



CloudSystemテンプレート 連携施策

第1.0版 (ITAバージョン1.5.0版)
Exastro developer

目次

1. 施策概要
2. CSテンプレート実行例
 1. 環境の準備
 2. 【手順①】 パラメータの編集
 3. 【手順②】 システム構築の実行(= Stack作成)
 4. 出来上がったシステムを確認
3. 参考 - 通常はユーザが見ることはないメニュー -
4. 付録A - 出来上がったシステムのEC2にAnsibleを使って設定変更する -

施策概要

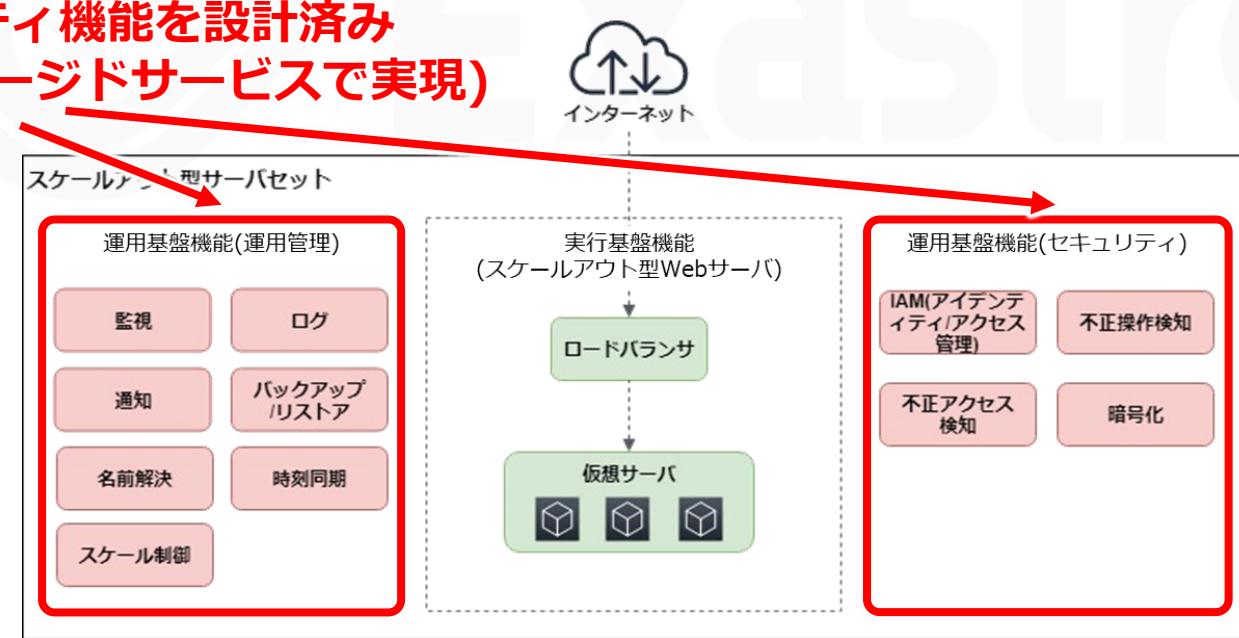


CS (CloudSystem) テンプレートとは？

CSテンプレートは「**設計済みクラウドシステムモデルのIaC(Infrastructure as a Code)**」です
運用管理やセキュリティなど運用上必須の機能が構築済みであることが特徴です

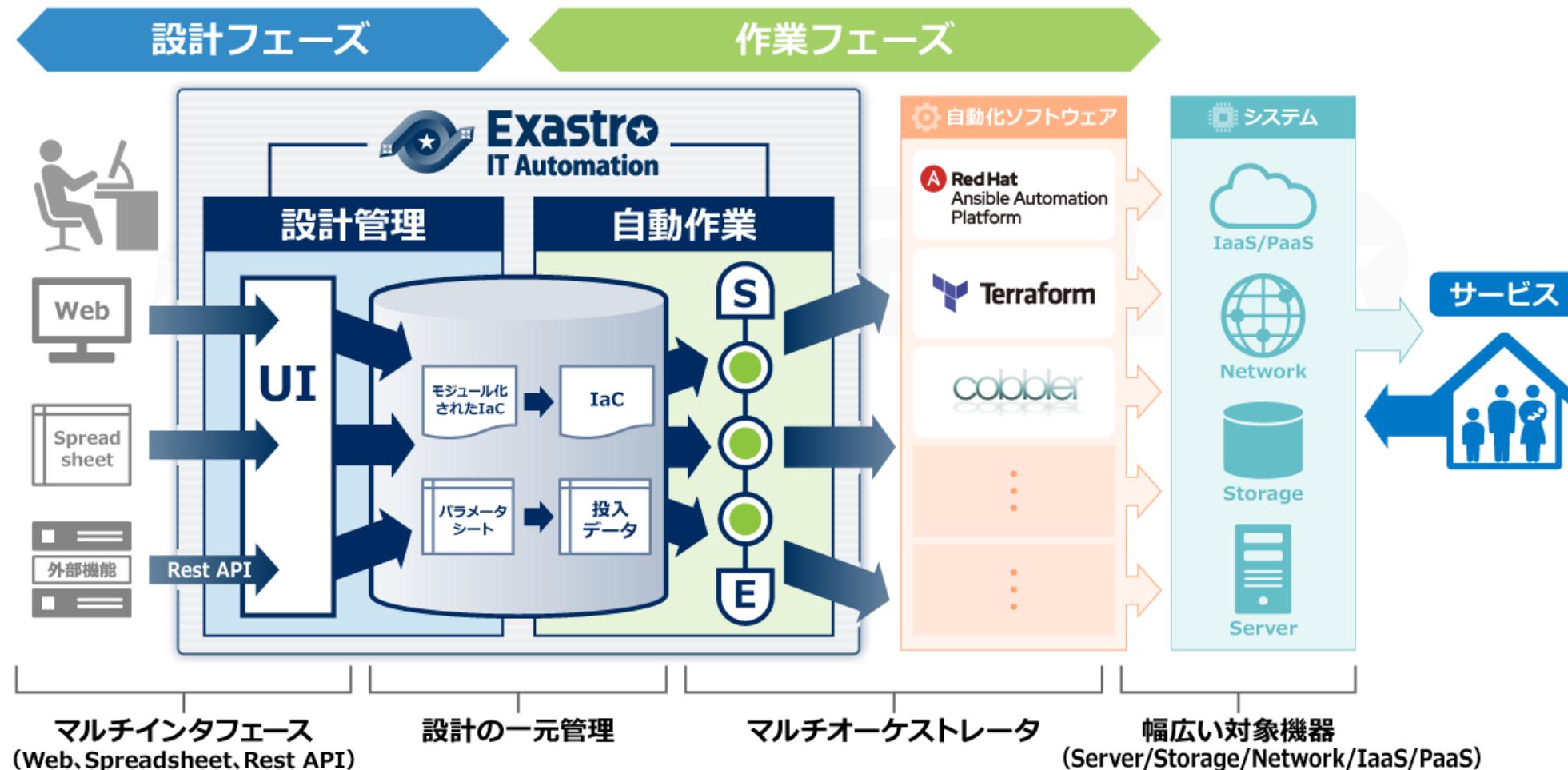
機能分類	概要
実行基盤機能	本テンプレートデモでは、ロードバランサ配下のスケールアウト型Webサーバを構成
運用基盤機能（運用管理）	監視、ログなど、プラットフォームの正常性を維持する機能群
運用基盤機能（セキュリティ）	アクセス制御など、プラットフォームのセキュリティを確保する機能群

