバの IP アドレスまたはホスト名>:<ポート番号 (SSL)>/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。

3.2.4 Web サービス接続の証明書をインポートする

共通コンポーネントのトラストストアに Web サービス接続の証明書をインポートする必要があります。

次のような Web サービス接続にサーバ証明書を使用するときに、共通コンポーネントのトラストストアに証明書をインポートする必要があります。

- Common Services
- Configuration Manager
- VMware vCenter
- BNA
- DCNM
- その他の Web サービス接続

次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、認証局の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、**hcnds64keytool** 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、新しいエイリアス名が既存のエイリアス名と衝突しないことを確認してください。

Windows の場合：

```
<共通コンポーネントのインストールフォルダ>%bin%hcnds64keytool -import -alias <エイリアス名> -keystore <共通コンポーネントのインストールフォルダ>%uCPsB11%hjdk
```

```
¥jdk¥lib¥security¥jssecacerts -storepass <トラストストアへのアクセスパスワード> -file <証明書ファイル名> -storetype JKS
```

hcmds64srv コマンドを実行して、サービスを再開します。

Unix の場合：

```
<共通コンポーネントのインストールディレクトリ>/uCPsB11/jdk/bin/keytool -import -alias <エイリアス名> -keystore <共通コンポーネントのインストールディレクトリ>/uCPsB11/hjdk/jdk/lib/security/jssecacerts -storepass <トラストストアへのアクセスパスワード> -file <証明書ファイル名> -storetype JKS
```

hcmds64srv コマンドを実行して、サービスを再開します。

追加のガイドライン

- ・ サードパーティ接続のセキュリティ設定の方法については、各製品のマニュアルを参照してください。たとえば、VMware vCenter の場合は、VMware のマニュアルを参照してください。
- ・ サードパーティのサーバ証明書を取得するには、関連する製品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。
- ・ DCNM をアップグレードすると、サーバ証明書が初期化されます。『Cisco DCNM Installation and Upgrade Guide for SAN Deployment』の「Restoring the certificates after an upgrade」に記載されている手順を実施する必要があります。
- ・ DCNM 11.5 を使用する場合は、『Cisco DCNM Installation and Upgrade Guide for SAN Deployment』の「Certificates」に記載されている手順に従って、Common Name に適切なホスト名を指定して証明書を作成します。

3.2.5 ESX クラスタサービスの VMware サーバ証明書をインストールする (Windows)

セキュア通信を使用するすべての Web サービス接続と同様に、Ops Center Automator が参照する Ops Center Automator 共通コンポーネントのトラストストアに、VMware サーバ証明書をインポートする必要があります。ただし、ESX クラスタサービステンプレートを使用する場合、VMware は一部の機能に独自（自己署名）のルート証明書を使用するため、VMware ルート証明書もインストールする必要があります。



メモ ESX クラスタサービステンプレートを使用しない場合は、この手順を完了する必要はありません。

Windows サーバに VMware サーバ証明書をインストールするには、次の手順に従う必要があります。

操作手順

1. VMware サーバ証明書を次のようにダウンロードします。
 - a. Web ブラウザを使用して vCenter ユーザーインターフェースにアクセスします。
 - b. 右側の画面で、[信頼されたルート CA 証明書をダウンロード] を選択します。
 - c. Ops Center Automator 共通コンポーネントのトラストストアが存在するサーバ上でダウンロードする場所を選択し、ダウンロードを確認します。
2. 共通コンポーネントのトラストストアが存在するサーバで、zip ファイルをダウンロードした場所に移動し、そのファイルを解凍します。



メモ ダウンロードしたファイルの拡張子が.zip でない場合は、拡張子を.zip に変更します。

両方の証明書ファイルが含まれる、.certs フォルダが解凍されます。

3. 証明書をインストールします。
 - a. 拡張子.crt のファイルの上で右クリックし、[証明書のインストール] を選択します。
証明書のインポート ウィザードが開きます。
 - b. [ローカル コンピューター] を選択し、[次へ] をクリックします。
 - c. [証明書をすべて次のストアに配置する] を選択します。
 - d. [参照] をクリックし、[信頼されたルート証明機関] を選択して、[完了] をクリックします。
 - e. 拡張子.crl のファイルについても手順 a から d を実施します。
- 証明書がトラストストアにインストールされます。



メモ ESX クラスタサービステンプレートを使用する場合は、『Hitachi Ops Center Automator ユーザーズガイド』に説明されているように、Python もインストールする必要があります。

3.2.6 ESX クラスタサービスの VMware サーバ証明書をインストールする (Linux)

セキュア通信を使用するすべての Web サービス接続と同様に、Ops Center Automator が参照する Ops Center Automator 共通コンポーネントのトラストストアに、VMware サーバ証明書をインポートする必要があります。ただし、ESX クラスタサービステンプレートを使用する場合、VMware は一部の機能に独自（自己署名）のルート証明書を使用するため、VMware ルート証明書もインストールする必要があります。



メモ ESX クラスタサービステンプレートを使用しない場合は、この手順を完了する必要はありません。

Linux サーバに VMware サーバ証明書をインストールするには、次の手順に従う必要があります。

操作手順

1. VMware サーバ証明書を次のようにダウンロードします。
 - a. Web ブラウザを使用して vCenter ユーザーインターフェースにアクセスします。
 - b. 右側の画面で、[信頼されたルート CA 証明書をダウンロード] を選択します。
 - c. Ops Center Automator 共通コンポーネントのトラストストアが存在するサーバ上でダウンロードする場所を選択し、ダウンロードを確認します。
2. 共通コンポーネントのトラストストアが存在するサーバで、zip ファイルをダウンロードした場所に移動し、そのファイルを解凍します。



メモ ダウンロードしたファイルの拡張子が.zip でない場合は、拡張子を.zip に変更します。

拡張子が.0 (xxx.0) のファイルが含まれる「lin」という名前のフォルダが解凍されます。

3. 「xxx.0」 ファイルを次のディレクトリにコピーします。
/etc/pki/tls/certs
証明書がトラストストアにインストールされます。



メモ ESX クラスターサービステンプレートを使用する場合は、『*Hitachi Ops Center Automator ユーザーズガイド*』に説明されているように、Python もインストールする必要があります。

3.2.7 Ops Center Automator サーバと REST API サーバの間で SSL 通信を使用するための設定を指定する（認証局によるサーバ証明書を使っている場合）

自己署名証明書または認証局の証明書を使用することで、Ops Center Automator サーバと REST API サーバの間で使用する SSL 通信を設定することが可能です。

詳細については、『*Hitachi Ops Center API Configuration Manager REST API リファレンスガイド*』の「REST API クライアントと REST API サーバ間で SSL 通信するよう設定する（自己署名証明書を使用する場合）」または「REST API クライアントと REST API サーバ間で SSL 通信するよう設定する（認証局が発行したサーバ証明書を使用する場合）」を参照してください。

