



IT Automation

オフラインインストール

※本書では「Exastro IT Automation」を「ITA」として記載します。

第1.5版

Exastro developer

目次

- 1. はじめに
 - 1.1 本資料について
- 2. システム構成
 - 2.1 環境構築
 - 2.2 動作環境・条件 (1/3)
 - 2.3 動作環境・条件 (2/3)
 - 2.4 動作環境・条件 (3/3)
- 3. ITA環境構築手順
 - 3.1 オフラインインストール
 - 3.2 事前準備
 - 3.3 ITA環境構築フロー
 - 3.4 環境構築 (1/8)
 - 3.5 環境構築 (2/8)
 - 3.6 環境構築 (3/8)
 - 3.7 環境構築 (4/8)
 - 3.8 環境構築 (5/8)
 - 3.9 環境構築 (6/8)
 - 3.10 環境構築 (7/8)
 - 3.11 環境構築 (8/8)
- 4. ITA動作確認
 - 4.1 動作確認 (1/6)
 - 4.2 動作確認 (2/6)
 - 4.3 動作確認 (3/6)
 - 4.4 動作確認 (4/6)
 - 4.5 動作確認 (5/6)
 - 4.6 動作確認 (6/6)
 - 4.7 参考

1. はじめに

1.1 本資料について

■ 本資料について

- 本資料では、ITAサーバをオフライン環境でご利用いただく場合の、構築手順について記載しています。



2. システム構成

2.1 連携実行機能

連携実行機能について

- ITAでは、さまざまな構築ツール等と連携することができ、以下のツールとの連携機能をサポートしています。

| ドライバ名 | 連携ツール (オーケストレーター) | 機能 | 内容 | 環境構築ツール での連携ツール インストール | 本資料での インストール 対象 |
|----------------|----------------------|----------|---|------------------------------|-----------------------|
| Material | git | 構築資材管理 | 標準構築ツールに登録された資材の払出 / 払戻と、Gitを介して対象資材のバージョン管理を行います。 | ○ | × |
| Create_param | - | メニュー作成 | メニューを作成・管理します。 | - | ○ |
| Hostgroup | - | ホストグループ | ホスト群を論理的な単位（機能・役割）でまとめたグループにして、投入するパラメータを管理します。 | - | ○ |
| Ansibleドライバー | Ansible | システム構築 | Red Hat社が提供するOSSのPF構築ツールです。 Playbookと呼ばれる構築コードをもとに、ネットワークで接続された機器に対して、ソフトウェアのインストール、各種設定、ファイル転送、パッチの適用などを行います。 | ○ | ○ |
| | AnsibleTower | システム構築 | PF構築自動化ツールであるAnsibleにアクセスコントロール、ジョブスケジューリング、タスクの可視化などの機能を拡張した管理プラットフォームです。 | × | |
| Cobblerドライバー | Cobbler | OSインストール | OSSのインストール自動化ツールです。 あらかじめ作成したテンプレートを元に、ネットワークで接続された機器に対して、OSのインストールを行うことができます。 | × | × |
| OpenStackドライバー | OpenStack | 仮想システム構築 | OSSのクラウド環境構築ツールです。 クラウド環境に対して、仮想マシン、ストレージ、ネットワークなどを構築することができます。 | × | × |
| Terraformドライバー | Terraform | システム構築 | HashiCorp社が提供するインフラストラクチャを効率化するオーケストレーションツールです。 HCL(HashiCorp Configuration Language)という言語でコード化したインフラストラクチャ構成について、実行計画を生成したうえで構築を実行します。また、Policy as Codeによるアクセスポリシーをコード化して管理することが可能です。 | × | × |

■ ITAをご利用いただくための環境について

- 「Exastro-ITA_システム構成／環境構築ガイド_基本編」を参照してください。



■ ライブラリ収集スクリプトを実行する場合の前提条件

- ライブラリ収集スクリプトを実行する場合には、ライブラリ収集用サーバ（オンライン環境） / ITAサーバ（オフライン環境）、両サーバの構築状態（OSのバージョン、インストール済のパッケージ）を、合わせる必要があります。
- ライブラリ収集用サーバ（オンライン環境）は、以下のリポジトリが参照できる状態である必要があります。

