



IT Automation

オールインワン型HA構成(AWS)

インストールマニュアル

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

ExastroIT Automation Version 1.9

Exastro developer

はじめに

1. 本資料について

- 本資料は、ITAオールインワンインストール済EC2サーバ2台とOSS製品群を組み合わせ、HAクラスタを構成するのに必要な情報を記述したものです。ITAサーバをHAクラスタ化することにより、単一Linuxサーバでは実現できない可用性・信頼性の高いITAサーバを構築することができます。
- 本資料にもとづいて構成したHAクラスタは、以下の機能を提供します。
 - 常にどちらか1台のサーバ上でITAサーバが稼働し、ITAのWeb/AP、Backyard、Ansibleドライバなどの全機能を提供します。
 - サービスを提供しているサーバがダウンすると、約2-3分前後で他のサーバ上でITAサーバが起動し、サービスが引き継がれます。この時、データベースのデータは失われません。
- デプロイ例は「5. システム構成例」の図を参照してください。
ユーザ環境に合わせて、IPアドレス、ホスト名、ディレクトリ名、ファイル名を読み替えて構築してください。
- 本資料は、以下の環境で構築することを前提に作成しています。

主要ソフトウェア(ITAを除く)	用途	検証バージョン
Red Hat with Linux 8 High Availability	OS (EC2のイメージ)	8.4
Pacemaker(※)	クラスタ制御	2.0.5-9
Corosync(※)	ハートビート	3.1.0
pcs(※)	Pacemakerコマンド	0.10.8

※この3つは上記のEC2のイメージに同梱されています

2. 責任範囲

- 本資料は、ITAサーバをクラスタ化する為の注意点や設定例を参考情報として示しています。

はじめに

3. 構築順序

・構築順序は下記のとおりです。

項目	サービス	1号機(Master)	2号機(Slave)
0-1. 事前準備(AWS設定)	VPC	-	-
	サブネット	-	-
	ルートテーブル	-	-
	セキュリティグループ	-	-
	IAM	-	-
	RDS	-	-
	EFS	-	-
	Route53	-	-
	EC2	EC2作成	EC2作成
	ElasticIP	ElasticIP関連付け	-
1. ITAインストール	ITA1.7.2オンラインインストール	ITAオンラインインストール	ITAオンラインインストール
2. Apacheリソース設定	Apache	Apacheステータス確認リソース作成	Apacheステータス確認リソース作成
		Apacheサービス停止・無効化	Apacheサービス停止・無効化
3. ITAサービス停止設定	-	ITAサービス停止・無効化	ITAサービス停止・無効化
4. EFS設定	EFS	EFS使用準備	EFS使用準備
		共有ディレクトリ設定	-
5. RDS設定	RDS	RDSデータ設定	-
		DB接続設定	DB接続設定
		MariaDB停止	MariaDB停止
6. Pacemaker使用準備	-	awscilインストール	awscilインストール
		hostsの設定	hostsの設定
7. Pacemaker設定	Pacemaker	クラスター設定	クラスター設定
		リソース設定	リソース設定
		Pacemaker起動順序設定	Pacemaker起動順序設定
		MariaDBアンインストール	MariaDBアンインストール

はじめに

4. Linux-HAクラスタスタックについて

- Linux-HAクラスタスタックは、サーバシステムの可用性を向上したり、ストレージ故障によるデータ喪失を防止したりすることを目的としたソフトウェア群です。次の2つのソフトウェアおよび関連パッケージによって構成されています。

Corosync

HAクラスタを構成するサーバの正常稼働を相互に監視するソフトウェア。

Pacemaker

HAクラスタが提供するサービスを監視し、サービスを提供していたサーバがダウンした場合に他のサーバでサービスを継続させるなどして、クラスタ全体の可用性を保ちます。

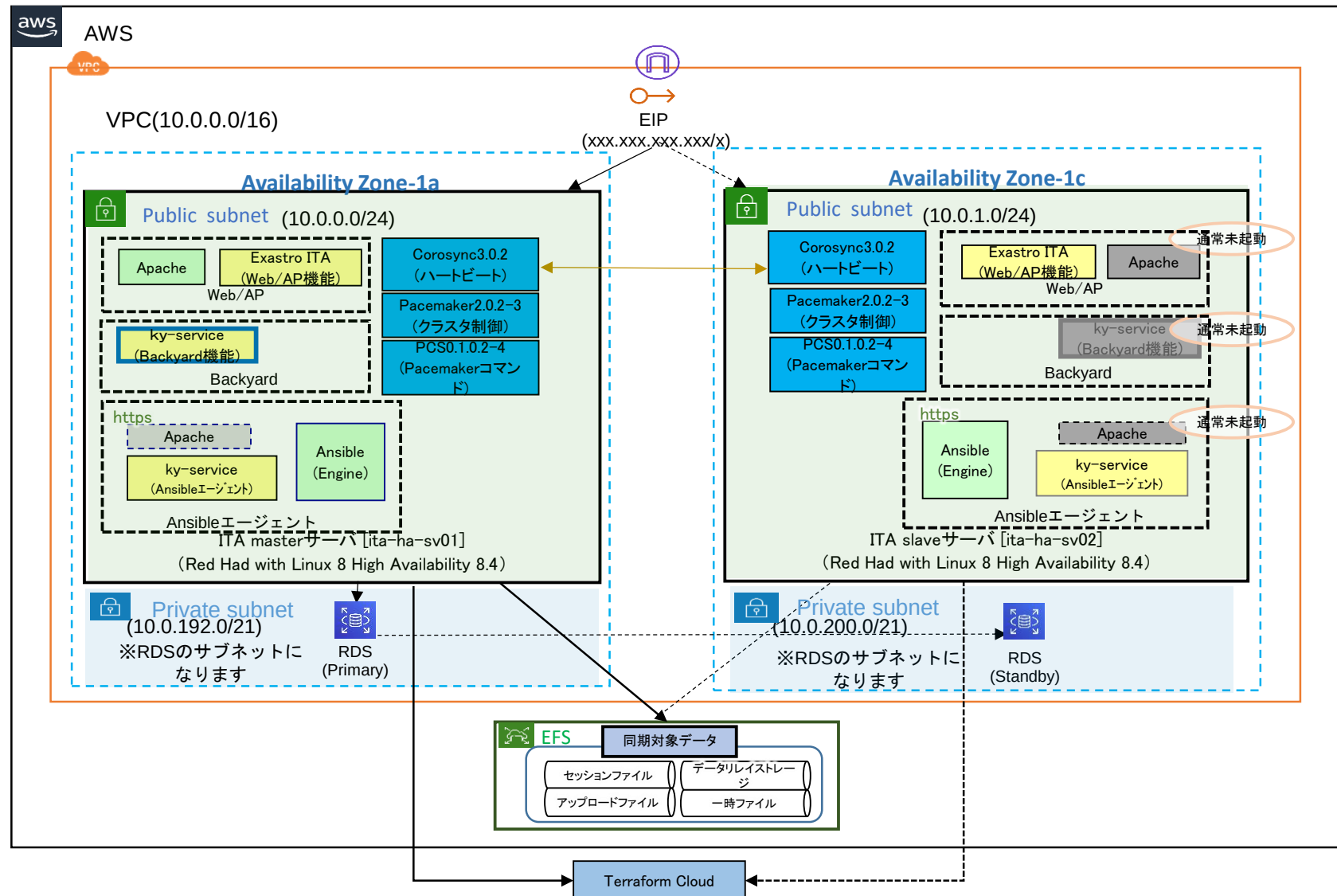
参照URL:

<http://linux-ha.osdn.jp/wp/>

はじめに

5. システム構成例

- ITAクラスタを構成する2台のサーバのディスクやネットワーク構成、関連プログラムは、下図のようになります。



はじめに

- ・ ネットワーク条件

- ・ 通常、各ネットワークインタフェースは、それぞれ固有のIPアドレスを持つが、ITAサーバのHAクラスタ構成を実現する為に追加のIPアドレス(ElasticIPアドレス)が必要になります。用意したElasticIPアドレスは、構築後、Pacemakerにより自動的にmasterサーバ側に割り当てられます

- ・ 本資料のネットワーク設定例

ネットワークIF	用途	masterサーバ	slaveサーバ
ElasticIP	ITA接続	自動的に発行されたグローバルIP	
eth0	ローカルIP	10.0.0.204	10.0.1.61

※masterサーバのローカルIPアドレスは、ElasticIPを関連付ける際に設定するローカルIPアドレスになります。

- ・ 利用クライアントからサーバへ以下の通信ポートが利用可能なこと。

通信種別	ポート番号
http	80/tcp
https	443/tcp

- ・ クラスタ構成サーバを同一ネットワーク上に構成し、以下の通信ポートが利用可能であること。

通信種別	ポート番号
ssh(scp)	22/tcp
MYSQL/Aurora(RDS)	3306/tcp
NFS	2409/tcp
Pacemaker	2224/tcp
Corosync	5405/udp

※Pacemaker、Corosyncポート参考URL:

https://access.redhat.com/documentation/ja-jp/red_hat_enterprise_linux/7/html/high_availability_add-on_reference/s1-firewalls-haar

はじめに

6. 作業前提

- 各種ソフトウェアのインストールはオンライン環境での手順を記載しています。
オフラインで実行する場合はソフトウェアの依存関係を構築サーバと合致させた上で、
事前のライブラリ収集を行ってください。
- 構築手順の前に、以下の基本的なOS設定を実施して下さい。なお、特に断らない限り、OS基本設定は両方のサーバで同一に揃える必要があります。
 - SELinuxの無効化
SELinuxが有効な状態でHAクラスタを正常に運用するには、SELinuxに関する、極めて高度な知識と経験が必要になります。
本手順ではSELinuxを無効での検証となっておりますが、無効化はユーザの自己責任で対応ください。
- ITAのインストールについて
 - ITAの動作要件については、以下のドキュメントを参照してください。
https://exastro-suite.github.io/it-automation-docs/asset/Documents_ja/Exastro-ITAシステム構成／環境構築ガイド基本編.pdf
IT Automation BASE システム構成／環境構築ガイド 基本編
1.1 サーバ動作要件
 - [https://exastro-suite.github.io/it-automation-docs/asset/Documents_ja/Exastro-ITAシステム構成／環境構築ガイド Ansible-driver編.pdf](https://exastro-suite.github.io/it-automation-docs/asset/Documents_ja/Exastro-ITAシステム構成／環境構築ガイドAnsible-driver編.pdf)
Ansible-driver システム構成／環境構築ガイド Ansible-driver編
3. システム要件

はじめに

- ・ インストール要件

以下URLのオールインワン構成のインストールマニュアル参照のこと。
ITAはオンライン／オフラインのどちらのインストール手順でも問題ありません。

https://exastro-suite.github.io/it-automation-docs/learn_ja.html#deploy

本資料では標準インストールの機能有効化を前提としています。

ita_answers.txtの機能設定例

```
ita_base:yes
createparam:yes
hostgroup:yes
ansible_driver:yes
cobbler_driver:no
terraform_driver:yes
cicd_for_iac:no
```

ITAの起動サービスは、ITAのバージョン、インストールする機能によって変動します。
本資料は、ITA ver1.7.xの標準インストールを想定している為、
上記の想定と異なるバージョンの利用と機能のインストールを行う場合は、
本手順の下記の箇所にの対象サービス内容について、適宜修正が必要となります。

- ・ 構築手順「3.ITAサービス停止設定」の対象サービス

AWS設定手順書

ここでは事前準備として、AWSの設定手順を説明していきます。
IPアドレスや設定値は任意で変更をお願いいたします。

今回の利用するサービスは下記のとおりです。

利用サービス一覧
1.VPC
2.サブネット
3.インターネットゲートウェイ
4.ルートテーブル
5.セキュリティグループ
6.IAM
7.RDS
8.EFS
9.Route53
10.EC2
11.ElasticIP

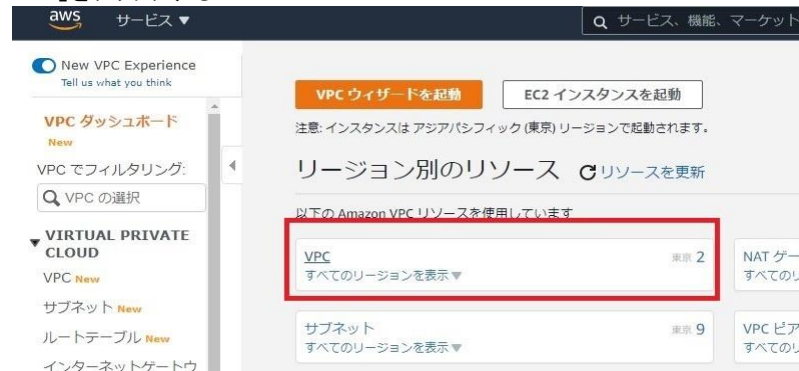
AWS設定手順書

1.VPC

1.コンソールにログインし、左上の「サービス」から「VPC」をクリックする



2.「VPC」をクリックする



3.「VPCを作成」をクリックする



AWS設定手順書

4.VPCの情報を入力する

VPC > お使いの VPC > VPC を作成

VPC を作成 情報

VPC は、Amazon EC2 インスタンスなどの AWS オブジェクトによって使用される AWS クラウドの分離された部分です。

VPC の設定

名前タグ - オプション
「Name」というキーと、指定した値を使用してタグを作成します。

exastro-VPC

IPv4 CIDR ブロック 情報
10.0.0.0/16

IPv6 CIDR ブロック 情報
☒ IPv6 CIDR ブロックなし
☐ Amazon 提供の IPv6 CIDR ブロック
☐ IPv6 CIDR 所有 (ユーザー所有)

テナンシー 情報
デフォルト

タグ

タグは、AWS リソースに割り当てられるラベルです。各タグはキーとオプションの値で構成されています。タグを使用してリソースを検索およびフィルタリングしたり、AWS のコストを追跡したりできます。

キー: Q Name X 値 - オプション: Q exastro-VPC X 削除

新しいタグを追加

さらに 49 個の タグ を追加できます。

キャンセル **VPC を作成**

入力例)

名前タグ	IPv4 CIDRブロック
exastro-VPC	10.0.0.0/16

5.VPCが作成できたことを確認する

VPC > お使いの VPC > vpc-0e755a2dfc566d299 / exastro-VPC

詳細

VPC ID	ステータス	IPv4 CIDR ブロック	IPv6 CIDR ブロック
vpc-0e755a2dfc566d299	Available	10.0.0.0/16	None