3.2.8 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確認することで、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。

共通コンポーネントのサーバ証明書の有効期限を確認するには、次のコマンドを実行します。

Windows の場合：

```
<共通コンポーネントのインストールフォルダ>%bin%\hcmds64keytool -printcert -v -file <サーバ証明書のパス>
```

Linux の場合：

```
<共通コンポーネントのインストールディレクトリ>/uCPSB11/jdk/bin/keytool -printcert -v -file <サーバ証明書のパス>
```



メモ 自己署名証明書の有効期限は、サーバ間の接続時には検証されません。Ops Center Automator サーバと Web サーバの接続時に証明書の有効期限を確認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。

3.3 監査ログ

監査ログには、Ops Center Automator サーバ上でのすべてのユーザーアクションが記録されます。監査ログには、外部サービス、認証、設定へのアクセス、サービスの開始や停止などのイベントが記録されます。監査ログを調べることで、システムの利用状況の確認や不正アクセスの監査ができます。

3.3.1 監査ログを設定する

監査ログには、Ops Center Automator サーバ上でのすべてのユーザーアクションが記録されます。監査ログには、外部サービス、認証、設定へのアクセス、サービスの開始や停止などのイベントが記録されます。監査ログを調べることで、システムの利用状況の確認や不正アクセスの監査ができます。

Windows の場合、監査ログデータは、イベントログファイル（アプリケーションログファイル）に出力されます。Linux の場合、データは syslog ファイルに出力されます。

以下の表に、共通コンポーネントを使用する製品によって生成される、監査ログデータのカテゴリを示します。異なる製品によってさまざまなタイプの監査ログデータが生成されます。

カテゴリ	説明
StartStop	ハードウェアやソフトウェアの起動または停止を示すイベント <ul style="list-style-type: none"> OS の起動またはシャットダウン ハードウェアコンポーネント（マイクロコンポーネントを含む）の起動または停止 ストレージシステムまたは SVP 上のソフトウェア、および共通コンポーネントを使用する製品の起動または停止
Failure	ハードウェアまたはソフトウェアの障害を示すイベント <ul style="list-style-type: none"> ハードウェア障害 ソフトウェア障害（メモリエラーなど）
LinkStatus	デバイス間のリンク状態を示すイベント リンクが接続しているか、または接続が切れているか
ExternalService	外部サービスとの通信結果を示すイベント <ul style="list-style-type: none"> NTP や DNS などの外部サーバとの通信 管理サーバ（SNMP）との通信
Authentication	デバイス、管理者、またはエンドユーザーが、接続や認証に成功または失敗したことを示すイベント <ul style="list-style-type: none"> ファイバーチャネルログイン デバイス認証（ファイバーチャネル・セキュリティプロトコル認証、iSCSI ログイン認証、SSL サーバ/クライアント認証） 管理者またはエンドユーザー認証
AccessControl	デバイス、管理者、またはエンドユーザーが、リソースへのアクセスに成功または失敗したことを示すイベント <ul style="list-style-type: none"> デバイスのアクセスコントロール 管理者またはエンドユーザーのアクセスコントロール
ContentAccess	重要データへのアクセスの試みが成功または失敗したことを示すイベント <ul style="list-style-type: none"> NAS 上の重要ファイルまたは HTTP がサポートされている場合のコンテンツへのアクセス 監査ログファイルへのアクセス
ConfigurationAccess	管理者が許可されている操作に成功または失敗したことを示すイベント <ul style="list-style-type: none"> 設定情報の参照または更新 アカウントの追加や削除を含むアカウント設定の更新 セキュリティ設定 監査ログ設定の参照または更新
Maintenance	実施したメンテナンス操作が成功または失敗したことを示すイベント <ul style="list-style-type: none"> ハードウェアコンポーネントの追加または削除 ソフトウェアコンポーネントの追加または削除
AnomalyEvent	しきい値超過などの異常が発生したことを示すイベント <ul style="list-style-type: none"> ネットワークトラフィックしきい値の超過

カテゴリ	説明
	<ul style="list-style-type: none"> CPU 負荷しきい値の超過 内部に一時的に保存された監査ログデータが制限に達するか、ラップアラウンドが発生したことの事前通知
	異常な通信が発生したことを示すイベント <ul style="list-style-type: none"> 通常使用しているポートに対する SYN フラッド攻撃またはプロトコル違反 未使用ポートに対するアクセス（ポートスキャンなど）

3.3.2 監査ログを有効にする

Ops Center Automator サーバの監査ログを有効にし、監査イベントを監査ログに出力するよう変更するには、まず、共通コンポーネント用の環境設定ファイル（auditlog.conf）を設定します。その後で、Ops Center Automator サーバを再起動する必要があります。



メモ

- 監査ログはデフォルトで無効になっています。必要に応じて設定を有効にしてください。
- 大量の監査ログデータが出力される場合があります。ログファイルのサイズを変更し、生成されたログファイルを必要に応じてバックアップまたはアーカイブしてください。

操作手順

- 管理者権限（Windows）または root 権限（Linux）のユーザーとして、Ops Center Automator にログインします。
- 以下のどちらかにある、auditlog.conf ファイルを開きます。

Windows の場合

<共通コンポーネントのインストールフォルダ>\%conf%\sec\auditlog.conf

Linux の場合：

<共通コンポーネントのインストールディレクトリ>/conf/sec/auditlog.conf



メモ auditlog.conf ファイルは、共通コンポーネント用の環境設定ファイルです。

- 監査ログを有効にするには、auditlog.conf ファイルの Log.Event.Category プロパティに監査イベントカテゴリを指定します。
- 監査ログを無効にするには、auditlog.conf ファイルの Log.Event.Category プロパティに指定されている監査イベントカテゴリをすべて削除します。
- Ops Center Automator サービスを再起動します。

3.3.3 auditlog.conf ファイルの設定

以下の値を auditlog.conf ファイルに設定できます。

Log.Facility（Linux でのみ有効）

Linux で、syslog ファイルに監査ログデータを出力するファシリティ（ログタイプ）の数値を指定します。（デフォルト値：1）

Windows では、Log.Facility が指定されても無視されます。無効な値や数値以外の文字が指定された場合は、デフォルト値が使用されます。

以下の表に、Log.Facility に指定できる値と syslog.conf ファイルで定義されているファシリティの対応を示します。

Log.Facility に 指定できる値	syslog.conf ファイルで定義されているファシリティ
1	user
2	mail※
3	daemon
4	auth※
6	lpr※
16	local0
17	local1
18	local2
19	local3
20	local4
21	local5
22	local6
23	local7
注※ この値を指定することはできませんが、推奨していません。	

syslog ファイルに出力される監査ログをフィルタリングするには、Log.Facility に指定されたファシリティと各監査イベントの重要度を組み合わせます。

以下の表に、監査イベントの重要度と syslog.conf ファイルで定義されている重要度の対応を示します。

監査イベントの重要度	syslog.conf ファイルで定義されている重要度
0	emerg
1	alert
2	crit
3	err
4	warning
5	notice
6	info
7	debug