**運用管理・セキュリティ機能を設計済み
(サーバを使わずマネージドサービスで実現)**



実行基盤機能は最低限の機能 (オートスケール+LB) のみを具備

Exastro IT Automationは 「システム情報をデジタル管理するためのフレームワーク」です



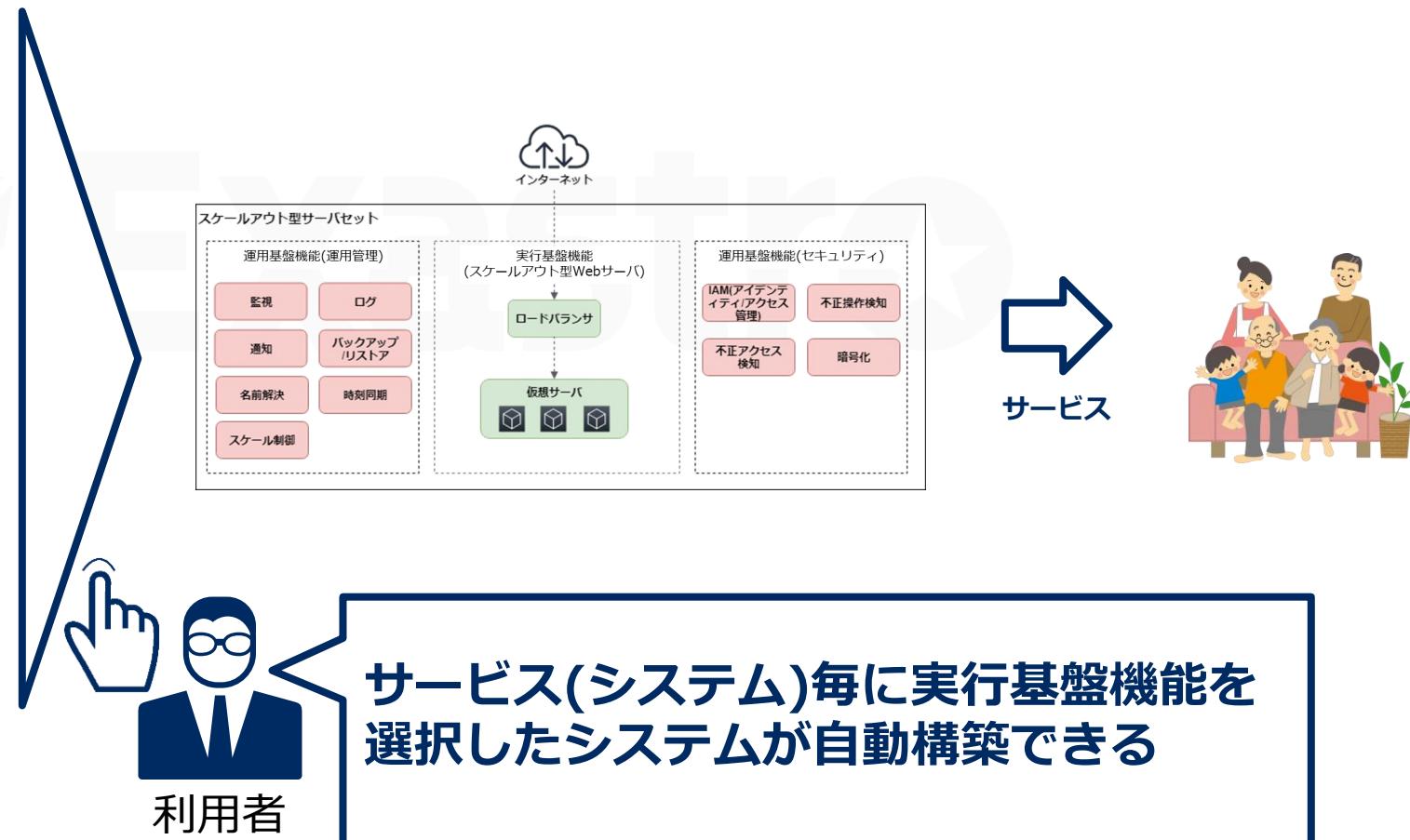
CS Template



Exastro
IT Automation

Exastroを活用したCSテンプレートの実行(1/3)

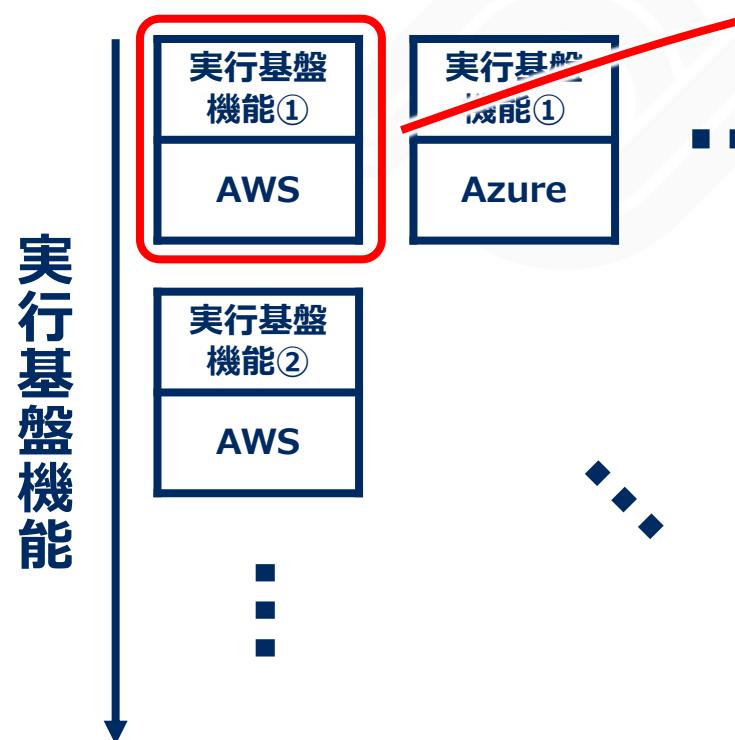
設計済みのCSテンプレートをただ提供するだけでなく、自動構築ツール
"Exastro IT Automation"にCSテンプレートを取り込んだ状態で、
パラメータ調整+実行の2アクションでシステムを提供することが可能です



CSテンプレートとExastroでの実行との関連イメージは下記の通りです。

CSテンプレート施策の広がり

様々なクラウドへの適用



Exastro ITAでの実行

Symphony

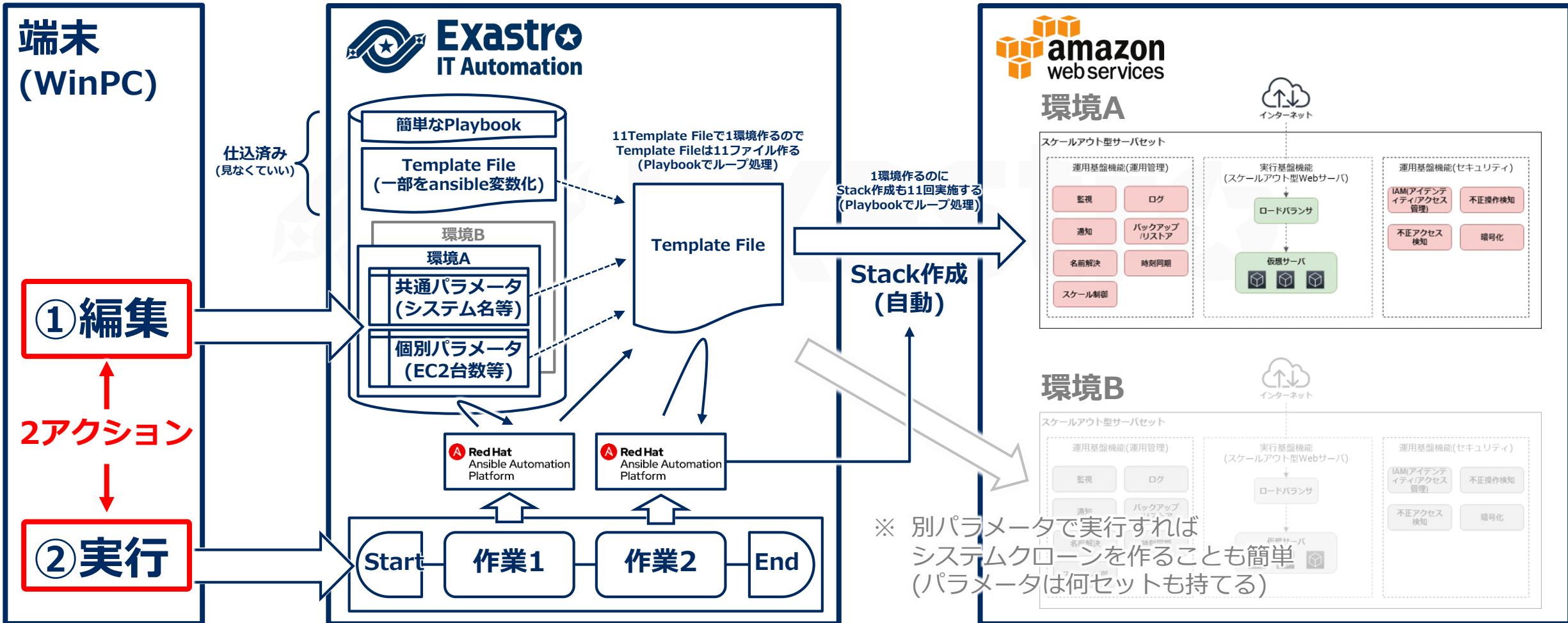


利用者



Exastroを活用したCSテンプレートの実行(3/3)