タグ

キー	値
exastro-VPC	exastro-VPC

AWS設定手順書

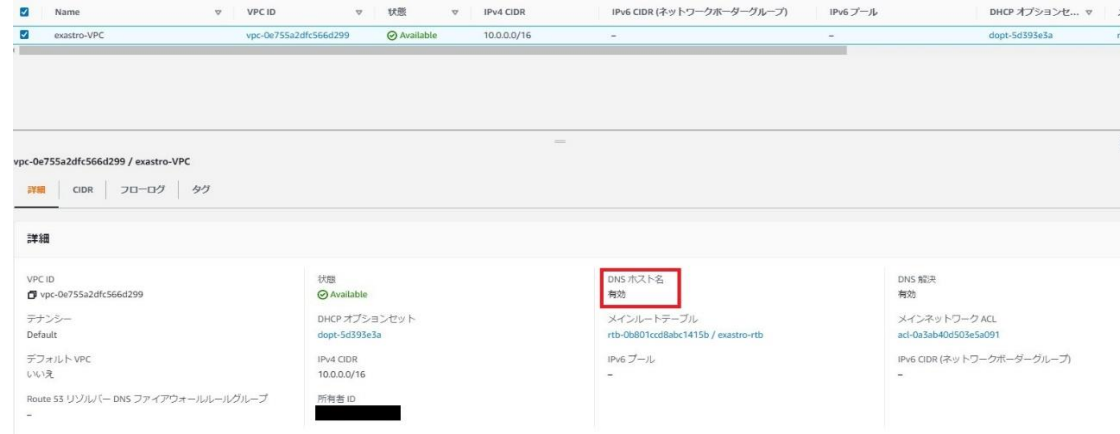
6.2の画面に戻りアクションから「DNSホスト名を編集」をクリックする



7.DNSホスト名を有効化し、「変更を保存」をクリックする

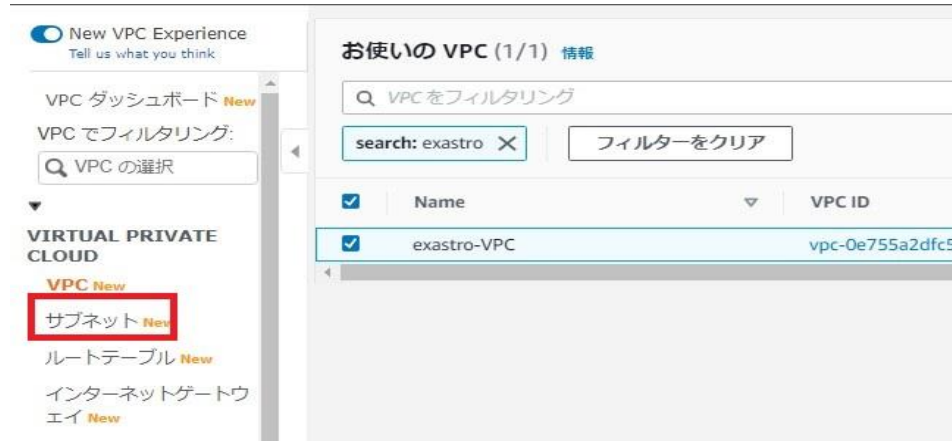


8.DNSホスト名が有効になっていることを確認する



2.サブネット

1.左側のメニューから「サブネット」をクリックする



2.「サブネットを作成」をクリックする



AWS設定手順書

3.先ほど作成したVPCをクリックし、サブネットの情報を入力する

サブネットを作成 [情報](#)

VPC

VPC ID
このVPCにはサブネットを割り当てます。

vpc-Oe755a2dfc566d299 (exastro-VPC) ▼

関連付けられた VPC CIDR

IPv4 CIDR
10.0.0.0/16

サブネットの設定

サブネットの CIDR ブロックとアベイラビリティゾーンを指定します。

サブネット 1 (1 個中)

サブネット名
(Name) というキーと、指定した値を使用してタグを作成します。

public-a

(このVPCのCIDRブロックを参照してください。)

アベイラビリティゾーン [情報](#)
サブネットが存在するゾーンを選択するか、Amazon が選択するゾーンを受け入れます。

アジアパシフィック (東京) / ap-northeast-1a ▼

IPv4 CIDR ブロック [情報](#)
10.0.0.0/24 ✕

▼ タグ - オプション

キー

Q Name ✕

値 - オプション

Q public-a ✕ 削除

新しいタグを追加

さらに 49 個の タグ、を追加できます。

削除

新しいサブネットを追加

キャンセル サブネットを作成

入力例)

サブネット名	アベイラビリティゾーン	IPv4 CIDRブロック
public-a	アジアパシフィック (東京)/ap-northeast-1	10.0.0.0/24
public-c	アジアパシフィック (東京)/ap-northeast-1	10.0.1.0/24
protect-rds-a	アジアパシフィック (東京)/ap-northeast-1	10.0.192.0/21
protect-rds-c	アジアパシフィック (東京)/ap-northeast-1	10.0.200.0/21

AWS設定手順書

サブネット 2 (4 個中)

サブネット名
「Name」というキーと、設定した値を使用してタグを作成します。
public-c

名前のは最大 255 文字です。

アベイラビリティゾーン 情報
サブネットが存在するゾーンを選択するか、Amazon が選択するゾーンを受け入れます。
アジア/パシフィック (東京) / ap-northeast-1c

IPv4 CIDR ブロック 情報
10.0.1.0/24

▼ タグ - オプション

キー
Name

値 - オプション
public-c

削除

新しいタグを追加
さらに 49 個のタグを追加できます。

削除

サブネット 3 (4 個中)

サブネット名
「Name」というキーと、設定した値を使用してタグを作成します。
protect-rds-a

名前のは最大 255 文字です。

アベイラビリティゾーン 情報
サブネットが存在するゾーンを選択するか、Amazon が選択するゾーンを受け入れます。
アジア/パシフィック (東京) / ap-northeast-1a

IPv4 CIDR ブロック 情報
10.0.192.0/21

▼ タグ - オプション

キー
Name

値 - オプション
protect-rds-a

削除

新しいタグを追加
さらに 49 個のタグを追加できます。

削除

サブネット 4 (4 個中)

サブネット名
「Name」というキーと、設定した値を使用してタグを作成します。
protect-rds-c

名前のは最大 255 文字です。

アベイラビリティゾーン 情報
サブネットが存在するゾーンを選択するか、Amazon が選択するゾーンを受け入れます。
アジア/パシフィック (東京) / ap-northeast-1c

IPv4 CIDR ブロック 情報
10.0.200.0/21

▼ タグ - オプション

キー
Name

値 - オプション
protect-rds-c

削除

新しいタグを追加
さらに 49 個のタグを追加できます。

削除

新しいサブネットを追加

キャンセル サブネットを作成

AWS設定手順書

4.作成後、状態がAvailableになっていることを確認する

4件のサブネットが正常に作成されました。 subnet-0d5864ac52e20e1e5, subnet-063695cadeb207c1b, subnet-097fba3cefd69f930, subnet-061dbdda51760b21d

サブネット (4) 情報

サブネットをフィルター

サブネット ID: subnet-0d5864ac52e20e1e5 X

サブネット ID: subnet-063695cadeb207c1b X

サブネット ID: subnet-097fba3cefd69f930 X

サブネット ID: subnet-061dbdda51760b21d X

フィルターをクリア

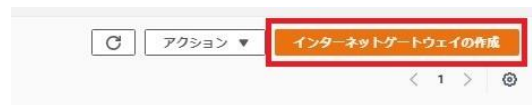
☐	Name	サブネット ID	状態	VPC	IPv4 CIDR	IPv6 CIDR	利用可能な IPv4 アドレス	アベイラビリティゾーン...	アベイラビリティゾーン...	ネットワーク
☐	protect-rds-a	subnet-097fba3cefd69f930	Available	vpc-0e755a2dfc566d299 exa...	10.0.192.0/21	-	2043	ap-northeast-1a	apne1-az4	ap-northeast
☐	public-a	subnet-0d5864ac52e20e1e5	Available	vpc-0e755a2dfc566d299 exa...	10.0.0.0/24	-	251	ap-northeast-1a	apne1-az4	ap-northeast
☐	protect-rds-c	subnet-061dbdda51760b21d	Available	vpc-0e755a2dfc566d299 exa...	10.0.200.0/21	-	2043	ap-northeast-1c	apne1-az1	ap-northeast
☐	public-c	subnet-063695cadeb207c1b	Available	vpc-0e755a2dfc566d299 exa...	10.0.1.0/24	-	251	ap-northeast-1c	apne1-az1	ap-northeast

3.インターネットゲートウェイ

- 1.左側のメニューから「インターネットゲートウェイ」をクリックする



- 2.「インターネットゲートウェイの作成」をクリックする



AWS設定手順書

3. インターネットゲートウェイの情報を入力する

VPC > インターネットゲートウェイ > インターネットゲートウェイの作成

インターネットゲートウェイの作成 情報

インターネットゲートウェイは、VPCをインターネットに接続する仮想ルーターです。新しいインターネットゲートウェイを作成するには、ゲートウェイの名前を以下から指定します。

インターネットゲートウェイの設定

名前タグ
(Name) というキーと、指定した値を使用してタグを作成します。

exastro-gw

タグ - オプション

タグは、AWS リソースに割り当てられるラベルです。各タグはキーとオプションの値で構成されています。タグを使用してリソースを検索およびフィルタリングしたり、AWS のコストを追跡したりできます。

キー

exastro-gw

値 - オプション

exastro-gw

削除

新しいタグを追加

さらに 49 個のタグ、を追加できます。

キャンセル インターネットゲートウェイの作成

入力例)
名前タグ exastro-gw

4. 作成後、左上のアクションから「VPCにアタッチ」をクリックする

インターネットゲートウェイ (1/2) 情報

インターネットゲートウェイをフィルタ

	Name	インターネットゲートウェイ ID	状態	VPC ID	所有者
☒	exastro-gw	igw-08bb0714c392aa193	Detached	-	

アクション

VPC にアタッチ

VPC からデタッチ

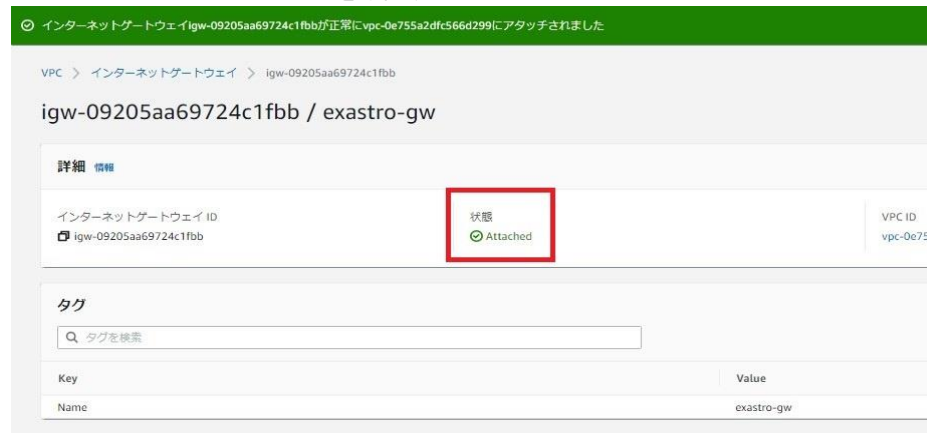
タグを管理

AWS設定手順書

5.作成したVPCを選択し、「インターネットゲートウェイのアタッチ」をクリックする



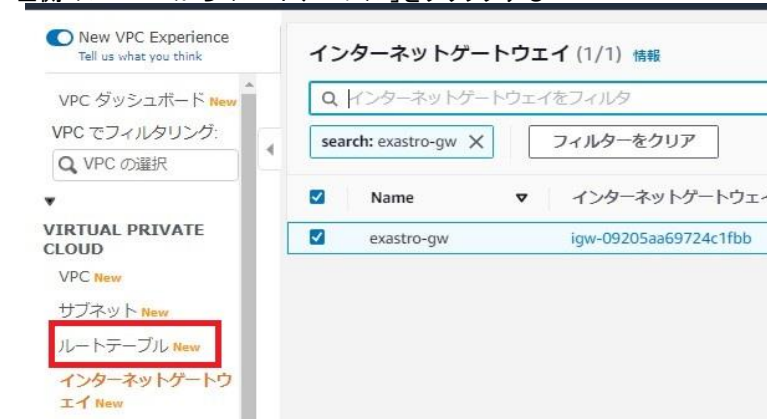
6.状態がAvailableになっていることを確認する



AWS設定手順書

4. ルートテーブル

1. 左側のメニューから「ルートテーブル」をクリックする



2. 作成したVPCのIDで検索ボックスから検索する



AWS設定手順書

3.対象のルートテーブルの下の「ルート」タブをクリックする

The screenshot shows the AWS Management Console interface for a route table. At the top, there is a table listing route tables with columns: Name, ルートテーブル ID, 明示的なサブネットの..., Edge の関連付け, メイン, and VPC. The first row is highlighted, showing 'exastro-rtb' with ID 'rtb-0b801ccd8abc1415b'. Below this table, the breadcrumb 'rtb-0b801ccd8abc1415b / exastro-rtb' is shown. Underneath, there are tabs: '詳細', 'ルート' (highlighted with a red box), 'サブネットの関連付け', 'Edge の関連付け', 'ルート伝播', and 'タグ'. The 'ルート (1)' section contains a search bar and a dropdown menu. Below this is a table with columns: 送信先, ターゲット, and ステータス. The first row shows '10.0.0.0/16' as the destination, 'local' as the target, and 'Active' as the status.

Name	ルートテーブル ID	明示的なサブネットの...	Edge の関連付け	メイン	VPC
exastro-rtb	rtb-0b801ccd8abc1415b	-	-	はい	vpc-0e755a2dfc566d299 exa...

rtb-0b801ccd8abc1415b / exastro-rtb

詳細 **ルート** サブネットの関連付け Edge の関連付け ルート伝播 タグ

ルート (1)

Q ルートのフィルタリング 両方 ▼

送信先	ターゲット	ステータス
10.0.0.0/16	local	Active

4.ルートの情報を追加する

The screenshot shows the 'Edit routes' page in the AWS Management Console. The breadcrumb is 'VPC > ルートテーブル > rtb-0b801ccd8abc1415b > Edit routes'. The title is 'Edit routes'. Below this is a table with columns: Destination, Target, Status, and Propagated. The first row shows '10.0.0.0/16' as the destination, 'local' as the target, 'Active' as the status, and 'いいえ' as the propagated status. The second row is being added, with '0.0.0.0/0' in the destination field and 'igw-09225aa69724c1fb5 (exastro-gw)' in the target field. The 'Status' column shows 'いいえ' and the 'Propagated' column shows 'いいえ'. There is a '削除' button next to the second row. At the bottom right, there are buttons: 'キャンセル', 'Preview', and '変更を保存' (highlighted with a red box).