Log.Event.Category

出力される監査イベントカテゴリを指定します。(デフォルト値：なし)

複数のカテゴリを指定する場合は、カテゴリとカテゴリをカンマ (,) で区切ります。この場合、カテゴリとコンマの間にスペースを挿入しないでください。Log.Event.Category が指定されていないと、監査ログデータは出力されません。Log.Event.Category は大文字小文字を区別しません。無効なカテゴリ名が指定された場合、指定したファイル名は無視されます。

有効なカテゴリ：StartStop、Failure、LinkStatus、ExternalService、Authentication、AccessControl、ContentAccess、ConfigurationAccess、Maintenance、AnomalyEvent

Log.Level (Windows でのみ有効)

出力される監査イベントの重要度を指定します。(デフォルト値: 6)

指定した重要度レベル以下のイベントがイベントログファイルに出力されます。

各監査イベントの重要度については、監査ログに出力される監査イベントのリストを参照してください。

Log.Level は、Windows でのみ有効です。Linux では、Log.Level が指定されても無視されます。また、無効な値や数字以外の文字が指定された場合は、デフォルト値が使用されます。

以下の表に、Log.Level に指定できる値とイベントログに表示されるレベルの対応を示します。

Log.Level に指定できる値	イベントログに表示されるレベル
0	エラー
1	
2	
3	
4	警告
5	情報
6	
7	

3.3.4 auditlog.conf ファイルのサンプル

以下に、auditlog.conf ファイルの例を示します。

```
# Specify an integer for Facility. (specifiable range: 1-23)
Log.Facility 1

# Specify the event category.
# You can specify any of the following:
# StartStop, Failure, LinkStatus, ExternalService,
# Authentication, AccessControl, ContentAccess,
# ConfigurationAccess, Maintenance, or AnomalyEvent.
Log.Event.Category StartStop,Failure,LinkStatus,ExternalService,Authentication,
AccessControl,ContentAccess,ConfigurationAccess,Maintenance,AnomalyEvent

# Specify an integer for Severity. (specifiable range: 0-7)
Log.Level 6
```

上記の例では、監査イベントのすべてのタイプが出力されています。

Windows の場合には、Log.Level 6 がエラー、警告、情報のレベルに対応するログデータを出力します。Linux の場合には、Log.Facility 1 が、syslog.conf ファイルに user ファシリティとして定義されている syslog ファイルに監査ログデータを出力します。

3.3.5 監査ログに出力されるデータのフォーマット

監査ログデータは Windows のイベントログファイルまたは Linux の syslog ファイルに出力されます。

監査ログに出力されるデータの形式を次に示します。

Windows の場合

プログラム名 [プロセス ID]: メッセージ部

Linux の場合

syslog ヘッダー部 メッセージ部

syslog ヘッダー部の形式は、OS の環境設定によって異なります。必要な場合は設定を変更してください。

例えば、rsyslog を使用し、/etc/rsyslog.conf で以下を指定する場合は、RFC5424 に従った形式でメッセージが出力されます。

```
$ActionFileDefaultTemplate RSYSLOG_SyslogProtocol23Format
```

メッセージ部の形式と内容は次のとおりです。メッセージ部のうち、最大 953 シングルバイト文字が syslog ファイルに表示できます。

統一識別子, 統一仕様リビジョン番号, 通番, メッセージ ID, 日付・時刻, 検出エンティティ, 検出場所, 監査事象の種別, 監査事象の結果, 監査事象の結果サブジェクト識別情報, ハードウェア識別情報, 発生場所情報, ロケーション識別情報, FQDN, 冗長化識別情報, エージェント情報, リクエスト送信元ホスト, リクエスト送信元ポート番号, リクエスト送信先ホスト, リクエスト送信先ポート番号, 一括操作識別子, ログ種別情報, アプリケーション識別情報, 予約領域, メッセージテキスト

項目※	説明
統一識別子	CELFSS に固定
統一仕様リビジョン番号	1.1 に固定
通番	監査ログメッセージのシリアル番号
メッセージ ID	メッセージ ID
日付・時刻	メッセージが出力された日時。この項目は、 yyyy-mm-ddThh:mm:ss.s タイムゾーンの形式で出力されます。
検出エンティティ	コンポーネント名またはプロセス名
検出場所	ホスト名
監査事象の種別	イベントタイプ
監査事象の結果	イベント結果
監査事象の結果サブジェクト識別情報	イベントに対応するアカウント ID、プロセス ID、または IP アドレス
ハードウェア識別情報	ハードウェアモデルまたはシリアル番号
発生場所情報	ハードウェアコンポーネントの識別情報
ロケーション識別情報	場所の識別情報
FQDN	完全修飾ドメイン名
冗長化識別情報	冗長性識別情報
エージェント情報	エージェント情報
リクエスト送信元ホスト	リクエスト送信元のホスト名
リクエスト送信元ポート番号	リクエスト送信元のポート番号
リクエスト送信先ホスト	リクエスト送信先のホスト名

項目※	説明
リクエスト送信先ポート番号	リクエスト送信先のポート番号
一括操作識別子	プログラムによる操作の通番
ログ種別情報	BasicLog または DetailLog に固定
アプリケーション識別情報	プログラム識別情報
予約領域	出力なし。予約領域です。
メッセージテキスト	コンテンツは監査イベントによって変わります。 表示できない文字は、アスタリスク (*) として出力されます。
注※ 一部の監査イベントに出力されない項目もあります。	

監査ログのログインイベントのメッセージ部の例を次に示します。

```
CELFSS,1.1,3,KNAE20002-I,2021-09-03T21:31:56.8+09:00,HAD,management-
host,Authentication,Success,subj:uid=sysadmin,autoAuth>Login,BasicLog,HAD
,"ログインに成功しました。"
```

3.4 別のホストへ Ops Center Automator を移動する

必要に応じて、Ops Center Automator を別のホストに移動できます。



メモ 移動元のホスト名または IP アドレスと移動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

以下の設定が移動元のホストと移動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。
- Ops Center Automator によって使用される OS ユーザーのアカウント。
- Ops Center 製品の環境（構成、バージョン、およびリビジョン）。
- Ops Center Automator のインストールパス。

Ops Center Automator 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. 移動元ホストで Ops Center Automator のバックアップを完了します。
 - 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移動先のホストに移動します。
4. 移動先のホストの管理サーバにログインします。
5. 移動先のホストで、Ops Center Automator 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設定を変更します。