Exastro IT AutomationにCSテンプレートを組み込むことにより、以下のような流れでAWS上の環境自動構築が実現可能です。



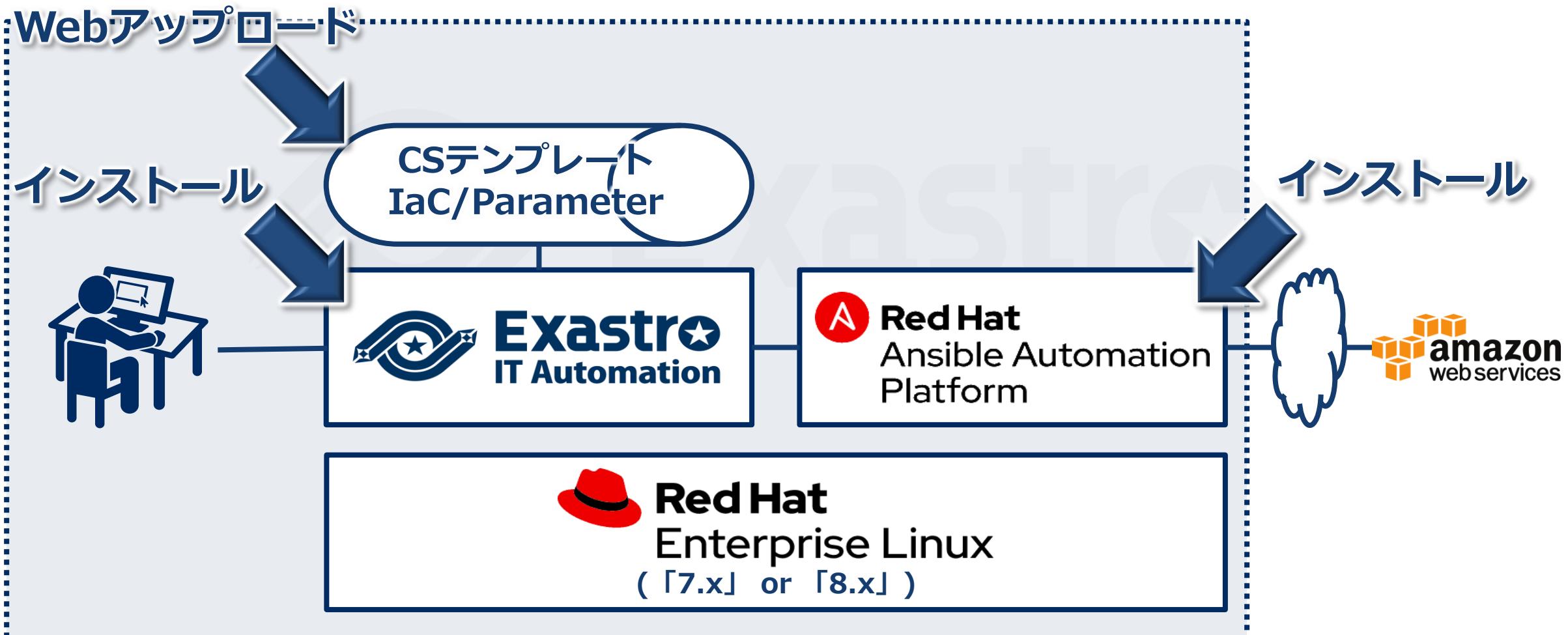
CSテンプレート実行例 環境の準備



【CSテンプレート実行例】環境の準備

以下のような環境を準備します。

(詳細は「CloudSystemテンプレート導入手順書」の導入準備と導入作業を参照のこと。)



CSテンプレート実行例

【手順①】 パラメータの編集



【手順①】 パラメータの編集 (1/3)

Exastro IT Automation パラメータ管理

ようこそ[インフラ管理者]さん
ログインID [infra-admin]

パネル表示 : 中 パスワード変更 ログアウト

Menu

メインメニュー

共通パラメータ

AutoScaleパラメータ

S3パラメータ

SNSパラメータ

CloudTrailパラメータ

Networkパラメータ

SecurityGroupパラメータ

Bastionパラメータ

VPCflowlogsパラメータ

CloudWatchAlarmパラメータ

GuardDutyパラメータ

パラメータ管理

構築/更新シナリオ

削除シナリオ

ドキュメント管理

AWSアカウント管理

Teams連携管理

管理コンソール

基本コンソール

メニュー作成

ホストグループ管理

Ansible共通

Ansible-Legacy

マスタ管理

【参照用】パラメータ

【参照用】構築シナリオ

【参照用】削除シナリオ

代入値管理

【ホスト用】代入値

このパネルをクリックすると…

パラメータのリストが見えます

【手順①】 パラメータの編集 (2/3)

ようこそ[インフラ管理者]さん
ログインID [infra-admin]
[パスワード変更](#) [ログアウト](#)

Exastro IT Automation パラメータ管理

Menu

メインメニュー

共通パラメータ (選択)

AutoScaleパラメータ

S3パラメータ

SNSパラメータ

CloudTrailパラメータ

Networkパラメータ

SecurityGroupパラメータ

Bastionパラメータ

VPCflowlogsパラメータ

CloudWatchAlarmパラメータ

GuardDutyパラメータ

管理者に連絡

説明

表示フィルタ

一覧/更新

△閉じる

▽開く

▽開く

△閉じる

オペレーション

更新	廃止	No.	オペレーション名	基準日時	実施予定日時	最終実行日時	システム名	AWSリージョン	システム環境	システム概要タグ	システム所有者タグ	システム状態	最終更新日時	最終更新者
更新	廃止	1	環境A	2020/06/18 11:28	2020/05/28 11:00	2020/06/18 11:28	EnvA-DemoSystem	ap-northeast-1	Development	SI-Template	NEC	SPSI	2020/06/18 10:36:06	インフラ管理者
更新	廃止	2	環境B	2020/06/18 11:27	2020/06/01 12:00	2020/06/18 11:27	EnvB-DemoSystem	ap-northeast-1	Development	SI-Template	NEC	SPSI	2020/06/10 14:09:04	システム管理者

フィルタ結果件数: 2

Excel出力

「システム名」等の共通パラメータはここで指定します
(複数環境のパラメータセットを管理できます)

登録

全件ダウンロードとファイルアップロード編集

変更履歴

【手順①】 パラメータの編集 (3/3)

Exastro IT Automation パラメータ管理 ようこそ[インフラ管理者]さん
ログインID [infra-admin] パスワード変更 ログアウト

Menu

メインメニュー

共通パラメータ

AutoScaleパラメータ

S3パラメータ

SNSパラメータ

CloudTrailパラメータ

Networkパラメータ

SecurityGroupパラメータ

Bastionパラメータ

VPCflowlogsパラメータ

CloudWatchAlarmパラメータ

GuardDutyパラメータ

説明

表示フィルタ

一覧/更新

△閉じる

オペレーション

更新	廃止	No.	オペレーション名	基準日時	実施予定日時	最終実行日時	ヘルスチェックパス	ブロックデバイス	ブロックデバイスピリューム	AMI	最終更新日時	最終更新者
更新	廃止	1	環境A	2020/06/18 11:28	2020/05/28 11:00	2020/06/18 11:28	/index.html	/dev/xvda	8	ami-011facbea5ec0363b	2020/06/10 14:13:17	システム管理者
更新	廃止	2	環境B	2020/06/18 11:27	2020/06/01 12:00	2020/06/18 11:27	/index.html	/dev/xvda	8	ami-011facbea5ec0363b	2020/06/10 14:13:17	システム管理者

フィルタ結果件数: 2

Excel出力

個別パラメータはカテゴリ毎に作ってます
例えば「オートスケールの最小/最大インスタンス数」
などの設定値を編集できます。
(複数環境のパラメータセットが管理できることは、
どの個別パラメータも共通パラメータと同様です。)

登録

全件ダウンロード CSVファイルダウンロード 編集

変更履歴

管理者に連絡

CSテンプレート実行例

【手順②】 システム構築の実行(= Stack作成)

[補足]Stackとは....