VPC > ルートテーブル > rtb-0b801ccd8abc1415b > Edit routes

Edit routes

Destination	Target	Status	Propagated
10.0.0.0/16	local	Active	いいえ
0.0.0.0/0	igw-09225aa69724c1fb5 (exastro-gw)	いいえ	いいえ

Add route

キャンセル Preview **変更を保存**

AWS設定手順書

5.追加されていることを確認する

Updated routes for rtb-0b801ccd8abc1415b / exastro-rtb successfully
Details

VPC > ルートテーブル > rtb-0b801ccd8abc1415b

rtb-0b801ccd8abc1415b / exastro-rtb

詳細 情報

ルートテーブル ID rtb-0b801ccd8abc1415b	メイン はい	明示的なサブネットの関連付け -
VPC vpc-0e755a2dfc566d299 exastro-VPC	所有者 ID [REDACTED]	

ルート | サブネットの関連付け | Edge の関連付け | ルート伝播 | タグ

ルート (2)

Q ルートのフィルタリング 開方 ▼

送信先	ターゲット	ステータス
10.0.0.0/16	local	Active
0.0.0.0/0	igw-09205aa69724c1fbb	Active

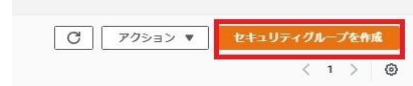
AWS設定手順書

5.セキュリティグループ

1.左側のメニューから「セキュリティグループ」をクリックする



2.「セキュリティグループを作成」をクリックする



3.セキュリティグループの情報を入力する

入力例)

セキュリティグループ名	exastro-sg
VPC	exastro-vpc

インバウンドルール

タイプ	プロトコル	ポート範囲	ソース
SSH	TCP	22	マイIP
HTTP	TCP	80	マイIP
HTTPS	TCP	443	マイIP
すべてのICMP-IPv4	IPv4	すべて	10.0.0.0/16 ※
カスタムUDP	UDP	5404-5405	10.0.0.0/16 ※
カスタムTCP	TCP	3121	10.0.0.0/16 ※
カスタムTCP	TCP	2224	10.0.0.0/16 ※

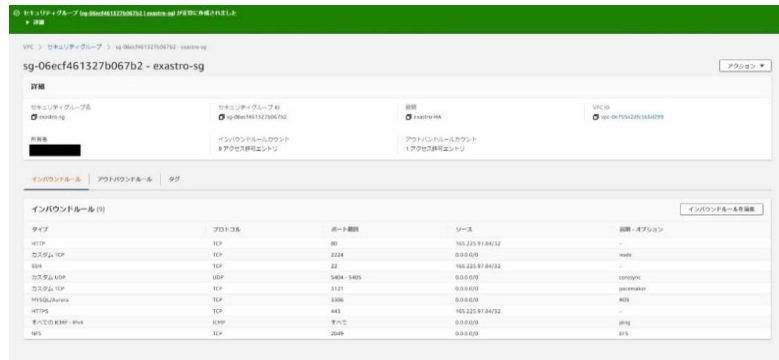
※ VPCのNWアドレス

アウトバウンドルール

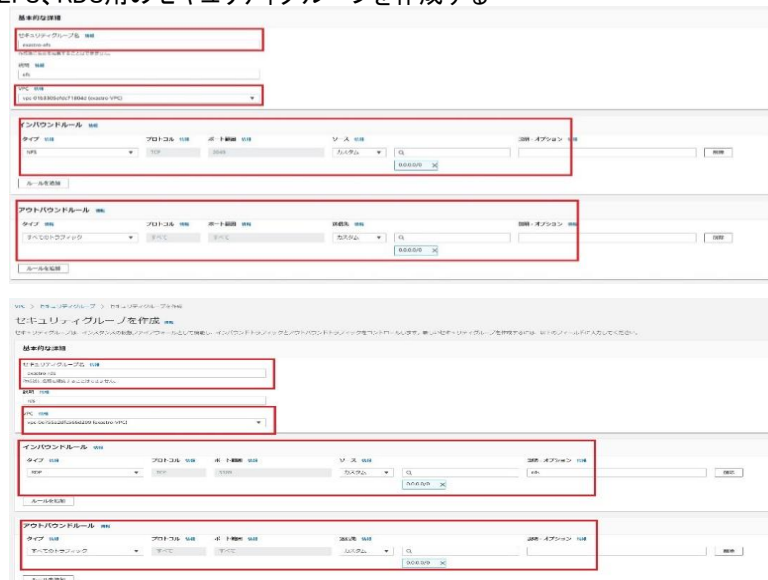
タイプ	プロトコル	ポート範囲	ソース
すべてのトラフィック	すべて	すべて	0.0.0.0/0

AWS設定手順書

4.作成されていることを確認する



5.EFS、RDS用のセキュリティグループを作成する



入力例)

セキュリティグループ名	exastro-efs
VPC	exastro-vpc

インバウンドルール

タイプ	プロトコル	ポート範囲	ソース
NFS	TCP	2049	10.0.0.0/16 ※

※ VPCのNWアドレス

アウトバウンドルール

タイプ	プロトコル	ポート範囲	ソース
すべてのトラフィック	すべて	すべて	0.0.0.0/0

入力例)

セキュリティグループ名	exastro-rds
VPC	exastro-vpc

インバウンドルール

タイプ	プロトコル	ポート範囲	ソース
MySQL/Aurora	TCP	3306	10.0.0.0/16 ※

※ VPCのNWアドレス

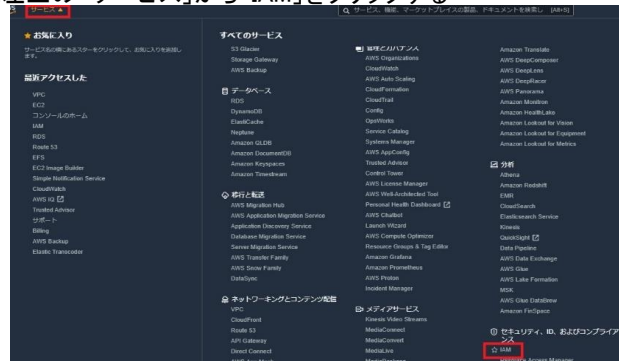
アウトバウンドルール

タイプ	プロトコル	ポート範囲	ソース
すべてのトラフィック	すべて	すべて	0.0.0.0/0

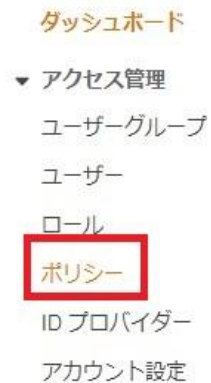
AWS設定手順書

6.IAM

1.左上の「サービス」から「IAM」をクリックする



2.左側のメニューの「ポリシー」をクリックする



3.「ポリシーの作成」をクリックする



AWS設定手順書

4.JSONタブを開いて、ポリシーを入力する

ポリシーの作成

1 2 3

ポリシーにより、ユーザー、グループ、またはロールに割り当てることができる AWS アクセス権限が定義されます。ビジュアルエディタでポリシーを作成または編集できます。詳細はこちら

ビジュアルエディタ **JSON** 管理ポリシーのインポート

```
1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Effect": "Allow",
6       "Action": [
7         "ec2:Describe*",
8         "ec2:DisassociateAddress",
9         "ec2:AssociateAddress"
10      ],
11      "Resource": [
12        "*"
13      ]
14    }
15  ]
16 }
```

セキュリティ: 0 エラー: 0 警告: 0 提案: 0

文字数: 150 / 6,144

キャンセル **次のステップ: タグ**

入力例)

ポリシー内容

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "ec2:Describe*",
        "ec2:DisassociateAddress",
        "ec2:AssociateAddress"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

5.「次のステップ: 確認」をクリックする

キャンセル 戻る **次のステップ: 確認**

AWS設定手順書

6. ポリシー名を入力し、「ポリシーの作成」をクリックする

ポリシーの作成

1 2 3

ポリシーの確認

名前* exastro-policy

英数字と「+','=','_」を使用します。最大 128 文字。

説明

最大 1000 文字。英数字と「+','=','_」を使用します。

概要

Q フィルター

サービス ▼

アクセスレベル

リソース

リクエスト条件

許可 (284 サービス中 1) 残りの 283 を表示

EC2

制限 リスト、読み込み、書き込み

すべてのリソース

なし

タグ

キー

▲ 値 ▼

リソースに関連付けられたタグはありません。

入力例)

名前

exastro-policy

* 必須

キャンセル

戻る

ポリシーの作成

7. 作成されたことを確認する

✓ exastro-policy が作成されました。

ポリシーの作成

ポリシーアクション ▼

ポリシーのフィルタ ▼

Q exastro

ポリシー名 ▼

タイプ

次として使用

説明

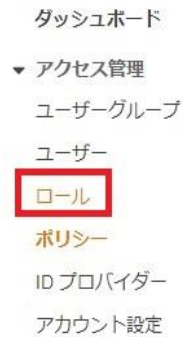
○ ▶

exastro-policy

ユーザーによる管理 なし

AWS設定手順書

8. 左側のメニューの「ロール」をクリックする



9. 「ロールの作成」をクリックする



10. 「AWSサービス」の「EC2」を選択し「次のステップ: アクセス権限」をクリックする



AWS設定手順書

11.先ほど作成したポリシーにチェックをし「次のステップ: タグ」をクリックする

ロールの作成

1 2 3 4

▼ Attach アクセス権限ポリシー

新しいロールにアタッチするポリシーを1つ以上選択します。

ポリシーの作成

ポリシーのフィルタ exastro 1件の結果を表示中

	ポリシー名	次として使用
☒	exastro-policy	なし

▼ アクセス権限の境界の設定

必須 キャンセル 戻る 次のステップ: タグ

12.「次のステップ: 確認」をクリックする

ロールの作成

1 2 3 4

タグの追加 (オプション)

IAM タグは、ロール に追加できるキーと値のペアです。タグには、E メールアドレスなどのユーザー情報を含めるか、役職などの説明文とすることができ、タグを使用して、この ロール のアクセスを整理、追跡、制御できます。詳細はこちら

キー	値 (オプション)	削除
新しいキーを追加		

さらに 50 個のタグを追加できます。

キャンセル 戻る 次のステップ: 確認

AWS設定手順書

13 ロール名を入力し、先ほど選択したポリシーになっていることを確認し「ロールの作成」をクリックする

ロールの作成

確認

以下に必要な情報を指定してこのロールを見直し、作成してください。

ロール名

英語字と「+,.@,=」を使用します。最大 64 文字。

ロールの説明

最大 1000 文字。英語字と「+,.@,=」を使用します。

信頼されたエンティティ

ポリシー

アクセス権限の境界

追加されたタグはありません。

キャンセル 戻る **ロールの作成**

入力例)

ロール名	exastro-role
ポリシー	exastro-policy

14.ロールができていることを確認する

ロールの作成 **ロールの削除**

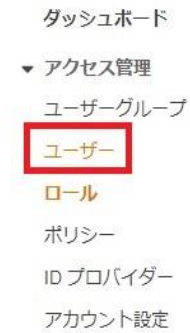
exastro

ロール名

☐ exastro-role

AWS設定手順書

15.左側のメニューの「ユーザー」をクリックする



16.「ユーザーの追加」をクリックする



17.ユーザー名とプログラムによるアクセスにチェックを入れ、「次のステップ: アクセス権限」をクリックする



入力例)

ユーザー名	exastro-user
アクセスの種類	プログラムによるアクセス

AWS設定手順書

18.既存のポリシーを直接アタッチを選択し、先ほど作成したポリシーをチェックして「次のステップ: タグ」をクリックする

ユーザーを追加 1 2 3 4 5

▼ アクセス許可の設定

ユーザーをグループに追加

アクセス権限を既存のユーザーからコピー

既存のポリシーを直接アタッチ

ポリシーの作成 🔄

ポリシーのフィルタ exastro 1件の結果を表示中

ポリシー名 ▼	タイプ	次として使用
☒ exastro-policy	ユーザーによる管理	Permissions policy (1)

▶ アクセス権限の境界の設定

キャンセル 戻る 次のステップ: タグ

19.「次のステップ: 確認」をクリックする

キャンセル 戻る 次のステップ: 確認

AWS設定手順書

20.内容を確認し「ユーザーの作成」をクリックする

ユーザーを追加

1 2 3 4 5

確認

選択内容を確認します。ユーザーを作成した後で、自動生成パスワードとアクセスキーを確認してダウンロードできます。

ユーザー詳細

ユーザー名	exastro-user
AWS アクセスの種類	プログラムによるアクセス - アクセスキーを使用
アクセス権限の境界	アクセス権限の境界が設定されていません

アクセス権限の概要

次のポリシー例は、上記のユーザーにアタッチされます。

タイプ	名前
管理ポリシー	exastro-policy

タグ

追加されたタグはありません。

キャンセル 戻る ユーザーの作成

21.csvのダウンロードを実施する ※サーバ設定時に使用する

ユーザーを追加

1 2 3 4 5

成功

以下に示すユーザーを正常に作成しました。ユーザーのセキュリティ認証情報を確認してダウンロードできます。AWS マネジメントコンソールへのサインイン手順を E メールでユーザーに送信することもできます。今回が、これらの認証情報をダウンロードできる最後の機会です。ただし、新しい認証情報はいつでも作成できます。

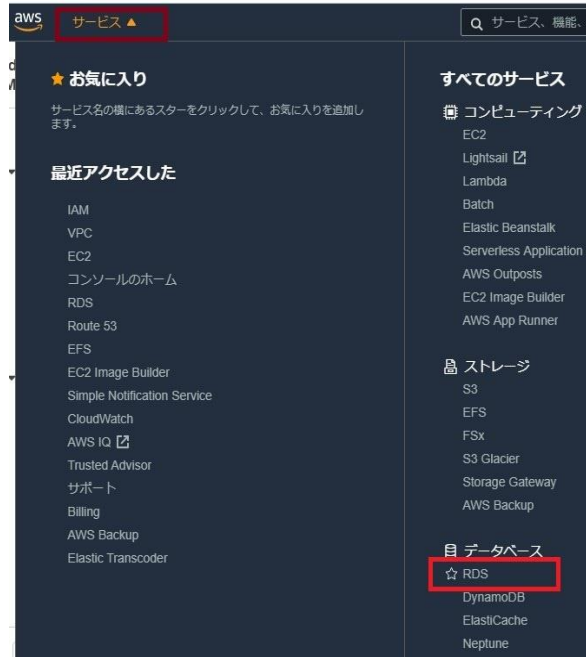
AWS マネジメントコンソールへのアクセス権を持つユーザーは「<https://943001175239.signin.aws.amazon.com/console>」でサインインできます