- ・ 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。

- ・ <バックアップ先のディレクトリ>%HBase%base%conf
- ・ <バックアップ先のディレクトリ>%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. Common Services から Ops Center Automator を削除し、登録します。詳細については、「[3.5 Common Services を別のホストに移動する](#)」を参照してください。
8. **hcnds64srv /start** コマンドを実行して、サービスを再開します。

3.5 Common Services を別のホストに移動する

Common Services を別のホストに移動できます。

操作手順

1. Common Services から Ops Center Automator を削除します。Ops Center Automator を削除するには、『Hitachi Ops Center インストールガイド』を参照してください。
2. **setupcommonservice** コマンドを実行し、Common Services をセットアップします。
3. 必要に応じて、ユーザーグループおよびサービスグループの権限を変更します。

3.6 システム構成を変更する

config_user.properties ファイルを編集すると、ログやタスクなど、Ops Center Automator のさまざまな設定を構成できます。ファイルを変更して保存した後で、Ops Center Automator エンジン Web サービスは再起動する必要があることに注意してください。

このファイルを編集することで、以下の設定を変更できます。

- ・ 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
- ・ 同時実行するプラグインの最大数

ファイルは、次のフォルダにあります。

<Ops-Center-Automator のインストールフォルダ>%conf

ファイルは、次の形式を使用します。

specification-key-name=setting

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送信されます。
- ・ 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、最後に指定したキーが有効になります。

表 1 config_user.properties ファイルの設定

カテゴリ	キー名	設定	値	デフォルト値
HTTP 接続ポート番号	server.http.port	Ops Center Automator サーバと共通コンポーネント間の HTTP 通信に使用されるポート番号を指定します。	0～65535	22015
ログ ¹	logger.message.server.MaxBackupIndex	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.MaxFileSize	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.MaxBackupIndex	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
	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
繰り返し	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	対象機器の SSH ポート番号を指定します。	0～65535	22
	telnet.port.number	対象機器の Telnet ポート番号を指定します。	0～65535	23

カテゴリ	キー名	設定	値	デフォルト値
一般的なコマンド リモートコマンド ファイル転送 ターミナル接続	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。 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カスタム部品 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。	1~1024	100
	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カスタム部品。 [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。	1~1024	100
	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。	1~1024	100

カテゴリ	キー名	設定	値	デフォルト値
		<p>注:プロパティ値の単位はキロバイト (KB) です。</p> <p>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。</p> <ul style="list-style-type: none"> - 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	<p>カスタム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。</p> <p>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。</p>	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 文字	"" (null 文字)
	plugin.localMode	<p>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。</p> <p>true : 有効 false : 無効</p>	true/false	true
リモートファイル操作の再試行	plugin.remoteFileAccess.retry.times	<p>カスタム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。</p> <p>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。</p>	0～100	0

カテゴリ	キー名	設定	値	デフォルト値
ターミナル接続	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。(1~1,024 文字) 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、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。	正規表現パターンで使用できる文字列。	login Login Name Username UserName
	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。(1~1,024 文字) 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。	正規表現パターンで使用できる文字列。	password Password PassWord
	telnet.connect.wait	対象機器との Telnet 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カスタム部品を含む、実行フォルダのパスを指定します。実行フォルダは、事前に作成しておく必要があります。 カスタム部品の [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。	0~128 文字の文字列	"" (null 文字)
	plugin.remoteCommand.executionDirectory.ssh	対象ホストの OS が UNIX の場合にカスタム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 文字の文字列	"" (null 文字)
	plugin.remoteCommand.workDirectory.ssh	対象ホストの OS が UNIX の場合、ファイル転送部品またはカスタム部品の実行時に使用	1~128	/tmp/ Hitachi_AO

カテゴリ	キー名	設定	値	デフォルト値
		される作業ディレクトリを指定します。ディレクトリ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	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	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	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	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	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	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 Ops Center Automator のログインユーザー 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
Service Builder	client.editor.upload.maxfilesize	[Service Builder Edit] ウィンドウで、Ops Center Automator の操作に使用される端末からサーバにアップロ	1～10	3

カテゴリ	キー名	設定	値	デフォルト値
		ードできる最大ファイルサイズ (MB 単位) を指定します。		
	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	1000
	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
デバッグ	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
長期実行中のタスクのチェック間隔しきい値	server.longRunning.check.interval	長期実行中のタスクのチェック間隔しきい値 (分数)	0～20160	2880
長期実行中の監視間隔	server.longRunning.monitor.interval	長期実行中の監視間隔 (秒数)	1～3600	60
Web クライアント	plugin.http.connect.timeout	HTTP/HTTPS 接続が確立され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 接続の確立後、データが読み込ま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

カテゴリ	キー名	設定	値	デフォルト値
部品実行	plugin.threadPoolSize	部品の最大同時実行数を指定します。 製品同梱のサービステンプレートのみを使用する場合、本プロパティ値を 100 に設定して運用が可能です。カスタムサービステンプレートを使用する場合は、デフォルト値から変更後、必ず評価を行い、問題が発生しないことを確認してから、本番運用に移行してください。	1～100	10

¹ タスクのログ出力しきい値は、サービス共有プロパティで設定します。

[例]

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 パフォーマンスモードを設定する

Ops Center Automator には、スタンダードモードとハイパフォーマンスモードの 2 つの動作モードがあります。ハイパフォーマンスモードは、複数のタスクの実行に適しており、スタンダードモードよりも多くのリソースを使用します。

スタンダードモードとハイパフォーマンスモードを切り替えるには、**changemode** コマンドを使用します。**changemode** コマンドの使用方法的詳細については、『*Hitachi Ops Center Automator ユーザーズガイド*』の「changemode コマンド」を参照してください。



メモ 複数の Online Migration with Configuration Manager タスクを実行する場合は、ハイパフォーマンスモードで操作する必要があります。詳細については、『Hitachi Ops Center Automator ユーザーズガイド』の「Online migration with Configuration Manager サービステンプレート」を参照してください。

3.8 メール通知を構成する

メール通知設定を構成し、障害発生時またはタスクに問題が発生した場合に、メール通知を受信するようにします。メールアドレス、件名、障害や問題について受信する情報のタイプを構成できます。



メモ システムのメール通知を有効にするには、[管理] タブでシステム・パラメータを設定する必要があります。詳細については、『Hitachi Ops Center Automator ユーザーズガイド』を参照してください。