システムで必要とするAWSリソースのかたまり。

CloudFormationではリソースをStackという単位で
管理することで環境の生成・破棄をシンプルにしている。

Stackを作成することで、AWSサービスの構築・設定が
行われる。

【手順②】システム構築の実行(= Stack作成) (1/4)

ようこそ[インフラ管理者]さん
ログインID [infra-admin]
パネル表示: 中 パスワード変更 ログアウト

Exastro IT Automation 基本コンソール

Menu

メインメニュー

機器一覧

投入オペレーション一覧

Symphonyクラスター一覧

Symphonyクラス編集

Symphony作業実行

Symphony作業確認

Symphony作業一覧

定期作業実行

パラメータ管理

構築/更新シナリオ

削除シナリオ

ドキュメント管理

AWSアカウント管理

Teams連携管理

管理コントローラ

基本コンソール

メニュー作成

ホストグループ

Ansible共通

Ansible-Legacy

マスタ管理

【参照用】パラメータ

【参照用】構築シナリオ

【参照用】削除シナリオ

代入値管理

【ホスト用】代入値

このパネルをクリックし…

「Symphony作業実行」クリック
(SymphonyはExastro用語で「ジョブフロー」のことです)

【手順②】システム構築の実行(= Stack作成) (2/4)

The screenshot shows the Exastro basic console interface. On the left is a navigation sidebar with various menu items like 'Menu', '説明' (Description), 'スケジューリング' (Scheduling), 'Symphony[フィルタ]' (Symphony [Filter]), 'Symphony[一覧]' (Symphony [List]), 'Symphony作業実行' (Symphony Job Execution), 'Symphony作業確認' (Symphony Job Confirmation), 'Symphony作業一覧' (Symphony Job List), and '定期作業実行' (Scheduled Job Execution). The main area displays a table of job execution logs. The first row, which is highlighted with a red box and has a red arrow pointing to it from the text 'Symphony(=ジョブフロー)を選択', contains the operation '1 オートスケールWebサーバ (構築/更新)' (Auto Scaling Web Server (Create/Update)). The second row contains the operation '3 オートスケールWebサーバ (削除)' (Delete Auto Scaling Web Server). The third row contains the operation '4 GuardDuty (構築/更新)' (GuardDuty (Create/Update)). The fourth row contains the operation '5 GuardDuty (削除)' (Delete GuardDuty). The fifth row contains the operation '6 AWS管理者 (構築/更新)' (AWS Manager (Create/Update)). The bottom section of the screenshot shows a filter for 'オペレーション[フィルタ]' (Operation Filter) and a list of environments: '環境A' (Environment A), '環境B' (Environment B), and '環境C' (Environment C). The first item in the environment list, '環境A', is also highlighted with a red box and has a red arrow pointing to it from the text 'Operation(=パラメータのセット)を選択 例えば「環境A」を選択'.

操作	説明	実行日時	実行者
1 オートスケールWebサーバ (構築/更新)	オートスケールWebサーバを構築または更新を実行します。 AWS CloudFormationでStackが未作成の場合、構築処理が実行されます。 Stackが作成済みかつパラメータに変更がある場合、更新処理が実行されます。 ◆ロール名「インフラ管理者」が付与されたITAユーザーが実行。	2020/06/09 16:41:00	データポータビリティプロシージャ
3 オートスケールWebサーバ (削除)	実行するPlaybookを変更する際はMovementを「現成し、「イメージファイルをWebサーバへ設定」と変えてください。 オートスケールWebサーバを削除します。 S3バケットも削除対象。	2020/06/09 16:41:31	データポータビリティプロシージャ
4 GuardDuty (構築/更新)	◆ロール名「インフラユーザー」が付与されたITAユーザーが実行。 GuardDuty構築または更新を実行します。 AWS CloudFormationでStackが未作成の場合、構築処理が実行されます。 Stackが作成済みかつパラメータに変更がある場合、更新処理が実行されます。	2020/06/09 16:41:45	データポータビリティプロシージャ
5 GuardDuty (削除)	◆ロール名「インフラ管理者」が付与されたITAユーザーが実行。 構築されたGuardDutyを削除します。	2020/06/09 16:41:55	データポータビリティプロシージャ
6 AWS管理者 (構築/更新)	◆ロール名「インフラ管理者」が付与されたITAユーザーが実行。 AWSの管理者ユーザー、管理者グループの作成または更新を実行します。	2020/06/09 16:42:05	データポータビリティプロシージャ

Symphony(=ジョブフロー)を選択
「オートスケールWebサーバ (構築/更新)」を選択

Operation(=パラメータのセット)を選択
例えば「環境A」を選択

【手順②】システム構築の実行(= Stack作成) (3/4)

Menu

メインメニュー

機器一覧

投入オペレーション一覧

Symphonyクラス一覧

Symphonyクラス編集

Symphony作業実行

Symphony作業確認

Symphony作業一覧

定期作業実行

Symphony実行

説明 AutoScaleWebシステムを構築します。
(対象: 運用基盤機能+実行基盤機能+デモ用のWebコンテンツ)

オペレーションID 17
オペレーション名 商用環境A

start

テンプレ作成 / オートスケールWeb / 構築・更新

CF実行 / オートスケールWeb / 構築・更新

機器一覧を同期

Teams通知(オートスケールWeb)

end

実行

Template Fileの変数にパラメータの値を埋め込みます
(Stack分ループします)

埋め込み済みのTemplate Fileを使ってStackを作成します
(Stack分ループします)

Webサーバ(EC2)をディスカバリーし
Exastro ITAの機器一覧に登録します

構築完了をSNS(Teams)で通知します

実行をクリック

【手順②】システム構築の実行(= Stack作成) (4/4)

正常終了すればシステム完成です

△閉じる

説明 オートスケールWebサーバを構築または更新を実行します。
AWS CloudFormationでStackが未作成の場合、構築処理が実行されます。
Stackが作成済みかつパラメータに変更がある場合、更新処理が実行されます。

SymphonyインスタンスID 322
Symphonyクラス名 オートスケールWebサーバ (構築/更新)

▼ start

[1] ● テンプレ作成 / オートスケールWeb / 構築・更新
DONE 開始日時 2020/06/18 16:16:03 終了日時 2020/06/18 16:16:21

[2] ● CF実行 / オートスケールWeb / 構築・更新
RUNNING 開始日時 2020/06/18 16:16:24 終了日時

[3] ● 機器一覧を同期(インフラ管理者)
[4] ● Teams通知(オートスケールWeb)