（※次頁に記載）



2.4 動作環境・条件 3/3

● 参照するリポジトリ一覧

| OS | リポジトリ |
|-----------|---|
| RHEL7 | https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm |
| | https://downloads.mariadb.com/MariaDB/mariadb_repo_setup |
| | http://rpms.remirepo.net/enterprise/remi-release-7.rpm |
| | rhel-7-server-optional-rpms |
| RHEL8 | https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm |
| | codeready-builder-for-rhel-8-xxxxxx-rpms |
| CentOS7 | epel-release |
| | https://downloads.mariadb.com/MariaDB/mariadb_repo_setup |
| | http://rpms.remirepo.net/enterprise/remi-release-7.rpm |
| CentOS8 | epel-release |
| | PowerTools |
| RHEL7_AWS | https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm |
| | https://downloads.mariadb.com/MariaDB/mariadb_repo_setup |
| | http://rpms.remirepo.net/enterprise/remi-release-7.rpm |
| | rhui-rhel-7-server-rhui-optional-rpms |
| RHEL8_AWS | https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm |
| | codeready-builder-for-rhel-8-rhui-rpms |

※RHEL7_AWS : AWS上のRHEL7

RHEL8_AWS : AWS上のRHEL8

xxxxxx : アーキテクチャ

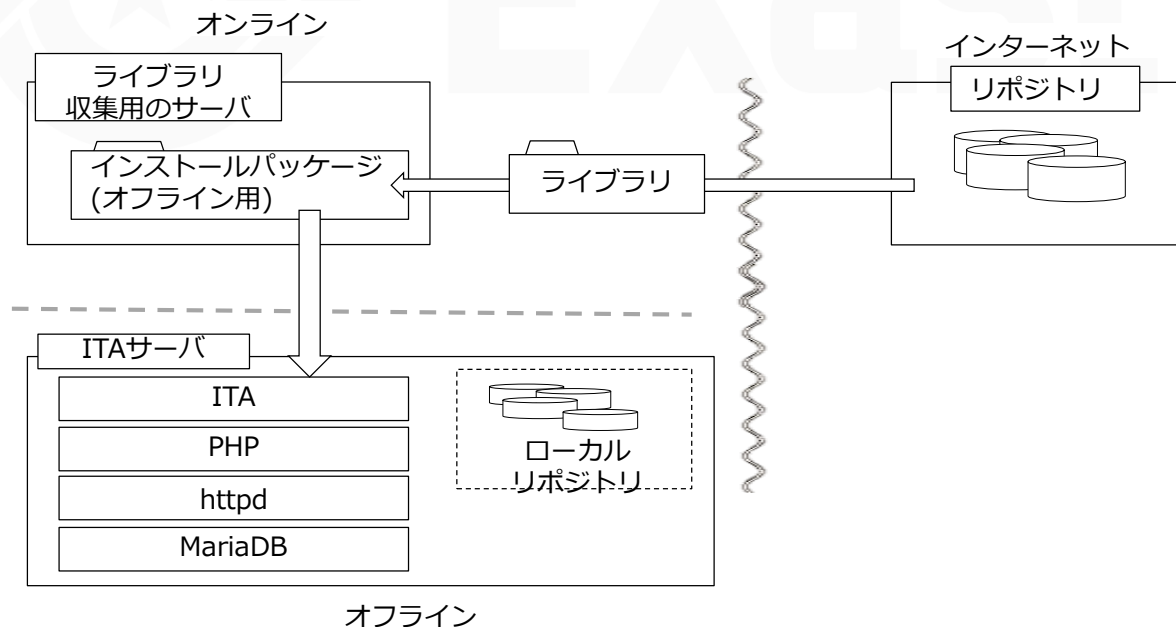
3. ITA環境構築手順

3.1 オフラインインストール

インストール手順について

ITAサーバがオフライン環境の場合、以下の手順で環境構築を行います。

- ライブラリ収集用のサーバ(オンライン)にてインターネット経由で必要なライブラリを収集し、ITAインストールパッケージとライブラリを一つに圧縮し、インストールパッケージ(オフライン用)を作成します。
- 記憶媒体等でITAサーバに、インストールパッケージ(オフライン用)を移動します。
- インストールパッケージ(オフライン用)からローカルリポジトリを作成し、必要なライブラリのインストールと、ITAインストーラーの実行を行います。