メール定義ファイル、mailDefinition は XML 形式です。次のディレクトリにあります。

<Ops-Center-Automator のインストールフォルダ>%conf

定義ファイルは、次の形式を使用します。

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<mail xmlns="http://www.example.com/products/it/software/xml/
automation/">
<title><メールタイトル></title>
<body><メール本文></body> </mail>
```

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 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 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>タグの値が省略された場合には、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。
- すべてのエントリで大文字と小文字が区別されます。

設定を変更するには、mailDefinition ファイルのメールの件名およびメール本文のセクションを編集します。

表 2 メール通知設定

設定	XML 要素	文字列長	デフォルト値
メール通知に使用されるメールの件名	<title>	0～9,999 バイトの文字列	[Ops Center Automator] \$TASK_NAME\$が \$TASK_STATUS\$に変更 されました。
メール通知に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_NA ME\$ タスク名： \$TASK_NAME\$ 実行 者:\$USER_NAME\$ タス

設定	XML 要素	文字列長	デフォルト値
			ク詳細： \$TASK_DETAIL_URL\$

表 3 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
&	&
<	<
>	>
"	"
'	'

表 4 メール通知に埋め込まれる文字

埋め込まれる文字	項目	備考
\$SERVICE_GROUP_NAME\$	サービスグループ名	サービスグループ名を表す文字列が設定されます。
\$TASK_NAME\$	タスク名	タスクのプロパティの形式に従ってタスク名が設定されます。
\$TASK_ID\$	タスク ID	なし
\$TASK_KIND\$	タスク種別	
\$SERVICE_NAME\$	サービス名	
\$TASK_TAGS\$	タスクのタグ	
\$TASK_STATUS\$	タスクの状態	
\$EXECUTION_DATE\$	実行操作日時	
\$PLANNED_START_DATE\$	開始予定日時	
\$START_DATE\$	開始日時	
\$END_DATE\$	終了日時	
\$SCHEDULE_PERIOD\$	定期実行周期	
\$SCHEDULE_TIME\$	定期実行時刻	
\$SCHEDULE_START_DATE\$	定期実行適用開始日	
\$USER_NAME\$	実行者	
\$TASK_DETAIL_URL\$	[タスク詳細] ウィンドウの URL	http で始まる URL が設定されます。

3.9 エージェントレス接続の対応 OS

次の OS およびバージョンを、エージェントレス接続の操作対象機器として利用できます。

操作対象機器の OS が Windows の場合は SMB および RPC、Linux の場合は SSH を使用して機器に接続します。ターミナル接続部品を使用して操作対象機器に接続する場合は、Telnet または SSH を使用します。

- Windows の場合：
 - Windows Server 2012 Standard (x64)
 - Windows Server 2012 Datacenter (x64)
 - Windows Server 2012 R2 Standard (x64)
 - Windows Server 2012 R2 Datacenter (x64)
 - Windows Server 2016 Standard (x64)
 - Windows Server 2016 Datacenter (x64)
 - Windows Server 2019 Standard (x64)[※]
 - Windows Server 2019 Datacenter (x64)[※]

注※ Ops Center Automator サーバのホスト OS が Windows の場合のみ、利用できます。

- Linux の場合：
 - Red Hat Enterprise Linux 7.1 - 7.9 (x64)
 - Red Hat Enterprise Linux 8.1、8.2、8.4 (x64)
 - Oracle Linux 7.2 - 7.9 (x64)
 - Oracle Linux 8.1、8.2、8.4 (x64)

カスタム部品、汎用コマンド実行部品、ファイル転送部品が操作対象機器の OS で指定されたコマンド以外で、各部品が実行するコマンドを次に示します。部品を使用する場合、各コマンドがインストール済みである必要があります。

- カスタム部品
/bin/bash、/usr/bin/id、/bin/echo、/usr/bin/find、/usr/bin/test、/bin/mkdir、/bin/chmod、/bin/gunzip、/bin/tar、/bin/rm、/bin/cp、/bin/uname、/bin/su
- 汎用コマンド実行部品
/bin/bash、/usr/bin/id、/bin/echo、/usr/bin/test、/bin/uname、/bin/su
- ファイル転送部品 (送信：部品プロパティ **transferMode** の値が **send** の場合)
/bin/bash、/usr/bin/id、/usr/bin/test、/bin/mkdir、/bin/chmod、/bin/gunzip、/bin/tar、/bin/rm、/bin/cp、/bin/uname、/bin/su
- ファイル転送部品 (受信：部品プロパティ **transferMode** の値が **receive** の場合)
/bin/bash、/usr/bin/id、/usr/bin/test、/bin/mkdir、/bin/chmod、/usr/bin/zip、/bin/rm、/bin/uname、/bin/su

カスタム部品とファイル転送部品では、SCP にて操作対象機器にファイルを転送します。操作対象の機器は、SCP でファイル転送可能な環境にしてください。なお、操作対象の機器が Linux で、接続するユーザーの **.bashrc** で文字列を出力している場合は、SCP に失敗するおそれがあります。また、エージェントレス接続先に Telnet または SSH で接続する場合、接続ユーザーのログイン スクリプトに対話環境が前提である **stty**、**tty**、**tset**、**script** コマンドなどを記載しないでください。記

載されている場合は、ログインスクリプトを変更する、または、これらのコマンドを実行しないログインスクリプトを使用するユーザーを新たに作成してください。

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

Ops Center Automator の部品およびサービスが、部品によるタスクが実行され、アクションが実施されるリモートマシンと通信できるようになる前に、リモートマシン接続情報を構成する必要があります。

開始する前に、以下のことを確認してください。

- 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。
＜Ops-Center-Automator のインストールフォルダ＞¥Automation¥conf¥plugin¥destinations
- ファイル名は、次の形式を使用します。
＜ホスト名＞.properties, ＜IPv4 アドレス＞.properties, ＜IPv6 アドレス＞.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.

サンプルファイルは、次の場所にあります。

＜Ops-Center-Automator のインストールフォルダ＞¥Automation¥conf¥plugin¥destinations¥#sample.properties

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。
- 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。
- 接続先プロパティファイルを編集した場合、そのファイルを参照する部品が実行されると、新しい定義が適用されます。

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ガイドライン

クラスタの対象機器に情報を入力する場合：

- クラスタ環境で対象機器が Windows Server 2012 または Windows Server 2012 R2 を実行している場合、作業フォルダ（wmi.workDirectory.sharedName および wmi.workDirectory.sharedPath）を設定する必要があります。設定しないと、部品が接続エラーの原因となります。

- ・ カスタム部品でスクリプトを実行する場合は、実行フォルダ（`common.executionDirectory`）を指定する必要があります。指定しないと、スクリプトは転送されません。

キー名	設定	有効値	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（ <code>config_user.properties</code> ）の <code>telnet.port.number</code> 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： <ul style="list-style-type: none"> ・ 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 ・ カスタム部品 この設定は、プロパティファイル（ <code>config_user.properties</code> ）の <code>ssh.port.number</code> 設定に優先します。	0～65535	0	65535
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われる文字列	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	正規表現パターンで使われる文字列	1 文字	1024 文字

キー名	設定	有効値	最小値	最大値
	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。			
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	Windows 対象機器のプロパティです。対象でのコマンド実行時にファイルが送信される共有フォルダの共有フォルダ名を指定します。フォルダは wmi.workDirectory.sharePath 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、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
wmi.workDirectory.sharedPath	Windows 対象機器のプロパティです。対象でのコマンド実行時にファイルが送信さ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使用している場合、実行フォルダは、このプロパティにリストされるパスの下に¥Hitachi¥CMALib¥HAD¥home になります。フォルダは wmi.workDirectory.shareName 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、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	Linux/Unix 対象機器のプロパティです。ファイル転送部品またはカスタム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	1 バイトの英数字、「/」、「-」、「_」、および「。」。	0 文字	128 文字

キー名	設定	有効値	最小値	最大値
	部品の接続先および受信先として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カスタム部品が使用されるときに、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。新しいディレクトリのアクセス権限は、必ず 777 であることを確認してください。優先されるのは、 <code>config_user.properties</code> ファイルで定義された <code>plugin.remoteCommand.workDirectory.ssh</code> の値です。0～128 文字の文字列を指定します。			
<code>common.executionDirectory</code>	対象に対してカスタム部品を実行するときの実行フォルダを指定します。部品定義で定義された実行フォルダ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、 <code>config_user.properties</code> ファイルで定義された <code>plugin.remoteCommand.executionDirectory.wmi</code> と <code>plugin.remoteCommand.executionDirectory.ssh</code> の値です。0～128 文字の文字列を指定します。	任意の文字列	0 文字	128 文字

3.11 エージェントレス接続の Windows 前提条件

エージェントレス接続を使用するには、次のセクションに記載されている Windows の前提条件が必要です。

サポートされるユーザー

エージェントレス接続では、次のユーザーを使用できます。

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

administrator グループに属するユーザーを使用する場合は、コマンド実行時に UAC（ユーザーアクセス制御）昇格が適用されないことに注意してください。

レジストリを編集する必要があります。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 を再起動する必要はありません。

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)

必要に応じて、コマンドプロンプトで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下にエントリを設定し、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で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の SSH 前提条件

エージェントレス接続を使用するには、次のセクションに記載されている SSH プロトコル前提条件が必要です。

SSH 前提条件は次の部品で必要です。

- カスタム部品
- 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。
`/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、コメントアウトの番号記号 (`#`) 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。
`systemctl restart sshd`



メモ このコマンドは、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。
`/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、コメントアウトの番号記号 (`#`) 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。