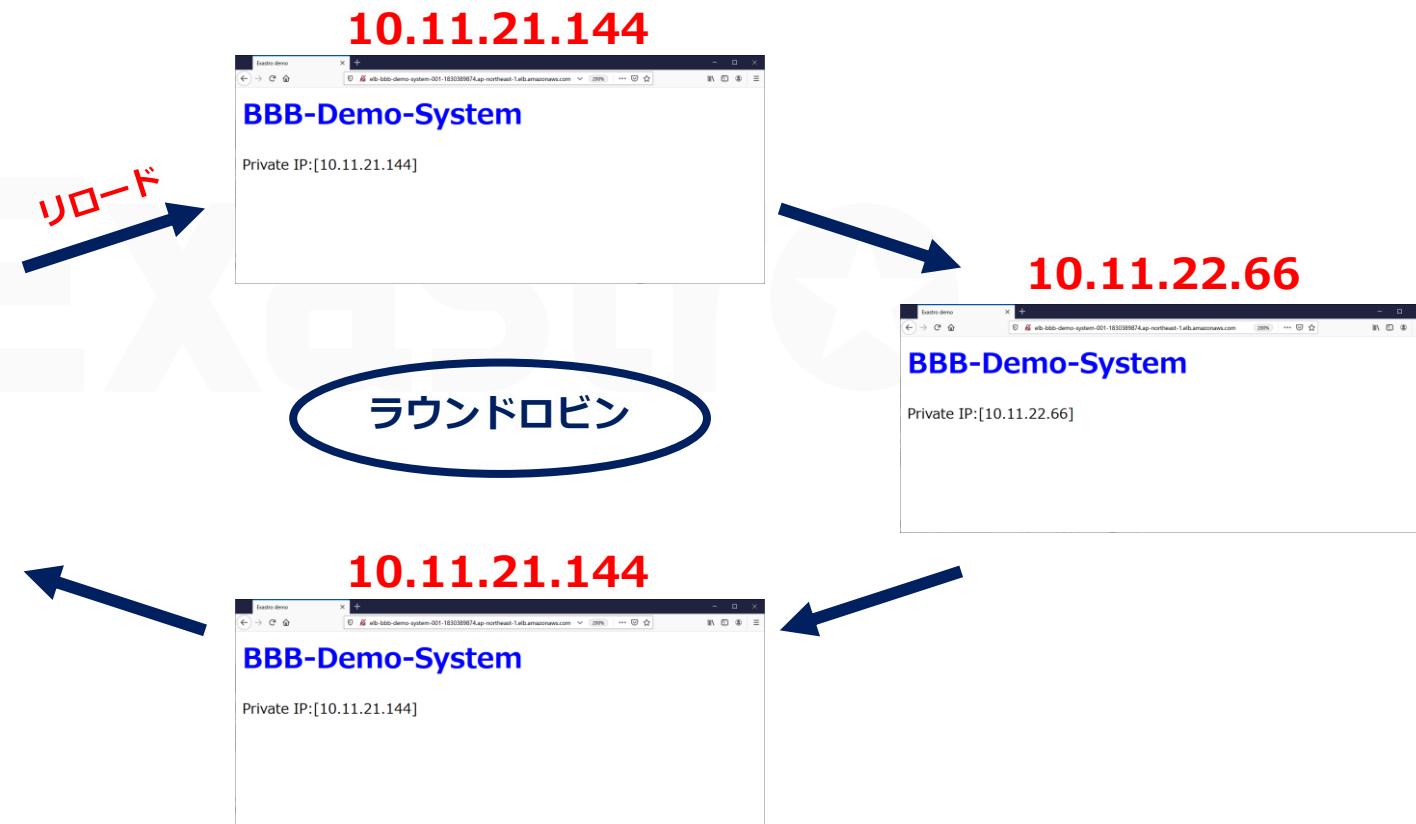
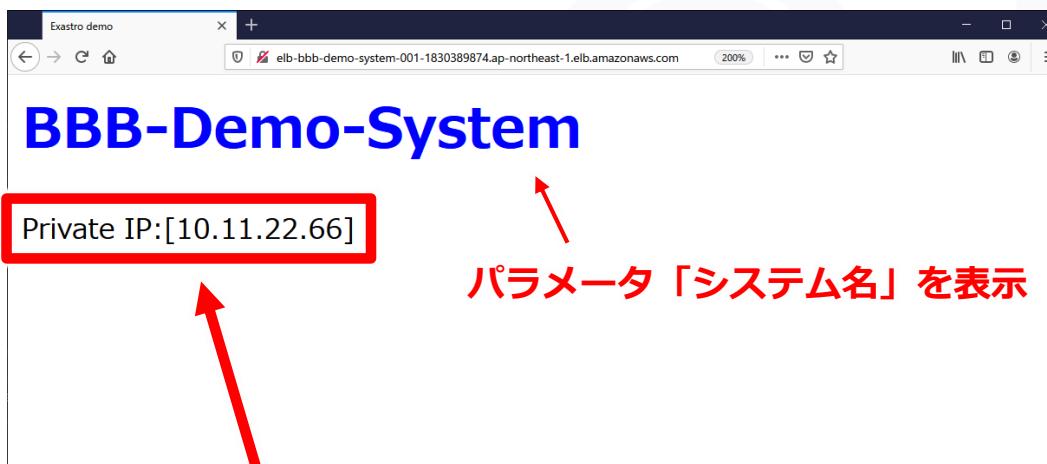
緊急停止

オペレーションID 1
オペレーション名 環境A
ステータス 実行中
実行ユーザ インフラ管理者
予約日時
緊急停止命令 未発令

出来上がったシステムを確認

ブラウザでロードバランサにアクセスすると、下記の通り簡単なWeb画面が閲覧できます。
(リロードするたびに、複数のWebサーバをラウンドロビンすることが確認できます。)

ロードバランサへのアクセス結果画面





参考 - 通常はユーザが見ることはないメニュー -

参考 - 通常はユーザが見ることはないメニュー - (1/4)

The screenshot shows the Exastro IT Automation Document Management interface. The top navigation bar includes the Exastro logo, a user greeting (ようこそ[インフラ管理者]さん), login information (ログインID [infra-admin]), and links for password change and logout. The left sidebar contains a menu with sections like Main Menu, Basic Design, Detailed Design, Structure Documents, and Evaluation Documents. The main content area displays a table of uploaded documents with columns for Action, Status, ID, Version, File Name, Reference, Last Update Date, and Last Updater. A red box highlights the document list. Below the table, a message states: "CSテンプレートの設計書類もExastro IT Automationで管理可能です。ドキュメント管理に必要なドキュメントをアップロードします。" (CS template design documents can also be managed by Exastro IT Automation. Upload the documents required for document management.)

更新	廃止	No	バージョン	ファイルアップロード	備考	最終更新日時	最終更新者
更新	廃止	1	1.0	CloudFormation実装方式設計書.pptx		2020/06/05 14:09:37	データポータビリティプロシージャ
更新	廃止	2	1.0	SIテンプレート機能設計書(スケールアウト型サーバ).xlsx		2020/06/05 14:09:52	データポータビリティプロシージャ
更新	廃止	3	1.0	SIテンプレート機能設計書(共通設計).xlsx		2020/06/05 14:10:04	データポータビリティプロシージャ
更新	廃止	4	1.0	SIテンプレート機能設計書(付帯機能_セキュリティ).xlsx		2020/06/05 14:10:23	データポータビリティプロシージャ
更新	廃止	5	1.0	SIテンプレート機能設計書(付帯機能_運用管理).xlsx		2020/06/05 14:11:01	データポータビリティプロシージャ

フィルタ結果件数: 5

CSテンプレートの設計書類もExastro IT Automationで管理可能です。
ドキュメント管理に必要なドキュメントをアップロードします。

Excel出力

登録

全件ダウンロードとファイルアップロード編集

変更履歴

管理者に連絡

参考 - 通常はユーザが見ることはないメニュー - (2/4)

Exastro IT Automation 構築/更新シナリオ

ようこそ[インフラ管理者]さん
ログインID [infra-admin]
パスワード変更 ログアウト

Menu

メインメニュー

オートスケールWebサーバ

GuardDuty

説明

表示フィルタ

一覧/更新

△閉じる

更新	廃止	No.	オペレーション				代入順序	パラメータ			最終更新日時	最終更新者
			オペレーション名	基準日時	実施予定日時	最終実行日時		スタック名	テンプレートファイル	対象		
更新	廃止	1	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	1 IAM-Role	"{{ TPF_IAM_ROLE }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	
更新	廃止	2	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	2 KMS	"{{ TPF_Security_KMS }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	
更新	廃止	3	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	3 S3	"{{ TPF_Storage_S3 }}"	●	2020/06/08 18:28:04	データポータビリティプロシージャ	
更新	廃止	4	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	4 SNS	"{{ TPF_AppIntegration_SNS }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	
更新	廃止	5	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	5 CloudTrail	"{{ TPF_Management_CloudTrail }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	
更新	廃止	6	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	6 Network	"{{ TPF_Network }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	
更新	廃止	7	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	7 SecurityGroup	"{{ TPF_SecurityGroup }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	
更新	廃止	8	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	8 Bastion	"{{ TPF_Bastion }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	
更新	廃止	9	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	9 AutoScaleServer	"{{ TPF_AutoScale }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	
更新	廃止	10	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	10 VPCflowlog	"{{ TPF_Management_VPCflowlog }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	
更新	廃止	11	環境A	2020/06/18 16:16	2020/05/28 11:00	2020/06/18 16:16	11 CloudWatchAlarm	"{{ TPF_Management_CloudWatchAlarm }}"	●	2020/06/05 09:58:09	データポータビリティプロシージャ	

フィルタ結果件数: 11

Excel出力

シナリオ毎に使うスタック(=Template File)を紐づけます

- ※ 例えば「オートスケールWebサーバ」では11本のスタックを紐づけています。
- ※ 構築対象に「●」を付けているものが構築されます。
- 「●」をメンテナンスすることでスタックを選択できます。
- ※ スタック単位にリトライ回数の設定も可能です。

参考 - 通常はユーザが見ることはないメニュー - (3/4)

Exastro IT Automation Ansible共通 ようこそ[インフラ管理者]さん ログインID [infra-admin] パスワード変更 ログアウト

☰ Menu

メインメニュー グローバル変数管理 ファイル管理 テンプレート管理

説明 表示フィルタ 一覧/更新

更新 廃止 素材ID: 1 TPF_AutoScale テンプレート埋込変数名: [SiTemplate_AutoScale.yaml](#)

更新 廃止 2 TPF_Security_KMS テンプレート埋込変数名: [SiTemplate_Security_KMS.yaml](#)

更新 廃止 3 TPF_Storage_S3 テンプレート埋込変数名: [SiTemplate_Storage_S3.j2](#)