📄 .csv のダウンロード

	ユーザー	アクセスキー ID	シークレットアクセスキー
▶	✔ exastro-user	██████████	***** 表示

7.RDS

1.左上のサービスからRDSをクリックする



2.左側のメニューから「サブネットグループ」をクリックする



AWS設定手順書

3.「DBサブネットグループ作成」をクリックする



4.DBサブネットグループの情報を入力し、「作成」をクリックする

RDS > Subnet groups > DB サブネットグループを作成

DB サブネットグループを作成

新しいサブネットグループを作成するには、名前と説明を入力し、既存の VPC を選択します。その後、その VPC に関連するサブネットを追加できます。

サブネットグループの詳細

名前
サブネットグループの作成後に名前を変更することはできません。
exastrodb-subnet
1～255 文字にする必要があります。英数字、スペース、ハイフン、アンダースコア、ピリオドを使用できます。

説明
rds-subnet

VPC
DB サブネットグループに使用するサブネットに対応する VPC 識別子を選択します。サブネットグループが作成された後、別の VPC 識別子を選択することはできません。
exastro-VPC (vpc-0e755a2dfc566d299)

サブネットを追加

アベイラビリティゾーン
追加するサブネットを含むアベイラビリティゾーンを選択します。
アベイラビリティゾーンを選択
ap-northeast-1a X ap-northeast-1c X

サブネット
追加するサブネットを選択します。リストには、選択したアベイラビリティゾーンのサブネットが含まれます。
サブネットを選択
subnet-097fba3cefd69f30 (10.0.192.0/21) X
subnet-061dbdda51760b21d (10.0.200.0/21) X

アベイラビリティゾーン	サブネット ID	CIDR ブロック
ap-northeast-1a	subnet-097fba3cefd69f30	10.0.192.0/21
ap-northeast-1c	subnet-061dbdda51760b21d	10.0.200.0/21

キャンセル 作成

入力例)

名前	exastrodb-subnet
説明	rds-subnet
VPC	exastro-VPC
アベイラビリティゾーン	ap-northeast-1a
	ap-northeast-1c
サブネット	subnet-097fba3cefd69f30 (10.0.192.0/21)
	subnet-061dbdda51760b21d (10.0.200.0/21)

AWS設定手順書

5. 左側のパラメータグループをクリックし、「パラメータグループの作成」をクリックする



6. パラメーターグループファミリーを選択し、グループ名を記載して「作成」をクリックする ※説明は任意です



入力例)

パラメーターグループファミリー	mariadb10.5
グループ名	exastro-param

7. 作成されたことを確認する



8. 作成したパラメーターグループをクリックして、パラメーターの設定を実施する ※一つ設定するにつき「変更の保存」を必ずクリックすること



AWS設定手順書

RDS > パラメータグループ > exastro-param

exastro-param

パラメータ

innodb_buffer_pool_size

編集のキャンセル 変更のプレビュー 元に戻す 変更の保存

< 1 > ⓘ

☐	名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
☐	innodb_buffer_pool_size	536870912	0-18446744073709551615	true	system	dynamic	integer	The size in bytes of the memory buffer innodb uses to cache data and indexes of its tables.

RDS > パラメータグループ > exastro-param

exastro-param

パラメータ

min_examined_row_limit

編集のキャンセル 変更のプレビュー 元に戻す 変更の保存

< 1 > ⓘ

☐	名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
☐	min_examined_row_limit	100	0-18446744073709551615	true	engine-default	dynamic	integer	Can be used to cause queries which examine fewer than the stated number of rows not to be logged.

RDS > パラメータグループ > exastro-param

exastro-param

パラメータ

innodb_log_buffer_size

編集のキャンセル 変更のプレビュー 元に戻す 変更の保存

< 1 > ⓘ

☐	名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
☐	innodb_log_buffer_size	67108864	262144-4294967295	true	system	static	integer	The size in bytes of the buffer that innodb uses to write to the log files on disk.

パラメータ

query_cache_size

編集のキャンセル 変更のプレビュー 元に戻す 変更の保存

< 1 > ⓘ

☐	名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
☐	query_cache_size	536870912	0-9223372036854774784	true	engine-default	dynamic	integer	The amount of memory allocated for caching query results.

exastro-param

パラメータ

join_buffer_size

編集のキャンセル 変更のプレビュー 元に戻す 変更の保存

< 1 > ⓘ

☐	名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
☐	join_buffer_size	134217728	128-18446744073709551615	true	engine-default	dynamic	integer	Increase the value of join_buffer_size to get a faster full join when adding indexes is not possible.

パラメータ

query_cache_type

編集のキャンセル 変更のプレビュー 元に戻す 変更の保存

< 1 > ⓘ

☐	名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
☐	query_cache_type	1	0-2	true	engine-default	static	integer	For query results either don't cache (=OFF), cache except for NO_CACHE (=ON), or only CACHE (=DEMAND)

パラメータ

tmp_table_size

編集のキャンセル 変更のプレビュー 元に戻す 変更の保存

< 1 > ⓘ

☐	名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
☐	tmp_table_size	67108864	0-4294967295	true	engine-default	dynamic	integer	If an in-memory temporary table exceeds the limit, MySQL automatically converts it to an on-disk MyISAM table. Increased value can improve perf for many advanced GROUP BY queries.

AWS設定手順書

パラメータ

Q: mrr_buffer_size

名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
mrr_buffer_size	67108864	8192-2147483647	true	engine-default	dynamic	integer	Size of buffer to use when using MRR with range access.

パラメータ

Q: max_heap_table_size

名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
max_heap_table_size	67108864	16384-1844674407370954752	true	engine-default	dynamic	integer	Maximum size to which MEMORY tables are allowed to grow.

パラメータ

Q: character_set_server

名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
character_set_server	utf8	big5, dec8, cp850, hb, ko8r, latin1, latin2, nwc7, ssc, uja, sjis, hebrew, tis620, eucjr, ko8rs, gb2312, greek, cp1250, gbk, latin5, armSCII8, utf8, ucs2, cp866, keybc2, macce, macroman, cp852, latin7, utf8mb4, cp1251, utf16, cp1256, cp1257, utf32, binary, georgian, cp932, euojms	true	engine-default	dynamic	string	The server's default character set.

パラメータ

Q: max_connections

名前	値	許可された値	変更可能	送信元	適用タイプ	データ型	説明
extra_max_connections		1-100000	false	engine-default	dynamic	integer	The number of connections on the extra_port.
max_connections	256	1-100000	true	system	dynamic	integer	The number of simultaneous client connections allowed.

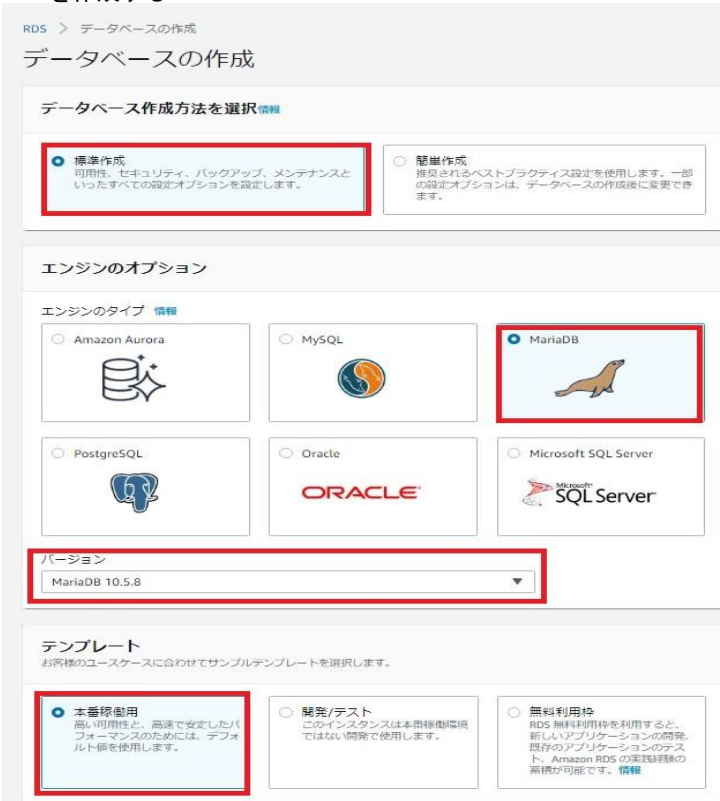
パラメーター	値
explicit_defaults_for_timestamp	1
tx_isolation	READ-COMMITTED
innodb_buffer_pool_size	536870912
innodb_log_buffer_size	67108864
min_examined_row_limit	100
join_buffer_size	134217728
query_cache_size	536870912
query_cache_type	1
max_heap_table_size	67108864
tmp_table_size	67108864
mrr_buffer_size	67108864
max_connections	256
character_set_server	utf8

AWS設定手順書

9.保存後、左側メニューのダッシュボードから「データベースの作成」をクリックする



10.RDSを作成する



※入力例の記載が無いものはデフォルト値です(パスワードは任意です)
入力例)

作成方法	標準作成
エンジンのオプション	MariaDB
バージョン	MariaDB 10.5.8
テンプレート	本番稼働用
DBインスタンス識別子	exastro-rds
マスターユーザー名	admin
可用性と耐久性	スタンバイインスタンスを作成する
VPC	exastro-VPC
サブネットグループ	exastrodb-subnet
パブリックアクセス可能	なし
VPCセキュリティグループ	既存の選択 exastro-rds
データベースポート	3306
DBパラメータグループ	exastro-param
オプショングループ	defaultmariadb-10.5
バックアップ	自動バックアップの有効化
モニタリング	拡張モニタリングの有効化
ログのエクスポート	監視ログ エラーログ 全般ログ スロークエリログ
選択ウィンドウ	土曜日 03:00 0.5時間

AWS設定手順書

設定

DB インスタンス識別子 情報

DB インスタンスの名前を入力します。この名前は、AWS アカウントが現在の AWS リージョンで所有しているすべての DB インスタンスにおいて一意である必要があります。

exastro-rds

DB インスタンス識別子は英文字と小文字を識別しませんが、すべて小文字で保存されます (例: "mydbinstance")。制約として、使用できるのは 1~60 文字以内で英数字またはハイフンのみです。1 文字目は英文字でなければなりません。また、ハイフンを連続で 2 つ使ったり、最後の文字をハイフンにしたがすることはできません。

▼ 認証情報の設定

マスターユーザー名 情報

DB インスタンスのマスターユーザーのログイン ID を入力します。

admin

1~16 文字の英数字。1 文字目は文字である必要があります

☐ パスワードの自動生成

Amazon RDS がパスワードを生成するか、お客様がご自分でパスワードを指定することができます

マスターパスワード 情報

制約事項: 表示可能な ASCII 文字で 8 文字以上で入力してください(次の文字を含めることはできません: / (スラッシュ)、\ (バックスラッシュ)、~ (チルダ)、_ (アンダースコア)、@ (アットマーク))

パスワードを確認 情報

DB インスタンスサイズ

DB インスタンスクラス 情報

処理能力とメモリの要件に合った DB インスタンスクラスを選択します。以下の DB インスタンスクラスオプションは、上記で選択したエンジンでサポートされているものに制限されます。

☒ 標準クラス (m クラスを含む)

☐ メモリ最適化クラス (r クラスと x クラスを含む)

☐ パースト可能クラス (t クラスを含む)

db.m6g.large
2 vCPUs 8 GiB RAM Network: 4,750 Mbps

☒ 以前の世代のクラスを含める

ストレージ

ストレージタイプ 情報

プロビジョンド IOPS (SSD)

ストレージ割り当て

100 GiB

最小: 100 GiB、最大: 65,536 GiB

プロビジョンド IOPS 情報

3000 IOPS

Minimum: 1,000 IOPS, Maximum: 80,000 IOPS

ストレージの自動スケーリング 情報

アプリケーションのニーズに基づいて、データベースのストレージに対する動的なスケーリングのサポートを提供します。

☒ ストレージの自動スケーリングを有効にする

この機能を有効にすると、指定したしきい値を超えた場合にストレージを増やすことができます。

最大ストレージしきい値 情報

データベースが指定したしきい値に自動スケールされると、料金が適用されます。

1000 GiB

最小: 101 GiB、最大: 65,536 GiB

可用性と耐久性

マルチ AZ 配置 情報

☒ スタンバイインスタンスを作成する (本番環境向けに推奨)

データの可用性を確保し、RPO のゼロを実現。システムバックアップの間のレイテンシーの急上昇を最小限に抑えるために、別の Availability ゾーン (AZ) にスタンバイを作成します。

☐ スタンバイインスタンスを作成しないでください

IT Automation オールインワン型HA構成(AWS)インストールマニュアル 39 / 68

AWS設定手順書

接続

Virtual Private Cloud (VPC) 情報
この DB インスタンスの接続エンドポイントを提供する VPC。
exastro-VPC (vpc-0e755a2dfc566d299)
この DB サブネットグループがある VPC のみが表示されます。

① データベースの作成後に、VPC の選択を置き換えることはできません。

サブネットグループ 情報
選択した VPC で DB インスタンスが使用できるサブネットと IP 範囲を定義する DB サブネットグループ。
exastrodb-subnet

パブリックアクセス可能 情報
☐ あり
VPC 外部の Amazon EC2 インスタンスとデバイスがお客様のデータベースに接続できます。データベースに接続できる VPC 内の EC2 インスタンスおよびデバイスを選択する 1 つ以上の VPC セキュリティグループを選択します。
☒ なし
RDS はパブリック IP アドレスをデータベースに割り当てません。VPC 内部の Amazon EC2 インスタンスとデバイスのみお客様のデータベースに接続できます。

VPC セキュリティグループ
RDS セキュリティグループを 1 つ以上選択し、データベースへのアクセスを許可します。セキュリティグループのルールで EC2 インスタンスと VPC 外のデバイスからの通信トラフィックが許可されていることを確認します (セキュリティグループはパブリックにアクセス可能なデータベースに必要です)。
☒ 既存の選択
既存の VPC セキュリティグループの選択
☐ 新規作成
新しい VPC セキュリティグループの作成

既存の VPC セキュリティグループ
VPC セキュリティグループを選択します
exastro-rds

▼ 追加の接続設定
データベースポート 情報
データベースがアプリケーションの接続に使用する TCP/IP ポート。
3306

▼ 追加設定
データベースオプション、暗号化、バックアップ、バックアップ、Performance Insights、拡張モニタリング、メンテナン、CloudWatch Logs、削除保護、有効

データベースの選択
最初のデータベース名
データベース名を指定しないと、Amazon RDS はデータベースを作成しません。
DB / パラメータグループ 情報
exastro-param
オプショングループ 情報
default-mariadb-10-5

バックアップ
データベースのポイントインタイムスナップショットを作成します。
☒ 自動バックアップの有効化
データベースのポイントインタイムスナップショットを作成します。