```
systemctl restart sshd
```



メモ このコマンドは、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、Ops Center Automator がインストールされる OS 上で作成することを推奨します。



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 RHEL 8.4（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。

Ops Center Automator への秘密鍵の配置

Ops Center Automator 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. Ops Center Automator 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。
`/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。
 - `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、コメントアウトの番号記号 (#) を削除します。)
 - `UsePAM` の値を **yes** に設定します。(UsePAM の行がコメントアウトされている場合は、コメントアウトの番号記号 (#) 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。

```
systemctl restart sshd
```



メモ このコマンドは、OS 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 Configuration Manager サーバで Java ヒープメモリサイズを設定する

複数の Online Migration with Configuration Manager タスクを実行する場合、Configuration Manager サーバが使用する Java ヒープのサイズを 6,144MB に変更する必要があります。

前提条件

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



ヒント 次の場所に格納されている `StartupV.properties` ファイルの `rest.java.heapMemory.size` プロパティの値を確認することで、現在設定されている値を確認できます。

REST API のインストール先フォルダ¥data¥properties¥StartupV.properties

このファイルが存在しない場合、またはファイルに `rest.java.heapMemory.size` プロパティが含まれていない場合は、デフォルト値が設定されていることを示しています。

操作手順

1. 次のコマンドを実行します。