管理者に連絡

Ansible変数化した「Template File」はここで管理

87
88
89 # 2.ElasticLoadBalancing
90 ##
91 # 2-1.ElasticLoadBalancing
92 # ElasticLoadBalancing1:
93 # Type: 'AWS::ElasticLoadBalancingV2::LoadBalancer'
94 Properties:
95 Fn::ImportValue: !Sub \${SystemName}-S3BucketNameAlbLog
96 ipAddressType: ipv4
97 LoadBalancerAttributes:
98 - Key: access_logs.s3.enabled
99 Value: true
100 - Key: access_logs.s3.bucket
101 Value:
102 Fn::ImportValue: !Sub \${SystemName}-S3BucketNameAlbLog
103 - Key: access_logs.s3.prefix
104 Value: !Ref AlbLogPrefix1
105 - Key: deletion_protection.enabled
106 Value: false
107 - Key: idle_timeout.timeout_seconds
108 Value: 60
109 - Key: routing.http2.enabled
110 Value: true
111 Name: !Sub "elb-\${SystemName}-001"
112 Scheme: internet-facing
113 SecurityGroups:
114 - Fn::ImportValue: !Sub \${SystemName}-ELBSecurityGroup
115 Subnets:
116 - Fn::ImportValue: !Sub \${SystemName}-SubnetIdPublic1
117 - Fn::ImportValue: !Sub \${SystemName}-SubnetIdPublic2
118 #- Fn::ImportValue: !Sub sub-\${SystemName}-public-001
119 #- Fn::ImportValue: !Sub sub-\${SystemName}-public-002
120 Tags:
121 - Key: Name
122 Value: !Sub "elb-\${SystemName}-001"
123 - Key: Environment
124 Value: !Sub "\${Environment}"
125 - Key: Application
126 Value: !Sub "\${Application}"
127 - Key: Owner
128 Value: !Sub "\${Owner}"
129 - Key: Department
130 Value: !Sub "\${Department}"
131 - Key: CostCenter
132 Value: !Sub "\${CostCenter}"
133
134
135
136 # 2-2.Listener
137

△開く △開く △閉じる

参考 - 通常はユーザが見ることはないメニュー - (4/4)

Exastro IT Automation Ansible-Legacy ようこそ[インフラ管理者]さん
ログインID [infra-admin] パスワード変更 ログアウト

説明

表示フィルタ

一覧/更新

更新 廃止 素材ID
1 テンプレートファイルの作成 CFn_templates.yaml
2 Cloudformation実行(構築／更新) CFn_module.yaml
3 Cloudformation実行(削除) CFn_module_delete.yaml
4 起動時コマンドの設定 set_user_data.yaml
5 ELBホスト名取得 elb_dnsname.yaml
6 IAM-Role作成 CFn_Iam_Role.yaml
7 IAMユーザー作成 CFn_Iam_User.yaml
8 構築用ファイルを作成 action_file_create.yaml
9 削除用ファイルを作成 action_file_delete.yaml
10 密密鍵を作業用ディレクトリに展開 set_secret_key.yaml
11 一覧を初期化 device_list_sync.yaml
12 ファイルを送信 send_imagefile.yaml
13 バケットの削除 delete_S3_bucket.yaml
14 チームメッセージを送信 send_teams_message.yaml

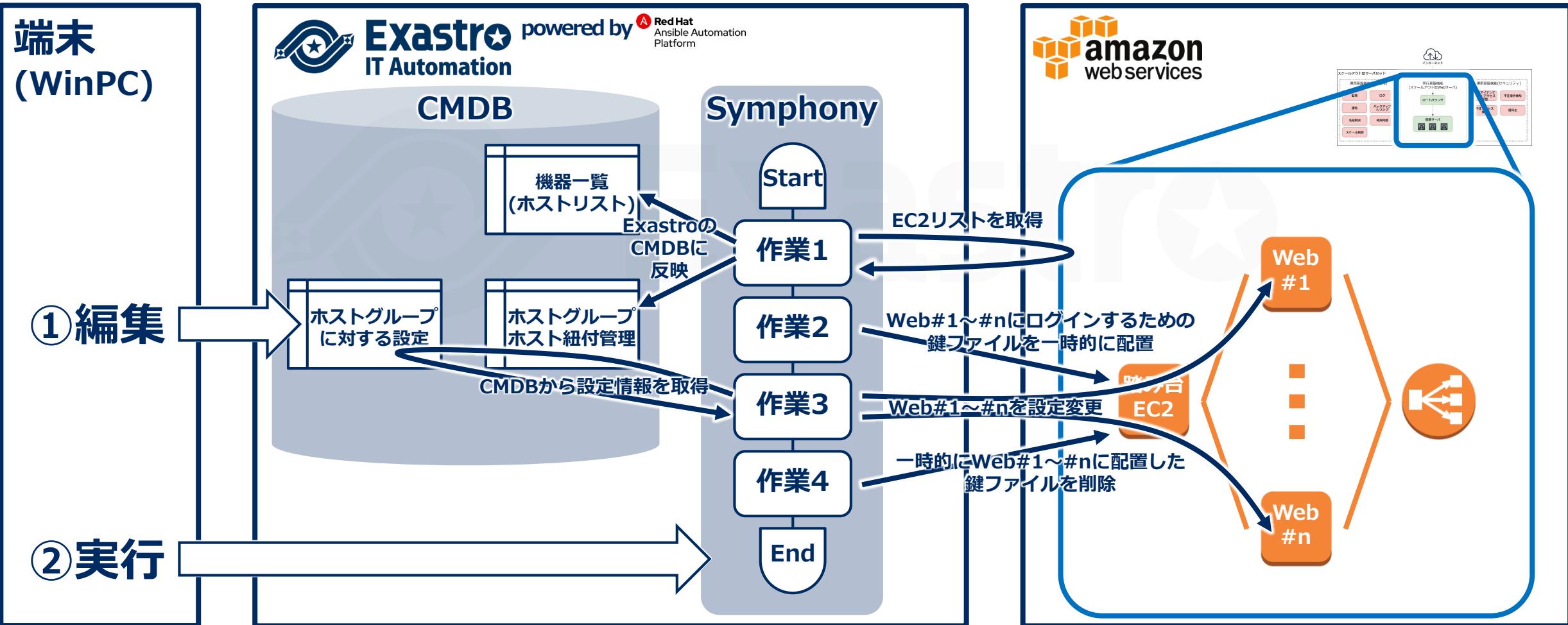
使用している
「Ansible Playbook」
はここで管理
※単純なPlaybookで
実現できています

1
2 name: Create_Templatefile
3 template: src="{{ item.1 }}" dest="{{ __symphony_workflowdir__ }}/{{ item.0 }}.yaml"
4 when: item.2 == '●'
5 with_together:
6 - "[[VAR_STACKNAME]]"
7 - "[[VAR_TPFFILE]]"
8 - "[[VAR_EXE_FLAG]]"
EOF
1 state: present
2 template: "[[__symphony_workflowdir__ }}/{{ item.0 }}.yaml"
3 template_parameters:
4 UpdateDate: "[[lookup('pipe','date +%Y%m%d%H%M')]]"
5 role_arn: "arn:aws:iam::{{ VAR_accountId }}:role/CloudFormationRole"
6 validate_certs: no
7 when: item.1 == '●'
8 with_together:
9 - "[[VAR_stackName]]"
10 - "[[VAR_EXE_FLAG]]"
11 environment:
12 AWS_ACCESS_KEY_ID: "{{ VAR_ACCESS_KEY }}"
13 AWS_SECRET_ACCESS_KEY: "{{ VAR_SECRET_ACCESS_KEY }}"
14 AWS_DEFAULT_REGION: "{{ VAR_REGION }}"
15 EOF



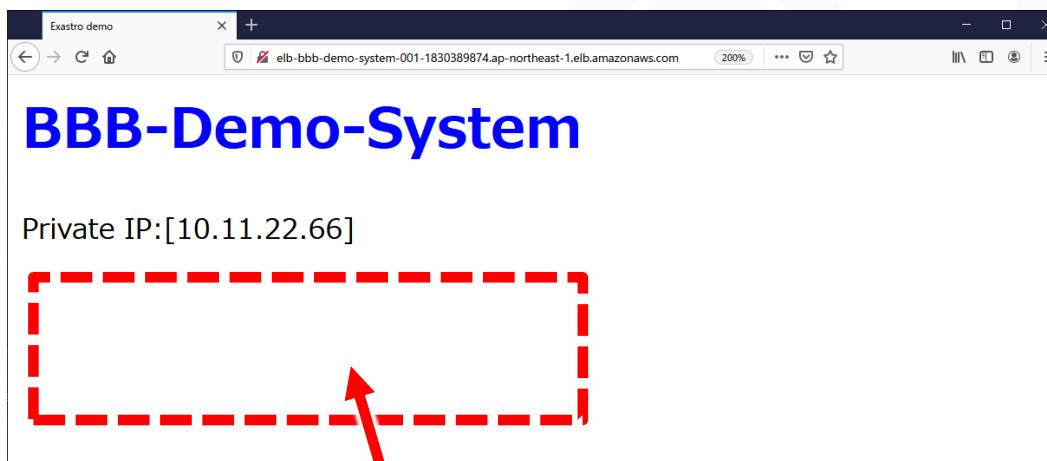
付録A - 出来上がったシステムのEC2にAnsibleを使って設定変更する -