バックアップ保持期間 情報
このインスタンスの自動バックアップを RDS が保存する日数を選択します。
7 日間

バックアップウィンドウ 情報
Amazon RDS によって作成されるデータベースの自動バックアップの期間を選択します。
☒ 選択ウィンドウ
☐ 設定なし

開始時間
05 : 00 UTC
期間
0.5 時間

☒ スナップショットにタグをコピー

暗号化
☒ 暗号を有効化
選択して対象のインスタンスを暗号化します。マスタキー ID とエンクリプション、Key Management Service (KMS) コンソールを使用して作成した後に、リストに表示されます。 情報

AWS設定手順書

マスターキー [情報](#)
 (default) aws/rds

アカウント
 943001175239

KMS キー ID
 a73ddb74-38e3-4254-8607-d2a2bea62356

Performance Insights [情報](#)
☒ Performance Insights を有効にする
 保持期間 [情報](#)
 デフォルト (7 日)

マスターキー [情報](#)
 (default) aws/rds

アカウント
 943001175239

KMS キー ID
 a73ddb74-38e3-4254-8607-d2a2bea62356

⚠ Performance Insights マスター KMS キーは、Performance Insights を有効にした後は変更できません。

モニタリング
☒ 拡張モニタリングの有効化
 拡張モニタリングメトリクスを有効にすると、さまざまなプロセスやスレッドで CPU がどのように使用されているのかを確認したいと常に便利です

詳細度
 60 秒

モニタリングロール
 デフォルト

[データベースの作成] をクリックすると、RDS が IAM ロール `rdsmonitoring-role` を作成することをお勧めします。

ログのエクスポート
 Amazon CloudWatch Logs に発行するログタイプを選択します
☒ 監査ログ
☒ エラーログ
☒ 全ログ
☒ スロークエリログ

IAM ロール
 サービスにリンクされた以下のロールは、CloudWatch Logs にログを発行するために使用されます。
 RDS service-linked role

① 全ログ、スロークエリ、監査ログがオンになっていることを確認してください。デフォルトではエラーログが有効になっています。 [詳細はこちら](#)

メンテナンス
 マイナーバージョン自動アップグレード [情報](#)
☐ マイナーバージョン自動アップグレードの有効化
 マイナーバージョン自動アップグレードを有効にすると、新しいマイナーバージョンがリリースされたときに自動的にアップグレードされます。自動アップグレードは、データベースのメンテナンスウィンドウに行われます。

メンテナンスウィンドウ [情報](#)
 Amazon RDS によってデータベースに適用されている保留中の変更またはメンテナンスの期間を選択します。
☒ 選択ウィンドウ
☐ 設定なし

開始日
 土曜日

開始時間
 03 : 00 UTC

期間
 0.5 時間

削除保護
☒ 削除保護の有効化
 データベースが誤って削除されるのを防ぎます。このオプションが有効になっている場合、データベースを削除することはできません。

11.RDSが作成され、利用可能になることを確認する ※少し時間がかかります

データベース

グループプリミティブ

変更

アクション

S3から復元

データベースの作成

DB 識別子	ロール	エンジン	リージョンと AZ	サイズ	ステータス	CPU	現在のアクティビティ	メンテナンス	VPC	マルチ AZ
exadb-rds	インスタンス	MySQL	ap-northeast-1	db.m5g.large	利用可能	1.00%	バックアップセッション	vpc-0c75a2dfc5664299	単一	

8.EFS

1.左上のサービスからEFSをクリックする



2.「ファイルシステムの作成」をクリックする



AWS設定手順書

3.名前を入力し、作成したVPCを選択、リージョンを選択し「カスタマイズ」をクリックする

ファイルシステムの作成

サービスの推奨設定を使用して EFS ファイルシステムを作成します。 [詳細はこちら](#)

名前 - オプション

ファイルシステムに名前を付けます。

exastro-efs

名前は 255 文字より長くすることはできません。使用できるのは文字、数字、および一部の記号 (+ - =, _ : /) のみです。

Virtual Private Cloud (VPC)

EC2 インスタンスをファイルシステムに接続する VPC を選択します。 [詳細はこちら](#)

vpc-0e755a2dfc566d299
exastro-VPC

可用性と耐久性

リージョンのストレージクラスを使用してファイルシステムを作成するには、リージョンを選択します (推奨)。One Zone ストレージクラスを使用してファイルシステムを作成するには、One Zone を選択します。 [詳細はこちら](#)

☒ リージョン

複数の AZ にデータを冗長的に保存

☐ 1 ゾーン

単一の AZ 内にデータを冗長的に保存

キャンセル

カスタマイズ

作成

入力例)

名前-オプション	exastro-efs
VPC	exastro-VPC
可用性と耐久性	リージョン

AWS設定手順書

4.自動バックアップを選択し、ネットワークは先ほどのVPCから選択し、EFS用に作成したセキュリティグループを追加し「次へ」をクリックする

金融

名前・オプション
ファイルシステムに名前を付けます。
exastro-efs

可用性と耐久性
リージョンのストレージクラスを使用してファイルシステムを作成するには、リージョンを選択します(推奨)。One Zoneストレージクラスを使用してファイルシステムを作成するには、One Zoneを選択します。詳細はこちら

リージョン
リージョンのAZにデータを冗長的に保存

1ゾーン
単一のAZ内にデータを冗長的に保存

自動バックアップ
EFSに自動バックアップを有効化
自動バックアップを有効化すると、EFSに自動バックアップが有効化されます。追加料金が適用されます。詳細はこちら

ライフサイクル管理
オブジェクトがアクセスストレージクラスにファイルを移動することで、アクセスパターンの変化に応じて自動的に費用を削減できます。詳細はこちら

最後のアクセスから 30 日

パフォーマンスモード
必要なIOPSに基づいて、ファイルシステムのパフォーマンスモードを設定します。詳細はこちら

汎用
ウェブアプリケーションやコンテンツ管理システムなど、レイテンシーの削減を要しやすいユースケースに最適

最大I/O
1秒あたりの最大スループットおよび最大バッチサイズをより高いレベルにスケール

スループットモード
ファイルシステムのスループット制限がどのように決定されるかを設定します。詳細はこちら

バースト
スループットはファイルシステムサイズに応じてスケール

プロビジョニング済み
指定した量に設定されたスループット

暗号化
保存時のファイルシステムのデータの暗号化を有効にすることを選択します。デフォルトでは、AWS KMS サービス (aws:KmsKeySystem) を使用します。詳細はこちら

保存時のデータの暗号化を有効にする

暗号化設定のカスタマイズ

ネットワークアクセス

ネットワーク

Virtual Private Cloud (VPC)
EC2 インスタンスをプライベートネットワークに接続する VPC を選択します。詳細はこちら

vpc-0a755a2b55662289
exastro-vpc

マウントターゲット
マウントターゲットは、Amazon EFS ファイルシステムをマウントできる NFSv4 エンドポイントを提供します。プライベートIPバージョンごとに 1 つのマウントターゲットを有効化することをお勧めします。詳細はこちら

アベイラビリティゾーン
ap-northeast-1a

サブネット ID
subnet-0b7b6a0f0909f30

IP アドレス
自動

セキュリティグループ
セキュリティグループを選択
sg-0d8baad sg-002ae0a6b2a07757 X
exastro-efs

セキュリティグループを選択
sg-0d8baad sg-002ae0a6b2a07757 X
exastro-efs

マウントターゲットを追加

キャンセル 次へ 完了

入力例)

ネットワーク	exastro-VPC
セキュリティグループ	exastro-efs

AWS設定手順書

5.「次へ」をクリックする

ファイルシステムポリシー - オプション

ポリシーオプション

この画面の一般的なポリシーオプションオプションを1つ以上選択するか、エディタを使用してカスタムポリシーを作成します。詳細はここから見る

- ☐ デフォルトでスタートアップアクセスを禁止する
- ☐ デフォルトで読み取り専用アクセスを強制する
- ☐ 読み取りアクセスを禁止
- ☐ すべて読み取り専用モードに設定して監査用の書き込みを強制する

*各オプションは、このポリシーと他のポリシーの両方で同時に使用できません。

▶ 監査のアクセス許可を有効

ポリシーエディタ (JSON)

エディタで変更すると、エディタが有効化されるまで有効化ポリシーオプションは変更できません。

キャンセル 前へ 次へ

6.内容を確認し「作成」をクリックする

確認して作成する

ステップ 1: ファイルシステムの概要

編集

ファイルシステム

フィールド	値	編集可能ですか?
名前	exastro-efs	あり
パフォーマンスモード	汎用	なし
スケーラビリティモード	バースト	あり
暗号化	あり	なし
KMS キー ID	-	なし
ライフサイクルポリシー	最後のアクセスから 30 日	あり
自動バックアップ	あり	あり
VPC ID	vpc-0a755a28f366d299 (exastro-VPC)	あり
アベイラビリティゾーン	リージョン	なし

タグ

タグキー ▼ タグ値 ▼

このリソースに割り当てられたタグがありません

タグを管理

ステップ 2: ネットワークアクセス

編集

マウントターゲット

アベイラビリティゾーン	サブネット	IP アドレス	セキュリティグループ
ap-northeast-1a	subnet-097ba3d0f0ff0f00	-	sg-0a88aa2a04043c1211
ap-northeast-1c	subnet-0c7fba0b51740212d	-	sg-0a88aa2a04043c1211

ステップ 3: ファイルシステムポリシー

編集

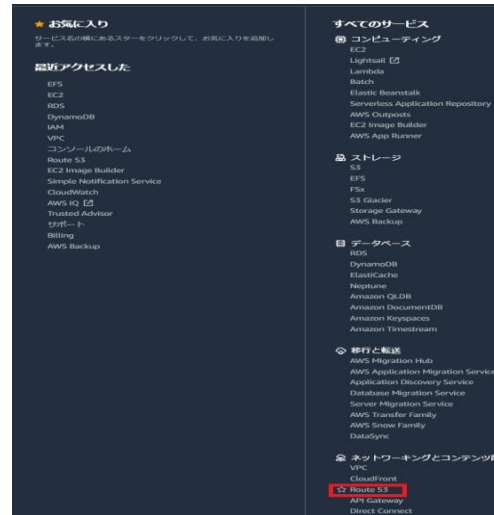
▼ ファイルシステムポリシー

エディタ

キャンセル 前へ 作成

9.Route53

1.左上のサービスからRoute53をクリックする



2.「ホストゾーンの作成」をクリックする



AWS設定手順書

3.ドメイン名を入力し、プライベートホストゾーンにチェックし、東京リージョンとVPCを選択し「ホストゾーンの作成」をクリックする

ホストゾーン設定
ホストゾーンは、example.com などのドメインおよびそのサブドメインのトラフィックのルーティング方法に関する情報を保持するコンテナです。

ドメイン名 Info
これは、トラフィックをルーティングするドメインの名称です。

許可された文字: 小文字、大文字、数字、ハイフン、およびドット。開始文字は小文字でなければなりません。例: {} * * . - / : ; < > ? @ [\] ^ _ " { } | , ~

説明 - オプション Info
この場で、同じ名前空間のホストゾーンを区別できます。

説明は最大 256 文字です。 0/256

タイプ Info
このタイプは、インターネットまたは Amazon VPC でトラフィックをルーティングするかどうかを示します。

☐ **パブリックホストゾーン**
パブリックホストゾーンは、インターネットのトラフィックのルーティング方法を決定します。

☒ **プライベートホストゾーン**
プライベートホストゾーンは、Amazon VPC 内でのトラフィックのルーティング方法を決定します。

ホストゾーンに関連付ける VPC Info
このホストゾーンを使用して 1 つ以上の VPC の DNS クエリを解決するには、当該の VPC を選択します。別の AWS アカウントで作成された VPC をホストゾーンに関連付けるには、AWS CLI などのプログラム的な方法を用いる必要があります。

☒ **プライベートホストゾーンに関連付ける各 VPC に対して、Amazon VPC 設定 enableDnsHostnames および enableDnsSupport を true に設定する必要があります。**

リージョン Info **VPC ID** Info

タグ Info
ホストゾーンにタグを適用して、整理、監視しやすくします。

このリソースに関連付けられたタグがありません。

タグは最大であと 50 個追加できます。

入力例)

ドメイン名	exastro-aws.local
タイプ	プライベートホストゾーン
リージョン	アジアパシフィック(東京)[ap-northeast-1]
VPC ID	vpc-0e755a2dfc566d299

4.「レコードの作成」をクリックする

レコードの作成 **ホストゾーンのタグ**

レコード (2) Info
Amazon Route 53 は、DNS レコードを保持するためのデータベースを提供します。レコードを変更するには、[編集] をクリックします。

☐ **レコード名**

☐	exastro-aws.local	NS	シンブル	-	ns-1536.awsdns-00.co.uk, ns-0.awsdns-00.com, ns-1024.awsdns-00.org, ns-512.awsdns-00.net
☐	exastro-aws.local	SOA	シンブル	-	ns-1536.awsdns-00.co.uk, awsdns-hostmaster.amazon.com. 1 7200 900 1209600 86400

AWS設定手順書

5.レコード名、CNAMEを選択しRDSのエンドポイントを値に入力し、「レコードを作成」をクリックする

Route 53 > ホストゾーン > exastro-aws.local > レコードを作成

レコードのクイック作成 [Info](#) [ウィザードに切り替える](#) [別のレコードを追加](#)

▼ レコード 1 [削除](#)

レコード名 [Info](#) .exastro-aws.local
有効な文字: a-z, 0-9, ! * # \$ % & ' () * +, - / ; < = > ? @ [\] ^ _ ` { | } . ~

レコードタイプ [Info](#) CNAME – 別のドメイン名および一部の AWS ... ▼

値 [Info](#)
複数の値を個別の行に入力します。

TTL (秒) [Info](#)

ルーティングポリシー [Info](#) シンプルルーティング ▼

[1分](#) [1時間](#) [1日](#)
推奨値: 60~172800 (2日間)

[キャンセル](#) [レコードを作成](#)

入力例)

レコード名	rds
レコードタイプ	CNAME
値	RDSのエンドポイント

6.レコード名、CNAMEを選択しEFSのエンドポイントを値に入力し、「レコードを作成」をクリックする

Route 53 > ホストゾーン > exastro-aws.local > レコードを作成

レコードのクイック作成 [Info](#) [ウィザードに切り替える](#) [別のレコードを追加](#)

▼ レコード 1 [削除](#)

レコード名 [Info](#) .exastro-aws.local
有効な文字: a-z, 0-9, ! * # \$ % & ' () * +, - / ; < = > ? @ [\] ^ _ ` { | } . ~

レコードタイプ [Info](#) CNAME – 別のドメイン名および一部の AWS ... ▼

値 [Info](#)
複数の値を個別の行に入力します。

TTL (秒) [Info](#)