3.2 事前準備

ITA環境構築ツール一覧

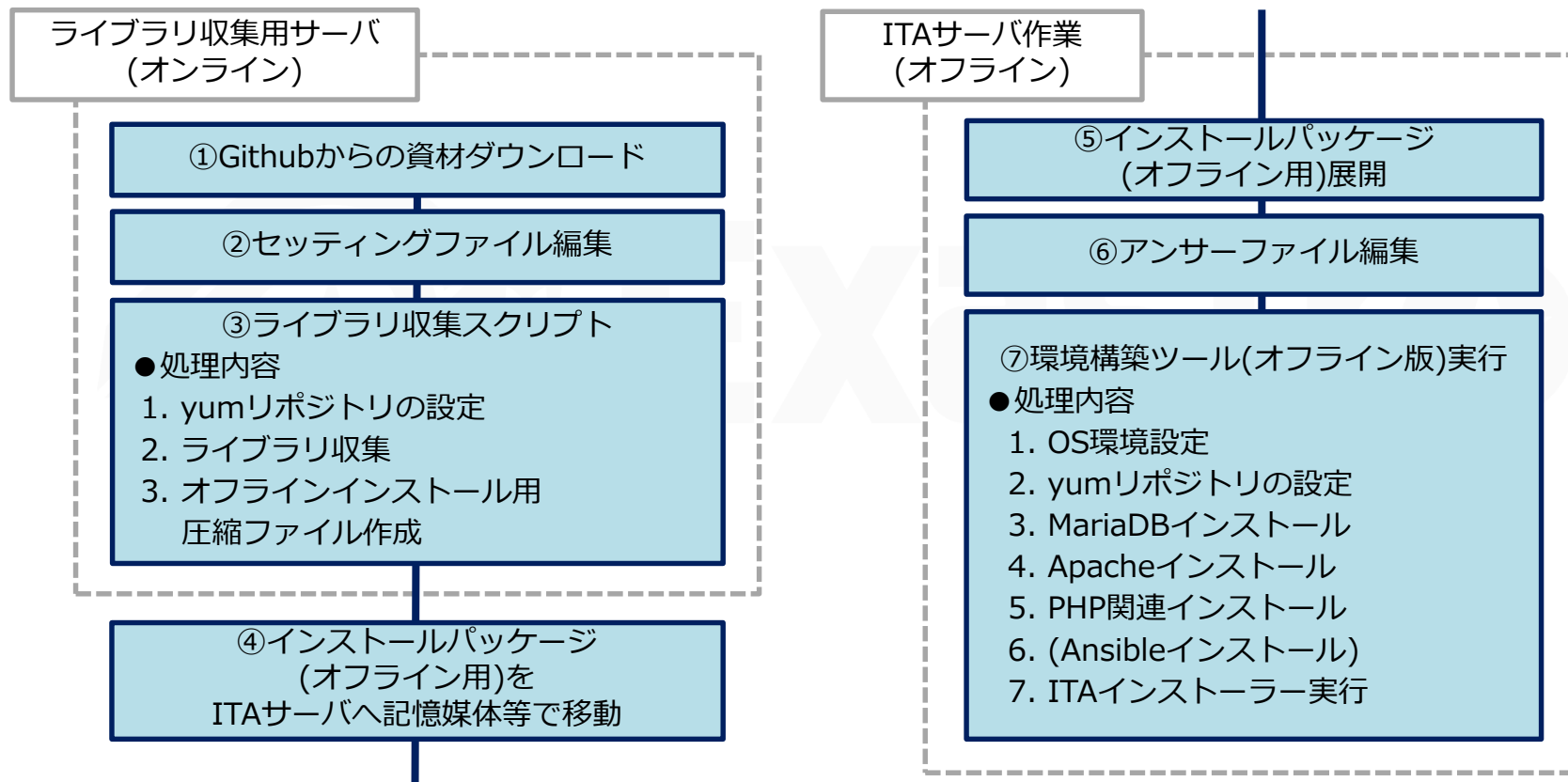
- ITA環境構築ツール一覧は以下となります。

| 説明 | ファイル | 格納先 |
|---------------------------|-------------------------|---|
| ライブラリ収集スクリプト | ita_gather_library.sh | /(インストール資材展開先)/ita_install_package/install_scripts/ |
| 環境構築ツール (オフラインインストール用) | ita_builder_offline.sh | /(インストール資材展開先)/ita_install_package/install_scripts/ |
| 環境構築ツール (オンラインインストール用) | ita_builder_online.sh | /(インストール資材展開先)/ita_install_package/install_scripts/ |
| セッティングファイル | ita_builder_setting.txt | /(インストール資材展開先)/ita_install_package/install_scripts/ |
| ITAインストーラー | ita_installer.sh | /(インストール資材展開先)/ita_install_package/install_scripts/ |
| アンサーファイル | ita_answers.txt | /(インストール資材展開先)/ita_install_package/install_scripts/ |