稼働中のEC2(オートスケール)に緊急でパッチ適用するケースなどにおいて Exastro ITA + Red Hat Ansible Automation Platformを活用頂けます

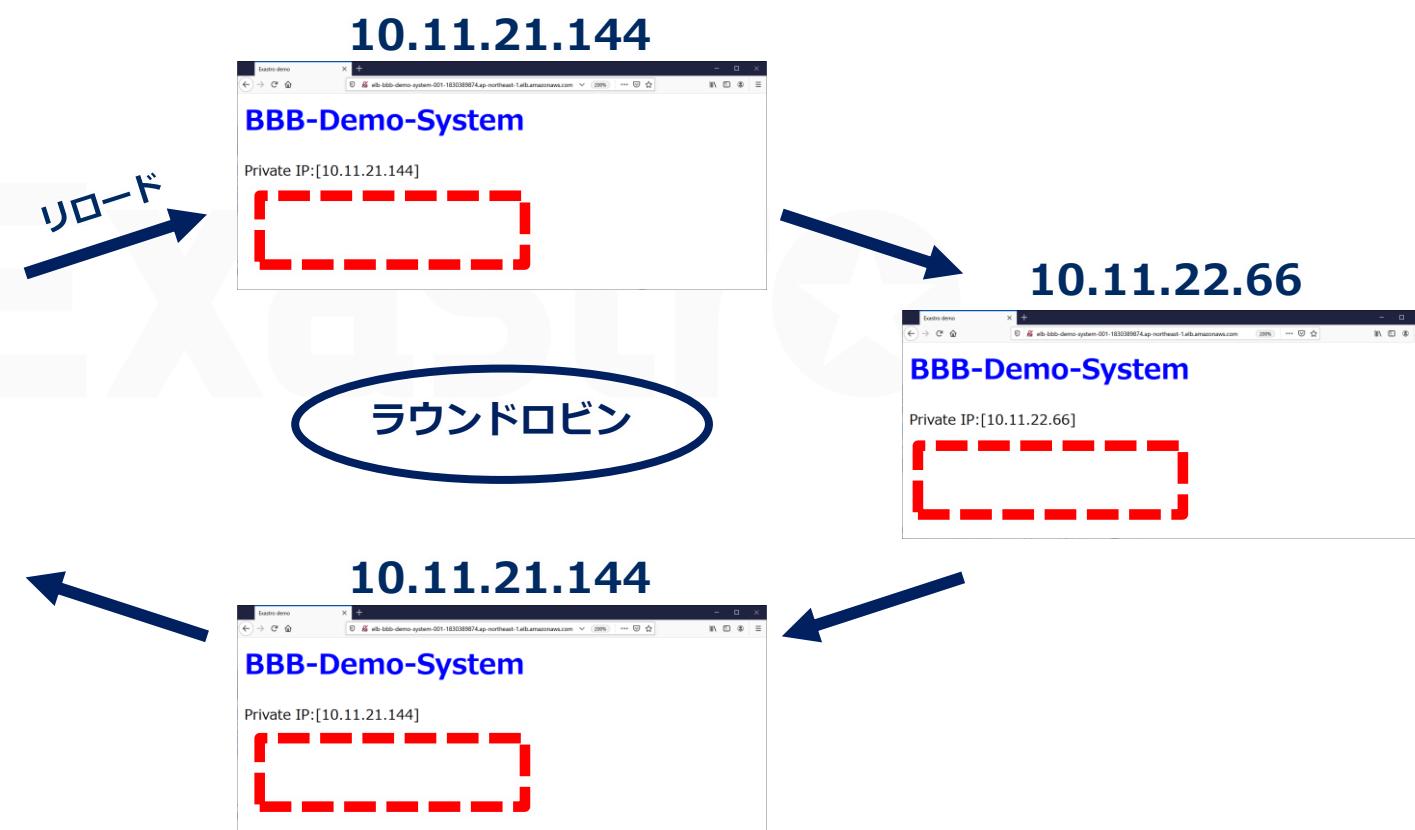


オートスケール対象の複数台のEC2に対してWeb画面に画像(※)を挿入します。
※ EC2に対するパッチ適用に見立ててご理解ください。

ロードバランサへのアクセス結果画面



画像が無い状態
(パッチ適用前と考えてください)



付録A - 出来上がったシステムのEC2にAnsibleを使って設定変更する - (3/6)

Exastro IT Automation Ansible共通 ようこそ[インフラ管理者]さん
ログインID [infra-admin] パスワード変更 ログアウト

☰ Menu ▽開く
メインメニュー
グローバル変数管理
ファイル管理

テンプレート管理 △閉じる

説明
表示フィルタ
一覧/更新

更新	廃止	素材ID	ファイル埋込変数名	ファイル素材	備考	最終更新日時	最終更新者
更新	廃止	1	CPF_STARTUP	ec2_userdata	WEBサーバの起動時設定	2020/05/28 09:17:47	データポータビリティプロシージャ
更新	廃止	2	CPF_SECRET_KEY_BASTION	si-template-omcs.pem	踏み台サーバの秘密鍵	2020/06/10 14:14:30	システム管理者
更新	廃止	3	CPF_SECRET_KEY_WEB	si-template-omcs.pem	WEBサーバの秘密鍵	2020/06/10 14:14:43	システム管理者
更新	廃止	4	CPF_IMAGEFILE	Exastro-ITA-logo1-rgb.png	Webサーバ表示用画像	2020/06/18 09:16:11	システム管理者
更新	廃止	5	CPF_AWS_SDK	aws_sdk.zip	AWS SDK for PHP	2020/05/28 09:19:11	データポータビリティプロシージャ
更新	廃止	6	CPF_DEVICE_LIST_SYNC	device_list_sync	機器一覧同期モジュール	2020/06/08 12:04:19	データポータビリティプロシージャ
更新	廃止	7	CPF_IMAGEFILE	super-tower.jpg	デモ表示用画像		

フィルタ結果件数: 6

Excel出力

登録

全件ダウンロードとファイルアップロード編集

変更履歴

※*は必須項目です。

挿入したい画像をアップロード
(パッチデータをアップロードするイメージ)

素材ID ファイル埋込変数名* ファイル素材* 備考 最終更新日時 最終更新者

7 CPF_IMAGEFILE 現在アップロードされているファイル:
[super-tower.jpg](#)
ファイルを選択 選択されていません
事前アップロード
アップロード状況:

戻る 更新

付録A - 出来上がったシステムのEC2にAnsibleを使って設定変更する - (4/6)

The screenshot shows the Exastro basic console interface. On the left, there's a sidebar with various menu items like 'Menu', 'メインメニュー', '機器一覧', '投入オペレーション一覧', 'Symphonyクラスター一覧', 'Symphonyクラス編集', 'Symphony作業実行', 'Symphony作業確認', 'Symphony作業一覧', and '定期作業実行'. The main area displays a table titled 'Symphony' with columns: 選択, SymphonyクラスID, Symphony名稱, 説明, 備考, 最終更新日時, and 最終更新者. There are six entries:

- 1 オートスケールWebサーバ (構築/更新) - Description: オートスケールWebサーバを構築または更新を実行します。 AWS CloudFormationでスタックが未作成の場合、構築処理が実行されます。スタックが作成済みかつパラメータに変更がある場合、更新処理が実行されます。
- 2 オートスケールWebサーバ (EC2へPlaybook実行) - Description: オートスケールWebサーバのEC2インスタンスに対してPlaybookを実行します。 実行するPlaybookを変更する際はMovementを新規作成し、「イメージファイルをWebサーバへ設定」と変更してください。 ◆ロール名「インフラユーザー」が付与されたITAユーザーが実行。
- 3 GuardDuty (構築/更新) - Description: GuardDuty構築または更新を実行します。 AWS CloudFormationでスタックが未作成の場合、構築処理が実行されます。スタックが作成済みかつパラメータに変更がある場合、更新処理が実行されます。
- 4 GuardDuty (削除) - Description: GuardDuty削除を実行します。 ◆ロール名「インフラ管理者」が付与されたITAユーザーが実行。
- 5 AWS管理者 (構築/更新) - Description: AWS管理者構築または更新を実行します。 ◆ロール名「インフラ管理者」が付与されたITAユーザーが実行。
- 6 AWS管理者 (削除) - Description: AWS管理者削除を実行します。 ◆ロール名「インフラ管理者」が付与されたITAユーザーが実行。

A red box highlights the second entry, '2 オートスケールWebサーバ (EC2へPlaybook実行)'. A red arrow points from this entry to the text 'Symphony(=ジョブフロー)を選択' and '「オートスケールWebサーバ (EC2へPlaybook実行)」を選択'.