ルーティングポリシー [Info](#) シンプルルーティング ▼

[1分](#) [1時間](#) [1日](#)
推奨値: 60~172800 (2日間)

[キャンセル](#) [レコードを作成](#)

入力例)

レコード名	efs
レコードタイプ	CNAME
値	EFSのエンドポイント

AWS設定手順書

10.EC2

1.左上のサービスからEC2をクリックする



2.「インスタンスを起動」をクリックする



3.Red Hat with Linux 8 High Availabilityと検索ボックスに入力し、選択する ※Pacemaker、Corosync、pcsが既存でインストールされているイメージです



AWS設定手順書

4.サイズを選択する ※Exastroに必要な最低限のスペックにて実施

ステップ 2: インスタンスタイプの選択

Amazon EC2 では、異なるユーザーケースに合わせて最適化されたさまざまなインスタンスタイプが用意されています。インスタンスとは、アプリケーションを実行できる仮想サーバーです。インスタンスタイプはさまざまな CPU、メモリ、ストレージ、ネットワークキャパシティの組み合わせによって構成されているため、使用するアプリケーションに合わせて適切なリージョンの組み合わせを柔軟に選択できます。インスタンスタイプおよびそれとコンプライアンスのニーズに適用する方法に関する情報は、[こちら](#)をご覧ください。

フィルター条件: **すべてのインスタンスファミリー** **実行状態** **他の表示/表示法**

現在表示中: 12 medium | 2 vCPU, 2.5 GHz, 4 GB メモリ (5/6)

ファミリー	タイプ	vCPU ①	メモリ (GB)	インスタンスストレージ (GB) ①	EBS 最適化済み ①	ネットワークパフォーマンス ①	IPv6 サポート ①
	t2.nano	1	0.5	EBS のみ	-	低から中	はい
	t2.micro	1	1	EBS のみ	-	低から中	はい
	t2.small	1	2	EBS のみ	-	低から中	はい
	t2.medium	2	4	EBS のみ	-	低から中	はい
	t2.large	2	8	EBS のみ	-	低から中	はい
	t2.xlarge	4	16	EBS のみ	-	中	はい
	t2.2xlarge	8	32	EBS のみ	-	中	はい
	t3.nano	2	0.5	EBS のみ	はい	最大 1 ギガビット	はい
	t3.micro	2	1	EBS のみ	はい	最大 1 ギガビット	はい
	t3.small	2	2	EBS のみ	はい	最大 1 ギガビット	はい
	t3.medium	2	4	EBS のみ	はい	最大 1 ギガビット	はい
	t3.large	2	8	EBS のみ	はい	最大 1 ギガビット	はい
	t3.xlarge	4	16	EBS のみ	はい	最大 1 ギガビット	はい

キャンセル 戻る **確認と作成** 次のステップ: インスタンスの起動

5.作成したVPC、publicなサブネット、自動割り当てパブリックIP、作成したIAMロールを選択し「次のステップ:ストレージの追加」をクリックする

[illegible]

入力例)

ネットワーク	exastro-VPC
サブネット	public-a
自動割り当てパブリックIP	有効
IAMロール	exastro-role
セキュリティグループ	exastro-sg

入力例)

ネットワーク	exastro-VPC
サブネット	public-c
自動割り当てパブリックIP	有効
IAMロール	exastro-role
セキュリティグループ	exastro-sg

AWS設定手順書

6.「次のステップ: タグの追加」をクリックする

1. IAM の設定 2. インスタンスタイプを選択 3. インスタンスの設置 4. ストレージの選択 5. ネットワークの選択 6. セキュリティグループの選択 7. 確認

ステップ 4: ストレージの追加

インスタンスは元のストレージ(EBS)設定を保持して作成されます。インスタンスに追加の EBS ボリュームやインスタンスストアボリュームをアタッチするか、ルートボリュームの拡張を編成することができます。また、インスタンスを作成してから追加の EBS ボリュームをアタッチすることもできますが、インスタンスストアボリュームはアタッチできません。Amazon EC2 のストレージオプションに関する詳細は、こちらをご覧ください。

ボリュームタイプ ①	デバイス ①	スナップショット ①	サイズ (GB) ①	ボリュームタイプ ①	IOPS ①	スループット (MB/s) ①	終了時に削除 ①	暗号化 ①	
ルート	/dev/sda1	snap-0a2aa7e2390810460	15	標準 HDD (gp2)	150 / 300	150 / 300	設定なし	☒	暗号化なし

新しい EBS ボリュームの作成

無料利用枠の付帯であるお薬価は 30 GB までの EBS 容量 (SSD) ストレージまたはマウントされたマウントされたストレージで構成されます。無料利用枠の付帯と使用制限に関する詳細はこちら。

キャンセル 戻る **次のステップ: タグの追加**

7.「次のステップ: セキュリティグループ」をクリックする

1. IAM の設定 2. インスタンスタイプを選択 3. インスタンスの設置 4. ストレージの選択 5. ネットワークの選択 6. セキュリティグループの選択 7. 確認

ステップ 6: タグの追加

タグは、本デモセットで提供されるサーバーの識別に役立ちます。たとえば、キーに「Name」、値に「MyInstance」を使用してタグを設定することができます。タグにはキーと値のペアがあり、インスタンス、ネットワーク、およびその他のリソースに適用できます。タグは、すべてのインスタンスとボリュームに適用されます。Amazon EC2 リソースのタグ付けに関する詳細はこちら。

キー ① (最大 128 文字)	値 ① (最大 256 文字)	インスタンス ①	ボリューム ①	ネットワークインターフェイス ①

このリソースにはタグが 1 つあります。タグの追加は、このリソースにタグを追加して Name タグを設定し、完了してください。タグを作成する場合は、タグの作成が IAM ポリシーに許可されていることを確認します。

タグの追加 (最大 50 個のタグ)

キャンセル 戻る **次のステップ: セキュリティグループの設定**

AWS設定手順書

8. 作成したセキュリティグループを選択し、「確認と作成」をクリックする

1. AMIの選択 2. インスタンスタイプを選択 3. インスタンスの名称 4. ストレージの構成 5. タグの追加 6. セキュリティグループの選択 7. 確認

ステップ 6: セキュリティグループの設定

セキュリティグループは、インスタンスのネットワークファイアウォールのルールセットです。このページで、特定のトラフィックに対してインスタンスへの到達を許可するルールを追加できます。たとえば、ウェブサーバーをセットアップして、インターネットトラフィックにインスタンスへの到達を許可する場合、HTTPおよびHTTPSポートに無制限のアクセス権限を与えます。新しいセキュリティグループを作成するか、次の既存のセキュリティグループから選択することができます。Amazon EC2 セキュリティグループに関する詳細はこちら。

セキュリティグループの割り当て: ○ 新しいセキュリティグループを作成する

● 既存のセキュリティグループを選択する

セキュリティグループID	名前	説明	アクション
☐ sg-086073a3b9680441f	default	default VPC security group	コピーして新規作成
☐ sg-0e0ba05d8043cd21f	efs-sg	efs	コピーして新規作成
☐ sg-0c3d4c9d234462c9	exastro-rds	rds	コピーして新規作成
☒ sg-06ec461327b067b2	exastro-sg	exastro-HA	コピーして新規作成

タイプ (1)	プロトコル (1)	ポート範囲 (1)	ソース (1)	説明 (1)
HTTP	TCP	80	0.0.0.0/0	
カスタム TCP ルール	TCP	2224	0.0.0.0/0	node
SSH	TCP	22	0.0.0.0/0	
カスタム UDP ルール	UDP	5404 - 5405	0.0.0.0/0	corosync
カスタム TCP ルール	TCP	3121	0.0.0.0/0	pacemaker
MySQL/Aurora	TCP	3306	0.0.0.0/0	RDS
HTTPS	TCP	443	0.0.0.0/0	

9. 確認し、問題なければ「起動」をクリックする ※public-a、public-clに一台ずつ作成する

ステップ 7: インスタンス作成の確認

AMI の詳細

Red Hat Enterprise Linux 8 with High Availability - ami-0c3d4c9d234462c9

Red Hat Enterprise Linux version 8 with High Availability (HVM), EBS General Purpose (SSD) Volume Type

インスタンスタイプ: m5a 80GB m5a

インスタンスタイプ

インスタンスタイプ	EC2	vCPU	メモリ (GiB)	インスタンスストレージ (GiB)	EBS 最適化利用	ネットワークパフォーマンス
m5a medium	-	2	4	EBS のみ	-	Low to Moderate

セキュリティグループ

セキュリティグループID	名前	説明
sg-06ec461327b067b2	exastro-sg	exastro-HA

選択されたすべてのセキュリティグループのインバウンドのルール

タイプ (1)	プロトコル (1)	ポート範囲 (1)	ソース (1)	説明 (1)
HTTP	TCP	80	0.0.0.0/0	
カスタム TCP ルール	TCP	2224	0.0.0.0/0	node
SSH	TCP	22	0.0.0.0/0	
カスタム UDP ルール	UDP	5404 - 5405	0.0.0.0/0	corosync
カスタム TCP ルール	TCP	3121	0.0.0.0/0	pacemaker
MySQL/Aurora	TCP	3306	0.0.0.0/0	RDS
HTTPS	TCP	443	0.0.0.0/0	
すべての ICMP - IPv4	すべて	該当なし	0.0.0.0/0	ping
NFS	TCP	2049	0.0.0.0/0	EFS

AWS設定手順書

10.SSHに必要な鍵を作成する場合はこちらで作成する

既存のキーペアを選択するか、新しいキーペアを作成します。 ×

キーペアは、AWS が保存するパブリックキーとユーザーが保存するプライベートキーファイルで構成されます。組み合わせることで、インスタンスに安全に接続できます。Windows AMI の場合、プライベートキーファイルは、インスタンスへのログインに使用されるパスワードを取得するために必要です。Linux AMI の場合、プライベートキーファイルを使用してインスタンスに SSH で安全に接続できます。

注: 選択したキーペアは、このインスタンスに対して権限がある一連のキーに追加されます。「パブリック AMI から既存のキーペアを削除する」の詳細情報をご覧ください。

新しいキーペアの作成

キーペア名

exasitro-key

キーペアのダウンロード

続行するには、事前にプライベートキーファイル (*.pem ファイル) をダウンロードする必要があります。それを、安全でアクセス可能な場所に保存します。一度作成されたファイルは再度ダウンロードすることはできません。

キャンセル インスタンスの作成

11.作成後アクション→ネットワーキング→ソース宛先/変更をクリックする

インスタンス (1/2) 編集

インスタンスをフィルタリング

search: ita X ファイルをクリア

Name	インスタンス ID	インスタンスの状態	インスタンス...	スタートアップ	アラームの状態	アベイラビリティ...	パブリック IP v4 DNS	アクション
ita-ha-sv02	i-064dfc3b317782f2d	実行中	電圧	i2.medium	2/2 のチェックに合格	アラーム...	+	ネットワークインターフェイスをアタッチ
ita-ha-sv01	i-02bcb6f95a09a200c	実行中	電圧	i2.medium	2/2 のチェックに合格	アラーム...	+	ネットワークインターフェイスをデタッチ

ネットワークインターフェイスをアタッチ

ネットワークインターフェイスをデタッチ

ソース/宛先チェックを変更

インスタンスの状態を管理

インスタンスの設定

ネットワーク

セキュリティ

イメージとテンプレート

モニタリングとトラブルシューティング

IP アドレスの管理

12.停止にチェックを入れ、「保存」をクリックする ※2台分実施すること

EC2 > インスタンス > i-064dfc3b317782f2d > 送信元/送信先チェックを変更

送信元/送信先チェック 情報

各 EC2 インスタンスはデフォルトで送信元と送信先のチェックを実行します。インスタンスは、送受信するすべてのトラフィックの送信元または送信先である必要があります。

インスタンス ID

i-064dfc3b317782f2d (ita-ha-sv02)

ネットワークインターフェイス 情報

eni-0b0e4030a10ebda1b

送信元/送信先チェック中 情報

☒ 停止

これが NAT インスタンスの場合は、送信元/送信先のチェックを停止する必要があります。NAT インスタンスは、送信元または送信先がそのものではない場合、トラフィックを送受信する必要があります。

AWS CLI コマンド

```
aws ec2 modify-instance-attribute --instance-id i-064dfc3b317782f2d --no-source-dest-check
```

コピー

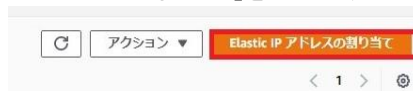
キャンセル 保存

11.Elastic IP

1.左側のメニューのElastic IPをクリックする



2.「IPアドレスの割り当て」をクリックする



AWS設定手順書

3. ネットワークボーダーグループが東京リージョンになっていることを確認し、「割り当て」をクリックする

EC2 > Elastic IP アドレス > Elastic IP アドレスの割り当て

Elastic IP アドレスの割り当て Info

Elastic IP アドレスの設定 Info

ネットワークボーダーグループ Info
Q ap-northeast-1 X

パブリック IPv4 アドレスプール

- Amazon の IPv4 アドレスプール
- AWS アカウントに持ち込むパブリック IPv4 アドレス (プールが見つからなかったためにオプションが無効化されています) [詳細はこちら](#)
- 顧客所有の IPv4 アドレスのプール (顧客所有のプールが見つからないため、オプションは無効です) [詳細はこちら](#)

グローバル静的 IP アドレス

AWS Global Accelerator は、AWS エッジロケーションからのエニーキャストを使用して世界中で発表されたグローバル静的 IP アドレスを提供するため、Amazon のグローバルネットワークを使用することで、ユーザートラフィックの可用性を向上しレイテンシーを低減させることができます。 [詳細はこちら](#)

Accelerator の作成

タグ - オプション

タグは、ユーザーが AWS リソースに割り当てるラベルです。各タグはキーとオプションの値で構成されています。タグは、リソースの検索とフィルタリング、および AWS のコストの追跡に使用できます。

リソースにタグが関連付けられていません。

新規タグを追加

最大 50 個のタグを追加できます

キャンセル **割り当て**

4. IP が生成されたらアクション→Elastic IPアドレスの関連付けをクリックする

◎ Elastic IP アドレスが正常に割り当てられました。
Elastic IP アドレス 18.176.97.211 この Elastic IP アドレスを関連付ける

Elastic IP アドレス (1/1)

Elastic IP アドレスをフィルタリング

パブリック IPv4 アドレス: 18.176.97.211 X フィルターをクリア

	Name	割り当てられた IPv...	タイプ	割り当て ID	関連付けられたインス...	プライベート IP ア...		
☒	exastro-EIP		パブリック IP	eipalloc-0b4c21a8ac8c144a2	-	-	-	ネットワークインター