3.3 ITA環境構築フロー

環境構築フロー（オフライン）

- 環境構築は以下のフローとなっています。



3.4 環境構築 (1/8)

※オンライン環境で実施します

※環境構築ユーザーはrootユーザーで実施すること。

Githubからの資材ダウンロード

- 以下のコマンドで資材をDLします。

```
# wget https://github.com/exastro-suite/it-automation/archive/vx.x.x.tar.gz
```

※wgetコマンドは事前にインストールしてください。

※バージョン(x.x.x)は適宜変更してください。

資材の展開

- .tar.gz ファイルを解凍します。

```
# tar xzf vx.x.x.tar.gz
```

ディレクトリ移動

- 環境構築を設定を行うセッティングファイルとシェルのあるディレクトリに移動します。

```
# cd it-automation-x.x.x/ita_install_package/install_scripts
```

3.5 環境構築 (2/8)

■ セットティングファイル編集

- ITA環境構築を設定を行うセットティングファイル (ita_builder_setting.txt) の編集方法を以下に示します。

| 種目 | 必須 | 初期値 | 説明 |
|----------------------|--|-----|--|
| linux_os | ○ | - | ITAサーバのOS ("CentOS7","CentOS8","RHEL7","RHEL8", "RHEL7_AWS", "RHEL8_AWS") *)RHEL7_AWS : AWS上のRHEL7 RHEL8_AWS : AWS上のRHEL8 |
| redhat_user_name | ITAサーバのOSがRHEL7またはRHEL8 の場合必須 ※AWS上のRHELの場合は不要 | - | redhatアカウントのユーザー名 |
| redhat_user_password | | - | redhatアカウントのパスワード |
| pool_id | | - | redhatアカウントのプールID |

3.6 環境構築 (3/8)

■ セットアップファイル(ita_builder_setting.txt)のサンプル

- セットアップファイル(ita_builder_setting.txt)のサンプルを以下に示します。

```
# Select Operation System. ("CentOS7","CentOS8","RHEL7","RHEL8","RHEL7_AWS","RHEL8_AWS")
# Select Operation System. ("CentOS7","CentOS8","RHEL7","RHEL8")
# e.g) linux_os:RHEL8
linux_os: RHEL7
#####
#Only when you select linux_os with RHEL7 or RHEL8
# Enter the Red Hat user name and user password
# e.g) redhat_user_name:sample
redhat_user_name:sample
# e.g) redhat_user_password:sample_password
redhat_user_password: sample_password
# e.g) pool_id:samplePoolID
pool_id: samplePoolID
```

インストール対象OS : RHEL7

POINT

RHELの場合のみ、
記載してください。

3.7 環境構築 (4/8)

ライブラリ収集スクリプトの実行

- 以下のコマンドで、ライブラリ収集スクリプトを実行します。

```
# sh ita_gather_library.sh
```

処理の確認

- ライブラリ収集スクリプトを実行すると、ita_gather.log に処理内容が出力されます。
- ログ格納パス
/(インストール資材展開先)/ita_install_package/install_scripts/log/

ファイルの移動

- インストールパッケージ（オフライン用）をITAサーバへ記憶媒体等で移動します。

※以降の手順は、ITAサーバ（**オフライン環境**）で実施します

インストールパッケージ（オフライン用）展開

- ITAサーバ上で、インストールパッケージ（オフライン用）を展開します。

```
# tar zxf ita_Verx.x_offline_yyyymmddhhmmss.tar.gz
```

3.8 環境構築 (5/8)

■ アンサーファイル(ita_answers.txt)を編集

- ITAのインストール設定を行うアンサーファイルを事前に作成してください。
- 初期値はita_base、ansible_driver、createparam、Hostgroupのインストール設定がyesとなっています。インストールしない場合は、設定値をnoとしてください。