```
REST API のインストール先フォルダ¥bin¥setProperty.bat  
rest.java.heapMemory.size 6144
```

このコマンドを実行すると、Configuration Manager サーバが再起動します。コマンドラインの最後に `-noRestart` を指定すると、サーバを再起動せずにコマンドが実行されます。

setProperty コマンドを実行すると、StartupV.properties ファイルの `rest.java.heapMemory.size` プロパティの値が 6144 に変更されます。ファイルが存在しない場合は作成されます。

このコマンドを実行するたびに、現在の StartupV.properties ファイルがバックアップされます。バックアップファイルは同じディレクトリに作成され、バックアップファイルの名前には作成日時が含まれます（例：StartupV_20200220-093320.properties）。

外部認証サーバでのユーザー管理

この章では、外部認証サーバでユーザー認証を設定する方法について説明します。

□ 4.1 外部認証サーバでのユーザー管理

4.1 外部認証サーバでのユーザー管理

外部認証サーバ（LDAP または Kerberos）に登録したユーザーアカウントを使用して Ops Center Automator にログインできます。外部認証サーバと連携するための設定は Common Services で行います。詳細は、『*Hitachi Ops Center インストールガイド*』の Active Directory との連携について説明している個所を参照してください。

Ops Center Automator をアンインストールする

この章では、Ops Center Automator をアンインストールする方法について説明します。

- 5.1 Ops Center Automator をアンインストールする (Windows)
- 5.2 クラスタ環境で Ops Center Automator をアンインストールする
- 5.3 Ops Center Automator をアンインストールする (Linux)

5.1 Ops Center Automator をアンインストールする (Windows)

Windows 環境で Ops Center Automator をアンインストールするには、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

- Ops Center Automator 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、タスクが停止または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。

操作手順

1. Windows に管理者としてログインします。
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。
＜共通コンポーネントのインストールフォルダ＞¥bin¥hcmds64srv /stop
3. [コントロールパネル] を開き、[プログラムと機能] または [プログラムのアンインストール] を選択します。
4. [Hitachi Ops Center Automator] を選択して [アンインストール] をクリックするか、プログラムを選択し、右クリックして [アンインストール] を選択します。
5. 画面の指示に従って、アンインストールを進めます。
[アンインストール前の確認] ウィンドウで [削除] をクリックすると、ソフトウェアのアンインストールプロセスが開始されます。
アンインストールプロセスによって、＜Ops-Center-Automator のインストールフォルダ＞¥Automation フォルダが削除されます。
6. Ops Center Portal から Ops Center Automator の情報を削除します。

操作結果

Ops Center Automator がホストからアンインストールされます。

5.2 クラスタ環境で Ops Center Automator をアンインストールする

Ops Center Automator 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 Ops Center Automator ソフトウェアをアンインストールします。



メモ Ops Center Automator をアンインストールした場合、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、Ops Center Automator 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 [移動] を選択し、[ノードを選択] または [このサービスまたはアプリケーションを別のノードに移動] を選択します。

2. 次のコマンドを使用して、Ops Center Automator の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。
＜共通コンポーネントのインストールフォルダ＞¥ClusterSetup
¥hcmds64clustersrvstate /soff /r <グループ名>
r オプションには、Ops Center Automator 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 Automator cluster の場合は、"Automator cluster"と指定します。
3. 次のコマンドを使用して、Ops Center Automator の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。

＜共通コンポーネントのインストールフォルダ＞¥ClusterSetup

¥hcmds64clustersrvupdate /sdel /r <グループ名>

r オプションには、Ops Center Automator 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 Automator cluster の場合は、"Automator 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 Ops Center Automator のサービスが削除されます。

4. 次のコマンドを使用して、共通コンポーネントおよび Ops Center Automator のサービスを停止します。
＜共通コンポーネントのインストールフォルダ＞¥bin¥hcmds64srv /stop
5. アクティブノードから Ops Center Automator をアンインストールします。
6. 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。
7. クラスタ管理ソフトウェアで、Ops Center Automator 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 [移動] を選択してから、[ノードを選択] または [このサービスまたはアプリケーションを別のノードに移動] を選択します。
8. スタンバイノードから Ops Center Automator をアンインストールします。
9. クラスタインストールの削除を実行した後、Ops Center Automator フォルダを削除して、スタンバイノードから Base64 フォルダも削除します。
10. 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
11. 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。
12. クラスタ管理ソフトウェアで、Ops Center Automator 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 [移動] を選択してから、[ノードを選択] または [このサービスまたはアプリケーションを別のノードに移動] を選択します。
13. Ops Center Portal から Ops Center Automator の情報を削除します。

5.3 Ops Center Automator をアンインストールする (Linux)

Linux 環境で Ops Center Automator をアンインストールするには、次の手順に従います。

操作手順

1. root ディレクトリ (/root など) に移動します。
2. 次のコマンドを実行します。
 <Ops-Center-Automator のインストールディレクトリ>/ADUninstall/
 uninstall.sh
3. Ops Center Portal から Ops Center Automator の情報を削除します。



Ops Center Automator のファイルの場所 とポート

ここでは、Ops Center Automator のインストール時に作成されるすべてのフォルダ、および、ポートの設定を一覧で説明します。

- [A.1 Ops Center Automator のファイルの場所](#)
- [A.2 ポート設定](#)

A.1 Ops Center Automator のファイルの場所

インストール先フォルダ

次の表は、Ops Center Automator 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。「Windows フォルダの場所」列、または「Linux ディレクトリの場所」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¥hitachi¥Automation
コマンドファイル	system-drive¥Program Files¥hitachi¥Automation¥bin
構成ファイル	system-drive¥Program Files¥hitachi¥Automation¥conf
サービステンプレートのフォルダ	system-drive¥Program Files¥hitachi¥Automation¥contents
データファイル	system-drive¥Program Files¥hitachi¥Automation¥data
ヘルプファイル	system-drive¥Program Files¥hitachi¥Automation¥docroot
事前設定プロパティ定義ファイル	system-drive¥Program Files¥hitachi¥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¥hitachi¥Automation¥inst
ライブラリファイル	system-drive¥Program Files¥hitachi¥Automation¥lib
ログファイル	system-drive¥Program Files¥hitachi¥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¥hitachi¥Automation¥ossSource
システムファイル	system-drive¥Program Files¥hitachi¥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¥hitachi¥Automation¥webapps
作業用フォルダ	system-drive¥Program Files¥hitachi¥Automation¥work
共通コンポーネント	system-drive¥Program Files¥hitachi¥Base64

Linux ディレクトリの詳細	Linux ディレクトリの場所
インストール先ディレクトリ	/opt/hitachi/Automation
コマンドファイル	/opt/hitachi/Automation/bin
構成ファイル	/opt/hitachi/Automation/conf
サービステンプレートのディレクトリ	/var/opt/hitachi/Automation/contents
データファイル	/var/opt/hitachi/Automation/data
ヘルプファイル	/opt/hitachi/Automation/docroot
事前設定プロパティ定義ファイル	/var/opt/hitachi/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ディレクトリ	/opt/hitachi/Automation/inst

Linux ディレクトリの詳細	Linux ディレクトリの場所
ライブラリファイル	/opt/hitachi/Automation/lib
ログファイル	/var/opt/hitachi/Automation/logs
オープンソースソフトウェアのソースファイル	/opt/hitachi/Automation/ossSource
システムファイル	/opt/hitachi/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/hitachi/Automation/work
共通コンポーネント	/opt/hitachi/Base64

A.2 ポート設定

Ops Center Automator は、以下のポートを使用します。

外部接続ポート

ポート番号	ファイアウォール	説明
22/tcp	Automator ↔ 操作対象	SSH に使用されます。 cjstartsv は、このポートを使用します。
23/tcp	Automator ↔ 操作対象	Telnet に使用されます。 cjstartsv は、このポートを使用します。
443/tcp	Automator ↔ Common Services	Common Services へのアクセスに使用されます。
445/tcp または udp	Automator ↔ 操作対象	共有管理に使用されます。 cjstartsv は、このポートを使用します。
135/tcp および 139/tcp	Automator ↔ 操作対象	共有管理に使用されます。 cjstartsv は、このポートを使用します。
22015/tcp	ブラウザ → Automator	HBase 64 Storage Mgmt Web Service へのアクセスに使用されます。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 → Automator	HBase 64 Storage Mgmt Web Service へのアクセスに使用されます。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。
25/tcp	Automator → 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。

ポート番号	ファイアウォール	説明
		cjstartsv は、このポートを使用します。
88/tcp または udp	Automator → Kerberos サーバ	cjstartsv は、このポートを使用します。
359/tcp	Automator → LDAP ディレクトリサーバ	ldap/tls に使用されます。 cjstartsv は、このポートを使用します。
636/tcp	Automator → 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。 cjstartsv は、このポートを使用します。
1812/udp	Automator → Radius サーバ	Radius サーバに使用されます。 cjstartsv は、このポートを使用します。
さまざまな Web サービス接続ポート/tcp	Automator → さまざまなサーバ	Web サービス接続に登録されているサーバに使用されます。

内部接続ポート

ポート番号	ファイアウォール	説明
22017/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22018/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22025/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22026/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22031/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22032/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22035/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22036/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22037/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv 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
22038/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22170/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22171/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22172/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22173/tcp	Automator → Automator	共通コンポーネントへのアクセスに使用されます。 cjstartsv は、このポートを使用します。
22220/tcp	Automator → Automator	組み込みデータベースで使用されます。



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。



Ops Center Automator のプロセス

ここでは、Ops Center Automator のプロセスを一覧で説明します。

- [B.1 プロセス一覧 \(Windows\)](#)
- [B.2 プロセス一覧 \(Linux\)](#)

B.1 プロセス一覧（Windows）

Windows の場合のプロセス一覧を次の表に示します。

この表には Ops Center Automator の状態を確認する場合に必要なプロセス情報を記載しています。Ops Center Automator のプロセス構成を表にしたものではありません。

プロセス	対応サービス名	説明
cjstartsv.exe	HAutomation Engine Web Service	共通コンポーネントで使します。
hcmdssvctl.exe		
cjstartsv.exe	HBase 64 Storage Mgmt SSO Service	共通コンポーネントで使します。
hcmdssvctl.exe		
httpd.exe	HBase 64 Storage Mgmt Web Service	共通コンポーネントで使します。
rotatelog.exe		
httpd.exe	HBase 64 Storage Mgmt Web SSO Service	共通コンポーネントで使します。
rotatelog.exe		
pdsvr.exe	HiRDB/EmbeddedEdition _HD1	共通コンポーネントのデータベースで使します。
pdprcd.exe		
pdmlgd.exe		
pdrdmd.exe		

B.2 プロセス一覧（Linux）

Linux の場合のプロセス一覧を次の表に示します。

この表には Ops Center Automator の状態を確認する場合に必要なプロセス情報を記載しています。Ops Center Automator のプロセス構成を表にしたものではありません。

プロセス	対応サービス名	説明
cjstartsv	hicommand64-hcs_ao	共通コンポーネントで使します。
hcs_ao		
cjstartsv	hicommand64-hcs_hssso	共通コンポーネントで使します。
hcs_hssso		
httpd	hicommand64-hcs_web	共通コンポーネントで使します。
rotatelog		
httpd	hicommand64-hcs_hweb	共通コンポーネントで使します。
rotatelog		
pdprcd	--	共通コンポーネントのデータベースで使します。
pdmlgd		
pdrdmd		



トラブルシューティング

ここでは、Ops Center Automator サーバでエラーが発生した場合の対処方法について説明します。
メッセージまたはログファイルを確認してエラーの原因を特定し、それに応じて対処してください。

- [C.1 保守情報を収集する](#)
- [C.2 ログファイルを収集する](#)

C.1 保守情報を収集する

問題が発生してもメッセージが出力されない場合、またはメッセージの指示に従っても問題を修正できない場合は、保守情報を収集してからユーザーサポートに連絡してください。

C.2 ログファイルを収集する

hcmds64getlogs コマンドを実行して、ログファイルを収集します。

操作手順

1. 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、管理サーバにログインします。
2. **hcmds64getlogs** コマンドを実行して、ログファイルを収集します。

Windows の場合：