アクション ▲

Elastic IP アドレスの割り当て

Elastic IP アドレスの解放

Elastic IP アドレスの関連付け

Elastic IP アドレスの関連付けの解除

AWS設定手順書

5. インスタンスを選択し(public-aで動作しているインスタンス)、プライベートIPアドレスを入力し(候補に出るのでそれを選択)「関連付ける」をクリックする

EC2 > Elastic IP アドレス > Elastic IP アドレスの関連付け

Elastic IP アドレスの関連付け

この Elastic IP アドレスに関連付けるインスタンスまたはネットワークインターフェイスを選択します (18.176.97.211)

Elastic IP アドレス: 18.176.97.211

リソースタイプ
Elastic IP アドレスに関連付けるリソースのタイプを選択します。

- ☒ インスタンス
- ☐ ネットワークインターフェイス

インスタンス
i-028cd99e08e2d8c

プライベート IP アドレス
Elastic IP アドレスを関連付けるプライベート IP アドレスです。
10.0.0.204

再関連付け
Elastic IP アドレスがすでにリソースに関連付けられている場合に、そのアドレスを別のリソースに再度関連付けられるかどうかを指定します。
☐ この Elastic IP アドレスの再関連付けを許可する

キャンセル **関連付ける**

これで全てのAWS上の設定は完了しました。
次は構築手順になります。

構築手順

実行ユーザ / 接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
1.ITAのインストール					
1-1	ITAをインストールする	●	●	以下URLのオールインワン構成のインストールマニュアル参照 https://exastro-suite.github.io/it-automation-docs/learn_ia.html#deploy	
1-2	ITAインストールディレクトリの設定をする	●	●	find /(<インストール資料展開先>)/it-automation-1.8.0 -type f xargs -I{} sed -i -e "s:%%ITA_DIRECTORY%%/(<ITAインストール先ディレクトリ>):g" {}	赤字は、環境に合わせて読み替えること
2.Apacheリソース設定					
2-1	apacheのserver statusを作成する	●	●	cat > /etc/httpd/conf.d/server_status.conf << STAT ExtendedStatus On <Location /server-status> SetHandler server-status Order deny,allow Deny from all Allow from localhost </Location> STAT	コマンドをまとめて実施する
2-2	Apacheサービス停止・無効化する	●	●	systemctl disable httpd systemctl stop httpd	
2-3	httpsサーバ証明書／秘密鍵をmasterからslaveへコピーする	●	●	scp -r -i [<鍵>].pem /etc/pki/tls/certs/(httpsサーバ証明書).cert <slaveサーバのローカルIPアドレス>:/etc/pki/tls/certs/ scp -r -i [<鍵>].pem /etc/pki/tls/certs/(httpsサーバ秘密鍵名).key <slaveサーバのローカルIPアドレス>:/etc/pki/tls/certs/	赤字は、環境に合わせて読み替えること (httpsサーバ証明書／秘密鍵は、ITAサーバインストール時に作成した、ファイル名に読み替えること)

構築手順

実行ユーザ/接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ	設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
3.ITAサービス停止設定				
3-1	ky-serviceを停止する	●●	systemctl stop ky_activatedirectory_roleuser_replication-workflow.service systemctl stop ky_ansible_execute-workflow.service systemctl stop ky_ansible_towermasterSync-workflow.service systemctl stop ky_bulk_excel-workflow.service systemctl stop ky_change_col_to_row.service systemctl stop ky_cmdbmenuanalysis-workflow.service systemctl stop ky_create_er-workflow.service systemctl stop ky_create_param_menu_execute.service systemctl stop ky_data_portability_execute-workflow.service systemctl stop ky_hostgroup_check_loop.service systemctl stop ky_hostgroup_split.service systemctl stop ky_legacy_role_valautostup-workflow.service systemctl stop ky_legacy_role_varsautolistup-workflow.service systemctl stop ky_legacy_valautostup-workflow.service systemctl stop ky_legacy_varsautolistup-workflow.service systemctl stop ky_mail.service systemctl stop ky_pioneer_valautostup-workflow.service systemctl stop ky_pioneer_varsautolistup-workflow.service systemctl stop ky_std_checkcondition-linklist.service systemctl stop ky_std_synchronize-Collector.service systemctl stop ky_std_synchronize-Conductor.service systemctl stop ky_std_synchronize-regularly2.service systemctl stop ky_std_synchronize-regularly.service systemctl stop ky_std_synchronize-symphony.service systemctl stop ky_terraform_execute-workflow.service systemctl stop ky_terraform_checkcondition-workflow.service systemctl stop ky_terraform_varsautolistup-workflow.service systemctl stop ky_terraform_valautosetup-workflow.service	ITAのバージョンによって、存在しないサービスがあります。 その場合は、サービス停止不要です。
3-2	ky-service 自動起動設定を無効にする	●●	systemctl disable ky_activatedirectory_roleuser_replication-workflow.service systemctl disable ky_ansible_execute-workflow.service systemctl disable ky_ansible_towermasterSync-workflow.service systemctl disable ky_bulk_excel-workflow.service systemctl disable ky_change_col_to_row.service systemctl disable ky_cmdbmenuanalysis-workflow.service systemctl disable ky_create_er-workflow.service systemctl disable ky_create_param_menu_execute.service systemctl disable ky_data_portability_execute-workflow.service systemctl disable ky_hostgroup_check_loop.service systemctl disable ky_hostgroup_split.service systemctl disable ky_legacy_role_valautostup-workflow.service systemctl disable ky_legacy_role_varsautolistup-workflow.service systemctl disable ky_legacy_valautostup-workflow.service systemctl disable ky_legacy_varsautolistup-workflow.service systemctl disable ky_mail.service systemctl disable ky_pioneer_valautostup-workflow.service systemctl disable ky_pioneer_varsautolistup-workflow.service systemctl disable ky_std_checkcondition-linklist.service systemctl disable ky_std_synchronize-Collector.service systemctl disable ky_std_synchronize-Conductor.service systemctl disable ky_std_synchronize-regularly2.service systemctl disable ky_std_synchronize-regularly.service systemctl disable ky_std_synchronize-symphony.service systemctl disable ky_terraform_execute-workflow.service systemctl disable ky_terraform_checkcondition-workflow.service systemctl disable ky_terraform_varsautolistup-workflow.service systemctl disable ky_terraform_valautosetup-workflow.service	ITAのバージョンによって、存在しないサービスがあります。 その場合は、設定の無効は不要です。

構築手順

実行ユーザ / 接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
4.EFS設定					
前提	・EFSが作成されていること ・ITAがインストール済みであること				
4-1	東京に時刻設定をする	●	●	timedatectl set-timezone Asia/Tokyo	
4-2	EFSの使用準備をする	●	●	cd /tmp yum -y install git git clone https://github.com/aws/efs-utils yum -y install make rpm-build cd efs-utils make rpm yum -y install ./build/amazon-efs-utils*rpm	
4-3	EFSファイル共有設定をする	●		mount -t efs -o tls [Route53に登録したEFSの名前解決設定の情報]/ /mnt cd /mnt mkdir -p ./{(ITAのインストールパス)}/data_relay_storage/symphony mkdir -p ./{(ITAのインストールパス)}/data_relay_storage/conductor mkdir -p ./{(ITAのインストールパス)}/data_relay_storage/ansible_driver mkdir -p ./{(ITAのインストールパス)}/ita_sessions mkdir -p ./{(ITAのインストールパス)}/ita-root/webconfs/sheets mkdir -p ./{(ITAのインストールパス)}/ita-root/webconfs/users mkdir -p ./{(ITAのインストールパス)}/ita-root/temp mkdir -p ./{(ITAのインストールパス)}/ita-root/uploadfiles mkdir -p ./{(ITAのインストールパス)}/ita-root/webroot/uploadfiles mkdir -p ./{(ITAのインストールパス)}/ita-root/webroot/menus/sheets mkdir -p ./{(ITAのインストールパス)}/ita-root/webroot/menus/users chmod 777 ./{(ITAのインストールパス)}/data_relay_storage/symphony chmod 777 ./{(ITAのインストールパス)}/data_relay_storage/conductor chmod 777 ./{(ITAのインストールパス)}/data_relay_storage/ansible_driver chmod 777 ./{(ITAのインストールパス)}/ita_sessions chmod 755 ./{(ITAのインストールパス)}/ita-root/webconfs/sheets chmod 755 ./{(ITAのインストールパス)}/ita-root/webconfs/users chmod 777 ./{(ITAのインストールパス)}/ita-root/temp chmod 777 ./{(ITAのインストールパス)}/ita-root/uploadfiles chmod 777 ./{(ITAのインストールパス)}/ita-root/webroot/uploadfiles chmod 755 ./{(ITAのインストールパス)}/ita-root/webroot/menus/sheets chmod 755 ./{(ITAのインストールパス)}/ita-root/webroot/menus/users	赤字は、環境に合わせて読み替えること AWSコンソールからEFSを作成してから本手順を実施する

構築手順

実行ユーザ / 接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
4-4	共有ファイルのコピーを実施する	●		<pre> cp -rp /(ITAインストールパス)/data_relay_storage/symphony /mnt/(ITAインストールパス)/data_relay_storage/ cp -rp /(ITAインストールパス)/data_relay_storage/conductor /mnt/(ITAインストールパス)/data_relay_storage/ cp -rp /(ITAインストールパス)/data_relay_storage/ansible_driver /mnt/(ITAインストールパス)/data_relay_storage/ cp -rp /(ITAインストールパス)/ita_sessions /mnt/(ITAインストールパス)/ cp -rp /(ITAインストールパス)/ita-root/webconfs/sheets /mnt/(ITAインストールパス)/ita-root/webconfs/ cp -rp /(ITAインストールパス)/ita-root/webconfs/users /mnt/(ITAインストールパス)/ita-root/webconfs/ cp -rp /(ITAインストールパス)/ita-root/temp /mnt/(ITAインストールパス)/ita-root/ cp -rp /(ITAインストールパス)/ita-root/uploadfiles /mnt/(ITAインストールパス)/ita-root/ cp -rp /(ITAインストールパス)/ita-root/webroot/uploadfiles /mnt/(ITAインストールパス)/ita-root/webroot/ cp -rp /(ITAインストールパス)/ita-root/webroot/menus/sheets /mnt/(ITAインストールパス)/ita-root/webroot/menus/ cp -rp /(ITAインストールパス)/ita-root/webroot/menus/users /mnt/(ITAインストールパス)/ita-root/webroot/menus/ cd /tmp umount /mnt </pre>	
4-5	複数ディレクトリをマウントする設定をする 設定ファイルを作成する	●	●	<pre> vi /etc/systemd/system/mount-nfs-sequentially.service [Unit] Description=Workaround for mounting NFS file systems sequentially at boot time After=remote-fs.target [Service] Type=oneshot ExecStart=/bin/mount -avt nfs4 RemainAfterExit=yes [Install] WantedBy=multi-user.target </pre>	
	上記設定を反映させる	●	●	<pre> systemctl daemon-reload systemctl enable mount-nfs-sequentially.service </pre>	

構築手順

実行ユーザ / 接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
5.RDS設定					
前提	・RDSが作成されていること ・ITAがインストール済みであること				
5-1	RDSに接続する	●		mysql -h [DBのホスト名 (Route53にRDSの名前解決設定の情報)] -P 3306 -u admin -p<adminパスワード> 例)mysql -h rds.exastro-aws.local -P 3306 -u admin -p*****	赤字は、環境に合わせて読み替えること
5-2	ユーザとDBを作成する	●		CREATE USER '<ユーザ名>' IDENTIFIED BY '<パスワード>'; CREATE USER '<ユーザ名>'@'localhost' IDENTIFIED BY '<パスワード>'; CREATE DATABASE [DB名] CHARACTER SET utf8; GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, REFERENCES, INDEX, ALTER, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, EVENT, TRIGGER ON [DB名].* TO '<ユーザ名>'@'%' WITH GRANT OPTION; GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, REFERENCES, INDEX, ALTER, CREATE TEMPORARY TABLES, LOCK TABLES, EXECUTE, CREATE VIEW, SHOW VIEW, CREATE ROUTINE, ALTER ROUTINE, EVENT, TRIGGER ON [DB名].* TO '<ユーザ名>'@'localhost' WITH GRANT OPTION; exit	赤字は、環境に合わせて読み替えること
5-3	DBへのデータ流し込みを実施する	●		mysqldump --default-character-set=utf8 -u <ita_answer.txtで設定したDBユーザー名> -p<ita_answer.txtで設定したDBパスワード> [ita_answer.txtで設定したDB名] > ITA_DB.sql sed -i 's/DEFINER=[^*]***/*/g' ITA_DB.sql mysql --default-character-set=utf8 -h [DBのホスト名 (Route53にEFSの名前解決設定の情報)] -P 3306-u <RDSのマスターユーザ名> -p<RDSのマスターパスワード> [DB名] < ITA_DB.sql 例) mysql --default-character-set=utf8 -h rds.exastro-aws.local -P 3306 -u admin -p***** ITA_DB < ITA_DB.sql	赤字は、環境に合わせて読み替えること

構築手順

実行ユーザ / 接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
5-4	DBの接続設定を実施する				
	MariaDB接続情報設定	●	●	①以下の文字列をbase64エンコード、rot13暗号を行った文字列を作成してください。 <code>echo -ne "mysql:dbname=[DB名];host=[DBのホスト名(Route53にRDSの名前解決設定の情報)]" base64 tr 'A-Za-z' 'N-ZA-Mn-z-a-m'</code> ②作成した文字列を以下のファイルに書き込んでください。※記載済みの値は削除してください <code>vi /([ITAインストール先ディレクトリ])/ita-root/conf/commonconfs/db_connection_string.txt</code>	例: ITAのDB名が「ITA_DB」、Route53に登録しているDNS名が <code>rds.exastro-aws.local</code> の場合、以下のコマンドでbase64エンコード、rot13暗号を行った文字列を取得できます。 <pre>echo -ne "mysql:dbname=ITA_DB;host=rds.exastro-aws.local" base64 tr 'A-Za-z' 'N-ZA-Mn-z-a-m'</pre> 赤字は、環境に合わせて読み替えること
	MariaDBのユーザ情報設定	●	●	①MariaDBのユーザ名をbase64エンコード、rot13暗号を行った文字列を作成してください。 <code>echo -ne "<ユーザ名>" base64 tr 'A-Za-z' 'N-ZA-Mn-z-a-m'</code> ②作成した文字列を以下のファイルに書き込んでください。※記載済みの値は削除してください <code>vi /([ITAインストール先ディレクトリ])/ita-root/conf/commonconfs/db_username.txt</code>	例: MariaDBのユーザ名が「ITA_USER」の場合、以下のコマンドでbase64エンコード、rot13暗号を行った文字列を取得できます。 <pre>echo -ne "ITA_USER" base64 tr 'A-Za-z' 'N-ZA-Mn-z-a-m'</pre> 赤字は、環境に合わせて読み替えること
	MariaDBのパスワード情報設定	●	●	①MariaDBのパスワードをbase64エンコード、rot13暗号を行った文字列を作成してください。 <code>echo -ne "<パスワード>" base64 tr 'A-Za-z' 'N-ZA-Mn-z-a-m'</code> ②作成した文字列を以下のファイルに書き込んでください。※記載済みの値は削除してください <code>vi /([ITAインストール先ディレクトリ])/ita-root/conf/commonconfs/db_password.txt</code>	例: MariaDBのパスワードが「ITA_PASSWD」の場合、以下のコマンドでbase64エンコード、rot13暗号を行った文字列を取得できます。 <pre>echo -ne "ITA_PASSWD" base64 tr 'A-Za-z' 'N-ZA-Mn-z-a-m'</pre> 赤字は、環境に合わせて読み替えること
5-5	MariaDBを停止する	●	●	<code>systemctl disable mariadb</code> <code>systemctl stop mariadb</code>	
6.Pacemaker使用準備					
6-1	awscliをインストールする	●	●	<code>yum -y install awscli</code>	
6-2	ユーザー作成で作成したcredentials.csvを確認する	●	●	<code>aws configure</code> AWS Access Key ID [None]: ***** AWS Secret Access Key [None]: ***** Default region name [None]: ap-northeast-1 Default output format [None]: json	赤字は、環境に合わせて読み替えること IAM作成時にダウンロードしたcsvを参照すること
6-3	ホスト名とIPをhostsに登録する	●	●	<code>uname -n</code> <code>vi /etc/hosts</code> ローカルIP ホスト名 ローカルIP ホスト名 例) 10.0.0.204 <ita-ha-sv01のホスト名> 10.0.1.61 <ita-ha-sv02のホスト名>	赤字は、環境に合わせて読み替えること