| 種目 | 必須 | 初期値 | 説明 |
|------------------|----|---------|---|
| install_mode | ○ | Install | インストールモードの設定 (インストール(Install)/アンインストール(Uninstall)) |
| ita_directory | ○ | – | インストールディレクトリ ITAをインストールするディレクトリを絶対パスで指定してください。 ディレクトリが無い場合作成されます。 |
| ita_language | ○ | en_US | ITA画面表示の言語 (日本語 (ja_JP) / 英語 (en_US)) |
| ita_os | ○ | RHEL7 | ITAのOS(RHEL7 系の場合は(RHEL7)/ RHEL8 系の場合は(RHEL8)) |
| db_root_password | ○ | – | MariaDBのrootパスワード |
| db_name | ○ | – | MariaDBのDB名 |
| db_username | ○ | – | MariaDBのDBユーザー名 |
| db_password | ○ | – | MariaDBのDBパスワード |
| ita_base | ○ | yes | ITA本体のインストール (“yes”のみ) |
| Material | ○ | no | 構築資材管理機能のインストール有無指定 |
| create_param | ○ | yes | メニュー作成機能のインストール有無指定 |
| Hostgroup | ○ | yes | ホストグループ機能のインストール有無指定 |
| ansible_driver | ○ | yes | Ansible driverのインストール有無指定 |
| cobbler_driver | ○ | no | Cobbler driverのインストール有無指定 |
| openstack_driver | ○ | no | OpenStack driverのインストール有無指定 |
| terraform_driver | ○ | no | Terraform driverのインストール有無指定 |

3.9 環境構築 (6/8)

■ アンサーファイル(ita_answers.txt)のサンプル

- アンサーファイル(ita_answers.txt)のサンプルを以下に示します

```
#Select install mode. ("Install" or "Uninstall")
# e.g) install_mode:Install
install_mode:Install

#Enter install directory.
# e.g) ita_directory:/exastro
ita_directory:/exastro

# Select language. ("en_US" or "ja_JP")
# e.g) ita_language:en_US
ita_language:ja_JP

# Select Operation System. ("RHEL7" or "RHEL8")
# e.g) ita_os:RHEL8
ita_os:RHEL7

# Enter the MariaDB root user's password
# e.g) db_root_password:sample_root_password
db_root_password:sample_root_password

# Decide the database name, username, and password for ITA.
# e.g) db_name:sample_db_name
db_name:sample_db_name
# e.g) db_username:sample_db_username
db_username:sample_db_username
# e.g) db_password:sample_db_password
db_password:sample_db_password

# Select the target you need to install.
# yes : need
# no  : no need
ita_base:yes
material:no
createparam:yes
hostgroup:yes
ansible_driver:yes
cobbler_driver:no
openstack_driver:no
terraform_driver:no
```

POINT

MariaDBのパスワードは、アンサーファイルで定義します。

※パスワードに記号を含めるとエラーになる場合があります。

3.10 環境構築 (7/8)

環境構築ツール(オフライン版)実行

- 以下のコマンドで、環境構築ツールを実行します。

```
# sh ita_builder_offline.sh
```

処理の確認

- 環境構築ツールを実行すると、ita_builder.log、ita_installer.log に処理内容が出力されます。
- ログ格納パス
/(インストール資材展開先)/ita_install_package/install_scripts/log/

3.11 環境構築（8/8）

環境構築時にインストールされるライブラリの一覧

- 環境構築ツールを実行することでインストールされるライブラリは、以下となります。

| インストールドライバ | ライブラリ概要 | ライブラリ名 |
|----------------|-----------|---|
| ita_base | インストールツール | yum-utils(*), createrepo(*) |
| ita_base | ITA共通 | zip, telnet, mailx, unzip, sudo, crontabs |
| ita_base | MariaDB | MariaDB, MariaDB-server, expect |
| ita_base | httpd | httpd, mod_ssl |
| ita_base | php | php, php-bcmath, php-cli, php-ldap, php-mbstring, php-mcrypt, php-mysqlnd, php-pear, php-pecl-zip, php-process, php-snmp, php-xml, php-json, php-gd, Python3, php-devel, libyaml, libyaml-devel, make |
| ita_base | phpプラグイン | php-yaml, HTML_AJAX-beta, PhpSpreadsheet |
| material | git | Git |
| ansible_driver | ansible | ansible, sshpass, pexpect, pywinrm, boto3, nmap-ncat, paramiko |

※ RHEL7,CentOS7

4. ITA動作確認

4.1 動作確認 (1/6)

メインメニューの表示による確認

- インストール処理終了後、自端末のWindowsPCから下記の手順により、ITAシステムメインメニューにアクセスし、ITA本体、各ドライバーが正常に表示されたことを確認してください。

URL接続

- 以下のURLより、ログイン画面にアクセスしてください。
- URL : [http:// \(サーバのIPアドレス\)](http://(サーバのIPアドレス))