At the bottom, another red box highlights the first entry in the 'Operation' table, '1 環境A'. A red arrow points from this entry to the text 'Operation(=パラメータのセット)を選択' and '例えば「環境A」を選択'.

選択	SymphonyクラスID	Symphony名稱	説明	備考	最終更新日時	最終更新者
<input type="radio"/>	1	オートスケールWebサーバ (構築/更新)	オートスケールWebサーバを構築または更新を実行します。 AWS CloudFormationでスタックが未作成の場合、構築処理が実行されます。スタックが作成済みかつパラメータに変更がある場合、更新処理が実行されます。		2020/06/09 16:41:00	データポータビリティプロシージャ
<input type="radio"/>	2	オートスケールWebサーバ (EC2へPlaybook実行)	オートスケールWebサーバのEC2インスタンスに対してPlaybookを実行します。 実行するPlaybookを変更する際はMovementを新規作成し、「イメージファイルをWebサーバへ設定」と変更してください。 ◆ロール名「インフラユーザー」が付与されたITAユーザーが実行。		2020/06/09 16:41:28	データポータビリティプロシージャ
<input type="radio"/>	3	GuardDuty (構築/更新)	GuardDuty構築または更新を実行します。 AWS CloudFormationでスタックが未作成の場合、構築処理が実行されます。スタックが作成済みかつパラメータに変更がある場合、更新処理が実行されます。		2020/06/09 16:41:45	データポータビリティプロシージャ
<input type="radio"/>	4	GuardDuty (削除)	GuardDuty削除を実行します。 ◆ロール名「インフラ管理者」が付与されたITAユーザーが実行。		2020/06/09 16:41:58	データポータビリティプロシージャ
<input type="radio"/>	5	AWS管理者 (構築/更新)	AWS管理者構築または更新を実行します。 ◆ロール名「インフラ管理者」が付与されたITAユーザーが実行。		2020/06/09 16:42:11	データポータビリティプロシージャ
<input type="radio"/>	6	AWS管理者 (削除)	AWS管理者削除を実行します。 ◆ロール名「インフラ管理者」が付与されたITAユーザーが実行。		2020/06/09 16:42:13	データポータビリティプロシージャ

オペレーション[フィルタ]

オペレーション[一覧]

フィルタ結果件数: 2

正常終了すればEC2への設定変更が完了です

Symphony作業確認

△閉じる

Menu

- メインメニュー
- 機器一覧
- 投入オペレーション一覧
- Symphonyクラス一覧
- Symphonyクラス編集
- Symphony作業実行
- Symphony作業確認**
- Symphony作業一覧
- 定期作業実行

SymphonyインスタンスID 324

説明 オートスケールWebサーバのEC2インスタンスに対してPlaybookを実行します。
実行するPlaybookを変更する際はMovementを新規作成し、「イメージファイルをWebサーバへ設定」と変更してください。

オペレーションID 1

オペレーション名 環境A

ステータス 正常終了
実行ユーザ インフラ管理者
予約日時
緊急停止命令 未発令

▼ start

[1] 機器一覧を同期(インフラユーザー)
DONE 開始日時 2020/06/18 16:46:29
終了日時 2020/06/18 16:46:41

[2] 密钥鍵を作業用ディレクトリへ展開
DONE 開始日時 2020/06/18 16:46:44
終了日時 2020/06/18 16:46:54

[3] イメージファイルをWebサーバへ設定
DONE 開始日時 2020/06/18 16:46:57
終了日時 2020/06/18 16:47:09

[4] 作業用ディレクトリから密钥鍵を削除
DONE 開始日時 2020/06/18 16:47:12
終了日時 2020/06/18 16:47:18

緊急停止

管理者に連絡

Exastro

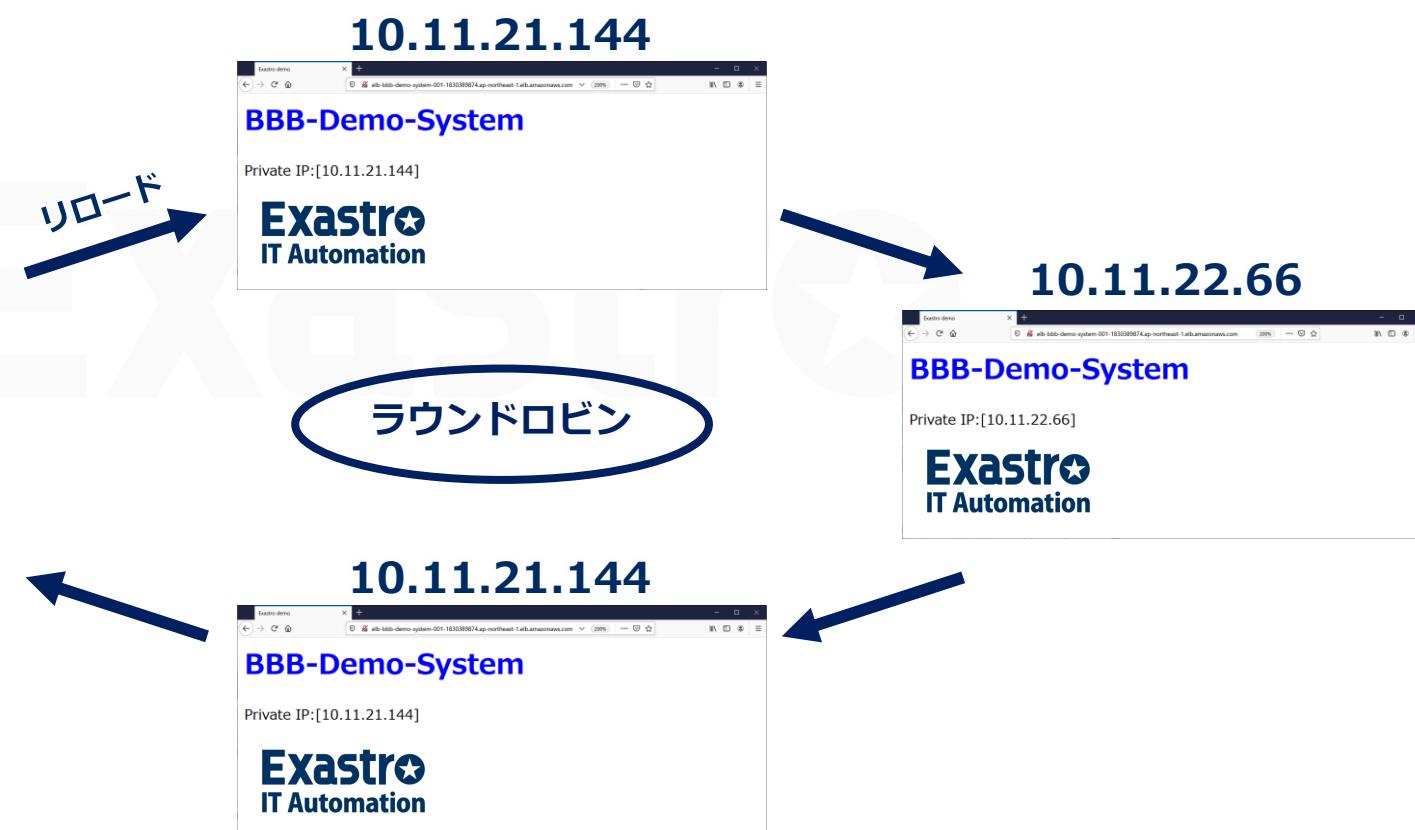
32

オートスケール対象の複数台のEC2に画像が挿入されます。
(パッチが適用されたとお考え下さい)

ロードバランサへのアクセス結果画面



全てのEC2に画像が挿入されます
(パッチが適用されたと考えてください)





Exastro 