構築手順

実行ユーザ / 接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
7.Pacemaker設定					
7-1	pcs の管理アカウントとなるユーザーID「hacluster」のパスワードを設定する	●	●	passwd hacluster 新しいパスワード:xxxxxx 新しいパスワードを再入力してください:xxxxxx passwd: すべての認証トークンが正しく更新できました。	赤字は、環境に合わせて読み替えること
7-2	pcsdサービスを起動し、システムの起動時に pcsd が有効になるよう設定する	●	●	systemctl enable pcsd systemctl start pcsd systemctl status pcsd	
7-3	クラスターを認証する	●		pcs host auth <ita-ha-sv01のホスト名> <ita-ha-sv02のホスト名> Username: hacluster Password:passwd haclusterで設定したパスワード	赤字は、環境に合わせて読み替えること
7-4	クラスターを指定してスタートする	●		pcs cluster setup aws-cluster --start <ita-ha-sv01のホスト名> <ita-ha-sv02のホスト名> --force	
7-5	クラスターを自動起動するように設定をする	●		pcs cluster enable --all	
7-6	STONITH (Shoot The Other Node In The Head) オプションを無効にする	●		pcs property set stonith-enabled=false	
7-7	クラスターが必要最低限の数に達していない場合でも特別な動作は行わず全ノードの管理を続	●		pcs property set no-quorum-policy=ignore	
7-8	属性待ち時間を設定する	●		pcs property set transition-delay="0s"	
7-9	自動フェイルバックを無効にする	●		pcs resource defaults resource-stickiness="INFINITY" migration-threshold="1"	
7-10	corosync設定をする	●	●	cp -p /usr/lib/systemd/system/corosync.service /etc/systemd/system/ sed -i 's/^#Restart=on:*/Restart=on-failure/' /etc/systemd/system/corosync.service sed -i 's/^#RestartSec=*/RestartSec=70/' /etc/systemd/system/corosync.service	
7-11	Pacemakerの内部プロセスが異常になった場合もノード故障として取り扱う	●		sed -i 's/^#¥ PCMK¥_fail¥_fast:*/PCMK_fail_fast=yes/' /etc/sysconfig/pacemaker	
7-12	corosync設定をする	●	●	cp -p /usr/lib/systemd/system/pacemaker.service /etc/systemd/system sed -i 's/^#¥ ExecStopPost=¥/bin¥/sh:*/ExecStopPost=¥/bin¥/sh -c 'pidof crmd ¥ ¥ killall -TERM corosync'/' /etc/systemd/system/pacemaker.service	
7-13	IP付け替えの設定をする	●	●	vi /etc/sysconfig/pacemaker /etc/sysconfig/pacemakerの中身 # Set the options to pass to valgrind, when valgrind is enabled. See # valgrind(1) man page for details. "--vgdb=no" is specified because # pacemaker-execd can lower privileges when executing commands, which would # otherwise leave a bunch of unremovable files in /tmp. VALGRIND_OPTS="--leak-check=full --trace-children=no --vgdb=no --num-callers=25 --log- file=/var/lib/pacemaker/valgrind-%p --suppressions=/usr/share/pacemaker/tests/valgrind- pcmk.suppressions --gen-suppressions=all" AWS_DEFAULT_REGION=ap-northeast-1 ←記載部分	赤字は、環境に合わせて読み替えること (リージョンが東京の場合は、末尾に AWS_DEFAULT_REGION=ap-northeast-1 と記載)

構築手順

実行ユーザ / 接続条件

rootユーザで実行すること

rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
7-14	httpd設定をする	●		pcs resource create httpd systemd:httpd ¥ op monitor interval=60 timeout=100 ¥ op start interval=0s timeout=100 ¥ op stop interval=0s timeout=100	
7-15	eip設定をする	●		pcs resource create eip ocf:heartbeat:awseip ¥ elastic_ip="ElasticIPのIPアドレス" awscli="\$\${which aws}" ¥ allocation_id=ElasticIPの割り当てID ¥ op start timeout="120s" interval="0s" on-fail="restart" ¥ op monitor timeout="20s" interval="30s" on-fail="restart" ¥ op stop timeout="120s" interval="0s" on-fail="block"	赤字は、環境に合わせて読み替えること
7-16	httpd、eipリソースグループ設定をする	●		pcs resource group add exastro httpd eip	
7-17	EFSリソース設定をする	●		pcs resource create symphony Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/data_relay_storage/symphony" directory="/(ITAインストール先ディレクトリ)/data_relay_storage/symphony" fstype="nfs" pcs resource create conductor Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/data_relay_storage/conductor" directory="/(ITAインストール先ディレクトリ)/data_relay_storage/conductor" fstype="nfs" pcs resource create ansible_driver Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/data_relay_storage/ansible_driver" directory="/(ITAインストール先ディレクトリ)/data_relay_storage/ansible_driver" fstype="nfs" pcs resource create ita_sessions Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/ita_sessions" directory="/(ITAインストール先ディレクトリ)/ita_sessions" fstype="nfs" pcs resource create webconfs_sheets Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/ita-root/webconfs/sheets" directory="/(ITAインストール先ディレクトリ)/ita-root/webconfs/sheets" fstype="nfs" pcs resource create webconfs_users Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/ita-root/webconfs/users" directory="/(ITAインストール先ディレクトリ)/ita-root/webconfs/users" fstype="nfs" pcs resource create temp Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/ita-root/temp" directory="/(ITAインストール先ディレクトリ)/ita-root/temp" fstype="nfs" pcs resource create uploadfiles Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/ita-root/uploadfiles" directory="/(ITAインストール先ディレクトリ)/ita-root/uploadfiles" fstype="nfs" pcs resource create webroot_uploadfiles Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/ita-root/webroot/uploadfiles" directory="/(ITAインストール先ディレクトリ)/ita-root/webroot/uploadfiles" fstype="nfs" pcs resource create menus_sheets Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/ita-root/webroot/menus/sheets" directory="/(ITAインストール先ディレクトリ)/ita-root/webroot/menus/sheets" fstype="nfs" pcs resource create menus_users Filesystem device="[Route53に登録したEFSの名前解決設定の情報]:/(ITAインストール先ディレクトリ)/ita-root/webroot/menus/users" directory="/(ITAインストール先ディレクトリ)/ita-root/webroot/menus/users" fstype="nfs"	赤字は、環境に合わせて読み替えること

構築手順

実行ユーザ/接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
7-18	EFSリソースグループ設定をする	●		pcs resource group add efsmount symphony conductor ansible_driver ita_sessions webconfs_sheets webconfs_users temp uploadfiles webroot_uploadfiles menus_sheets menus_users	
7-19	各kyファイルリソース設定をする	●		pcs resource create ky_activatedirectory_roleuser_replication-workflow systemd:ky_activatedirectory_roleuser_replication-workflow ¥ op monitor interval=60 timeout=100 ¥ op start interval=0s timeout=100 ¥ op stop interval=0s timeout=100 pcs resource create ky_aws_s3_bucket systemd:ky_aws_s3_bucket ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_aws_s3_bucket_replication systemd:ky_aws_s3_bucket_replication ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_bulk_excel-workflow systemd:ky_bulk_excel-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_change_col_to_row systemd:ky_change_col_to_row ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_cmdbmenuanalysis-workflow systemd:ky_cmdbmenuanalysis-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_create_er-workflow systemd:ky_create_er-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_create_param_menu_execute systemd:ky_create_param_menu_execute ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_data_portability_execute-workflow systemd:ky_data_portability_execute-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_hostgroup_check_loop systemd:ky_hostgroup_check_loop ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_hostgroup_split systemd:ky_hostgroup_split ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60	

構築手順

実行ユーザ / 接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
7-19	各kyファイルリソース設定をする (前頁の続き)			pcs resource create ky_legacy_role_valautostup-workflow systemd:ky_legacy_role_valautostup-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_legacy_role_varsautolistup-workflow systemd:ky_legacy_role_varsautolistup-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_legacy_valautostup-workflow systemd:ky_legacy_valautostup-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_legacy_varsautolistup-workflow systemd:ky_legacy_varsautolistup-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_pioneer_valautostup-workflow systemd:ky_pioneer_valautostup-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_pioneer_varsautolistup-workflow systemd:ky_pioneer_varsautolistup-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_std_checkcondition-linklist systemd:ky_std_checkcondition-linklist ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_std_synchronize-Conductor systemd:ky_std_synchronize-Conductor ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_std_synchronize-regularly systemd:ky_std_synchronize-regularly ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_std_synchronize-regularly2 systemd:ky_std_synchronize-regularly2 ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_std_synchronize-symphony systemd:ky_std_synchronize-symphony ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_terraform_execute-workflow systemd:ky_terraform_execute-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60	

構築手順

実行ユーザ / 接続条件
rootユーザで実行すること
rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
7-19	各kyファイルリソース設定をする (前頁の続き)			pcs resource create ky_terraform_checkcondition-workflow systemd:ky_terraform_checkcondition-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_terraform_varsautolistup-workflow systemd:ky_terraform_varsautolistup-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60 pcs resource create ky_terraform_valautosetup-workflow systemd:ky_terraform_valautosetup-workflow ¥ op monitor interval=30 timeout=60 ¥ op start interval=0s timeout=60 ¥ op stop interval=0s timeout=60	
7-20	kyリソースのグループ化を実施する	●		pcs resource group add ky_services ky_activatedirectory_roleuser_replication-workflow ky_ansible_execute-workflow ky_ansible_towermasterSync-workflow ky_bulk_excel-workflow ky_change_col_to_row ky_cmdbmenuanalysis-workflow ky_create_er-workflow ky_create_param_menu_execute ky_data_portability_execute-workflow ky_hostgroup_check_loop ky_hostgroup_split ky_legacy_role_valautostup-workflow ky_legacy_role_varsautolistup-workflow ky_legacy_valautostup-workflow ky_legacy_varsautolistup-workflow ky_pioneer_valautostup-workflow ky_pioneer_varsautolistup-workflow ky_std_checkcondition-linklist ky_std_synchronize-Conductor ky_std_synchronize-regularly ky_std_synchronize-regularly2 ky_std_synchronize-symphony ky_terraform_execute-workflow ky_terraform_checkcondition-workflow ky_terraform_varsautolistup-workflow ky_terraform_valautosetup-workflow	
7-21	これまでに作成したリソースを指定し、リソースグループ起動/停止の順序を指定する	●		pcs constraint order set efsmount ky_services exastro	
7-22	これまでに設定したリソースが同ノードで起動する制限を指定する	●		pcs constraint colocation set efsmount ky_services exastro	
7-23	HAソフトのサービスを開始する	●	●	systemctl enable corosync systemctl enable pacemaker systemctl start corosync systemctl start pacemaker	

構築手順

実行ユーザ/接続条件

rootユーザで実行すること

rootユーザがsshログイン可能であること

No.	作業項目	サーバ		設定内容 (※master、slaveの両方が対象の場合は、並行して設定してください。)	備考
		master	slave		
7-24	クラスターのステータスを確認する	●	●	pcs status (表示例) Cluster name: aws-cluster Cluster Summary: * Stack: corosync * Current DC: <ita-ha-sv01のホスト名> (version 2.0.5-9.el8_4.1-ba59be7122) - partition with quorum * Last updated: Thu Jun 10 06:46:16 2021 * Last change: Wed Jun 9 18:39:29 2021 by root via cibadmin on <ita-ha-sv01のホスト名> * 2 nodes configured ←左記の表示であること * 34 resource instances configured Node List: * Online: [<ita-ha-sv01のホスト名> <ita-ha-sv02のホスト名>] ←master,slaveがOnlineであること Full List of Resources: * Resource Group: exastro: * eip (ocf::heartbeat:awsaip): Started <ita-ha-sv01のホスト名> ←masterがStartedであること * httpd (systemd:httpd): Started <ita-ha-sv01のホスト名> * Resource Group: efsmount: * symphony (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * conductor (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * ansible_driver (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * ita_sessions (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * webconfs_sheets (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * webconfs_users (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * temp (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * uploadfiles (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * webroot_uploadfiles (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * menus_sheets (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * menus_users (ocf::heartbeat:Filesystem): Started <ita-ha-sv01のホスト名> * Resource Group: ky_services: ~~~~~省略~~~~~ Daemon Status: - corosync: active/enabled - pacemaker: active/enabled - pcsd: active/enabled	赤字は、環境に合わせて読み替える
7-25	ITAの接続を確認する	●		以下のURLより、ログイン画面にアクセスする http(s)://(ElasticIP)	赤字は、環境に合わせて読み替える
7-26	MariaDBの削除を実施する	●	●	yum list installed grep -i mariadb yum remove mariadb.x86_64 mariadb-backup.x86_64 mariadb-common.x86_64 mariadb-connector-c.x86_64 mariadb-connector-c-config.noarch mariadb-errmsg.x86_64 mariadb-gssapi-server.x86_64 mariadb-server.x86_64 mariadb-server-utils.x86_64	前項の手順について、ITAへ正常にアクセスが可能であることを、確認してから削除を行う。

作業終了