※インストール後は、HTTPとHTTPSの両方のアクセスが可能です。
HTTPはセキュリティ的に脆弱なので、HTTPSでのアクセスを推奨します。
HTTPSでのアクセス方法は、動作確認 (4/6) 以降を確認してください。

ログイン

- ITAのログイン画面が表示されたら、指定のログインID、初期パスワードを入力して、[ログイン]ボタンをクリックしてください。
 - ・ ログインID : administrator
 - ・ 初期パスワード : password
- インストール後に初めてログインした場合は、「パスワード変更画面」に遷移します。
- パスワード変更画面から、初期パスワードを変更してください。

4.2 動作確認 (2/6)

ITAログイン画面

- 正常にインストールされている場合、以下のようなログイン画面が表示されます。



4.3 動作確認 (3/6)

各メニューの表示による内容確認

- ログイン後、以下のメニューが正常に表示されることを確認してください。

| 機能 | メニュー |
|---------------|--------------------|
| ITA本体 | 管理コンソール |
| | 基本コンソール |
| | エクスポート/インポート |
| | Symphony |
| | Conductor |
| メニュー作成 | メニュー作成 |
| ホストグループ | ホストグループ管理 |
| Ansible ドライバー | Ansible共通 |
| | Ansible-Legacy |
| | Ansible-Pioneer |
| | Ansible-LegacyRole |

4.4 動作確認 (4/6)

HTTPSでアクセスするための準備作業

- 操作端末 (Windows) のhostsファイルへITA実装サーバのIPアドレスとホスト名を設定してください。
- Windows10の場合、以下のhostsファイルとなります。

```
C:¥Windows¥System32¥drivers¥etc¥hosts
```

- hostsファイルに以下の設定を追加してください。

```
"ITA実装サーバのIPアドレス" exastro-it-automation
```

例)

```
192.168.0.3 exastro-it-automation
```

4.5 動作確認 (5/6)

- 操作端末 (Windows) への証明書インポートを行います。
証明書はITAインストールパッケージの以下のパスに格納されています。
- FFFTP、WinSCPなどのツールを利用し操作端末にダウンロードします。

| ITAサーバのOS | ファイルパス | ファイル名 |
|-------------------|---|---------------------------|
| RHEL 7, CentOS 7系 | /(インストール資料展開先)/ita_install_package/ext_files_for_CentOS7.x/etc_pki_tls_certs/ | exastro-it-automation.crt |
| RHEL 8, CentOS 8系 | /(インストール資料展開先)/ita_install_package/ext_files_for_CentOS8.x/etc_pki_tls_certs/ | exastro-it-automation.crt |

- Webブラウザに証明書のインポートをしてください。
- 例として、Google Chromeにインポートする手順を以下に示します。
 - ① Google Chromeを起動し、右上の[設定]ボタンから[設定(S)]へ進みます。
 - ② 画面下部の[詳細設定]をクリック後、表示される項目から[証明書の管理]をクリックします。
 - ③ [信頼されたルート証明機関]タブへ進み、左下の[インポート]をクリックします。
 - ④ 証明書のインポートウィザードが起動します。[次へ]をクリックします。
 - ⑤ インポートするファイル名を指定し、[次へ]をクリックします。
 - ⑥ [証明書をすべての次のストアに配置する(P)]をチェックされている状態を確認します。
 - ⑦ [信頼されたルート証明機関]を選択し、[次へ]をクリックします。
※選択されていない場合は右の[参照(R)]から[信頼されたルート証明機関]を選択してください。
 - ⑧ [完了]をクリックします。

4.6 動作確認 (6/6)

■ HTTPSでのURL接続

- 以下のURLより、ログイン画面にアクセスしてください。

- URL : **<https://exastro-it-automation>**

※ホスト名の代わりに、サーバーのIPアドレスでアクセスすることも可能です。

接続後はHTTPの場合と同様となります。



■ HTTPまたはHTTPSアクセスの制限

HTTPまたはHTTPSアクセスを制限する場合は、以下の手順を実施してください。

- ファイル「/etc/httpd/conf.d/vhosts_exastro-it-automation.conf」を編集する。
HTTPを制限する場合は、「<VirtualHost *:80 >」から「</VirtualHost>」をコメントアウト(#)をする。
HTTPSを制限する場合は、「<VirtualHost *:443 >」から「</VirtualHost>」をコメントアウト(#)をする。
- 以下のコマンドによりApacheを再起動する。
`systemctl restart httpd`



Exastro