```
<共通コンポーネントのインストールフォルダ>%bin%hcmds64getlogs /dir 出力フォルダのパス
```

Linux の場合：

```
<共通コンポーネントのインストールディレクトリ>/bin/hcmd64getlogs -dir 出力ディレクトリのパス
```

操作結果

アーカイブファイルが指定先に出力されます。

『*Hitachi Ops Center Automator ユーザーズガイド*』の「**hcmds64getlogs** コマンド」を参照してください。

索引

A

auditlog.conf
 サンプル 60
 設定 58

C

cssslsetup コマンド 43

H

Hitachi Ops Center 製品 18

I

IPv6 74
IP アドレス
 変更する 41

J

Java ヒープメモリサイズ 81

L

Linux 88

O

Ops Center Automator
 アンインストールする 86, 88
 インストールする 24, 30
 関連製品 18
 基本的なシステム構成 18
 セキュリティ設定 43
 ワークフロー 19
Ops Center Automator のファイルの場所 89
Ops Center Automator のプロセス 95

Ops Center Automator をアンインストールする 85
Ops Center Automator をインストールする 21

S

SSL 43
 Web ベースの管理クライアントでセットアップする 52
 セキュアなクライアント通信のためにサーバ上でセットアップする 44
 セキュアなクライアント通信のために使用 44

U

URL
 管理サーバの URL を変更する 42
 変更する (Linux) 31
 変更する (Windows) 31

W

Windows 24, 77, 86

あ

アンインストールする 86, 88

い

インストール 24
インストール後のタスク 31
インストールする
 Common Services を別のホストに移動する 63
 Ops Center Automator 24, 30
 Ops Center Automator を別のホストに移動する 62
 別のホスト 62, 63
 ポートの衝突を回避する 23
インストールの前提条件 22
インストールを確認する 32

う

- ウイルススキャン
 - 抑止する 30
- ウイルススキャンを抑止する
 - データベースフォルダ 30

え

- エージェントレス 77

か

- 概要 17
 - 関連製品 18
 - 基本的なシステム構成 18
 - ワークフロー 19
- 監査ログ 56
- 監査ログ (Ops Center Automator サーバ)
 - 環境設定ファイルの設定 58
 - 出力形式 60
- 管理クライアント
 - SSL をセットアップする 52
 - 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 44

く

- クラスタ 24
 - インストールの前提条件 25
- クラスタ環境構成を確認する 27

こ

- 構成する
 - 管理サーバの URL 42
 - 基本的なシステム 18
 - サーバの IP アドレス 41
 - サーバのホスト名 41

さ

- サーバ 77

し

- シングルサインオン 33

せ

- セキュア通信 43

セキュリティ設定

- Web ベースの管理クライアントで SSL をセットアップする 52
- 概要 43
- 管理クライアントのセキュア通信 44
- セキュアなクライアント通信のためサーバ上でセットアップする 44
- 前提条件 77

そ

- ソフトウェアをアンインストールする
 - アンインストール手順 86

て

- 定義ファイル 71

と

- 登録する
 - Ops Center Automator 33
- トラブルシューティング 97

な

- 名前解決 23

に

- 認証
 - メソッド 20
 - 外部 83

は

- はじめに 11
- パフォーマンスモード 70

ふ

- ファイルの場所 90
- プランニング
 - ポートの衝突を回避する 23
- プロパティ 74
- プロパティファイル (config_user.properties) 63

ほ

- ポート
 - 衝突を回避する 23
 - ポートを変更したときに更新を必要とするプロパティ 39
- ポート設定 91
- 保守
 - 情報 98
- ホスト 71, 77
- ホスト名
 - 変更する 41

ま

- マシン 77
- マニュアルの構成 12

め

- メール通知 71
- メール通知の構成 71

ゆ

- ユーザー管理 83

ら

- ライセンスを登録する 32

り

- リモート接続情報 74
- リモートマシン用接続情報 74

ろ

- ログファイル
 - 収集する 98
- ログファイルを収集する 98

わ

- ワークフロー
 - 概要 19

