



# IT Automation Terraform Driver 【実習編】

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

Exastro IT Automation ver 1.8  
Exastro developer

# 目次

## 1. はじめに

- 1.1 はじめに
- 1.2 作業環境

## 2. 実習 Terraform Driver

- 2.1 シナリオ
- 2.2 事前準備

## 3. 仕込み編

- 3.1 インターフェース情報の登録
- 3.2 Organizationの登録と連携
- 3.3 Workspaceの登録と連携
- 3.4 作業パターン(Movement)の登録
- 3.5 Module素材の登録
- 3.6 Policy素材の登録
- 3.7 Policy Setの登録
- 3.8 Policy SetとPolicyの紐付け
- 3.9 Policy SetとWorkspaceの紐付け
- 3.10 MovementにModule素材を指定

## 4. 実行編

- 4.1 オペレーションの登録
- 4.2 変数値の設定
- 4.3 Planを確認
- 4.4 PolicyCheckログを確認
- 4.5 VMのサイズを変更して再度確認
- 4.6 再度PolicyCheckログを確認
- 4.7 作業実行
- 4.8 実行状態確認
- 4.9 数値を変更して再度実行

# 1. はじめに

# 1.1 はじめに

## メインメニュー

- 本書では、メニューグループの「**Terraform**」について、実践形式で学習いただけます。

The screenshot displays the Exastro IT Automation dashboard. The top navigation bar includes the Exastro logo, a 'Menu' button, and a 'DASHBOARD' tab. The user is logged in as 'administrator' with options to change the password or log out. The main content area is divided into a left sidebar and a central grid. The sidebar shows a 'メインメニュー' (Main Menu) section. The central grid features a 'メニューグループ' (Menu Group) section with various icons. The 'Terraform' icon, located at the bottom of the menu group, is highlighted with a red rectangle. To the right of the menu group are three large circular gauges labeled 'Movement', 'Status', and 'Result', each showing a 'Total' value of 0. Below these gauges are three tables: 'Movement', 'Status', and 'Result', each with columns for 'CON', 'SYN', and 'SUM'. The 'Movement' table lists 'Ansible Legacy', 'Ansible Pioneer', 'Ansible Legacy Role', and 'Terraform'. The 'Status' table lists '実行中' (Running), '未実行(予約)' (Not executed (Reserved)), and '未実行' (Not executed). The 'Result' table lists '正常終了' (Normal completion), '異常終了' (Abnormal completion), '想定外エラー' (Unexpected error), '緊急停止' (Emergency stop), and '予約取消' (Reservation cancellation). At the bottom of the dashboard is a '作業履歴' (Job History) section with a table of job records.

Exastro IT Automation

ようこそ[システム管理者]さん  
ログインID [administrator]  
パスワード変更 ログアウト

Menu DASHBOARD

メインメニュー

メニューグループ

管理コンソール 基本コンソール エクスポートノイ... Symphony Conductor メニュー作成

比較 ホストグループ管理 Ansible共通 Ansible-Legacy Ansible-Pioneer Ansible-LegacyR...

Terraform

Movement

作業状況

作業結果

作業履歴

## 1.2 作業環境

### 作業環境

- 本書で使用する作業環境は以下の通りです。
- ITAサーバとは他に、AWS、Azure、Terraform Enterpriseアカウントをご用意ください。

#### ITAホストサーバ

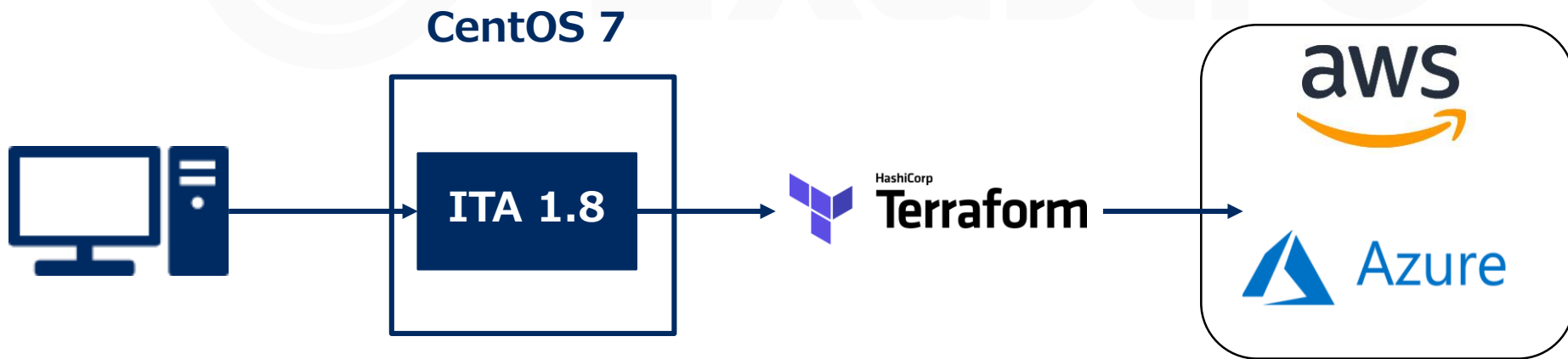
- CentOS7(※)
- ITA ver 1.8

#### Terraform

- Terraform Enterprise

#### ターゲット

- AWS
- Microsoft Azure



※今回はホストサーバーとしてCentOS7を利用致しますが、ITAはRHEL7系およびRHEL8系のOSで導入いただけます。

## 2. 実習 Terraform Driver

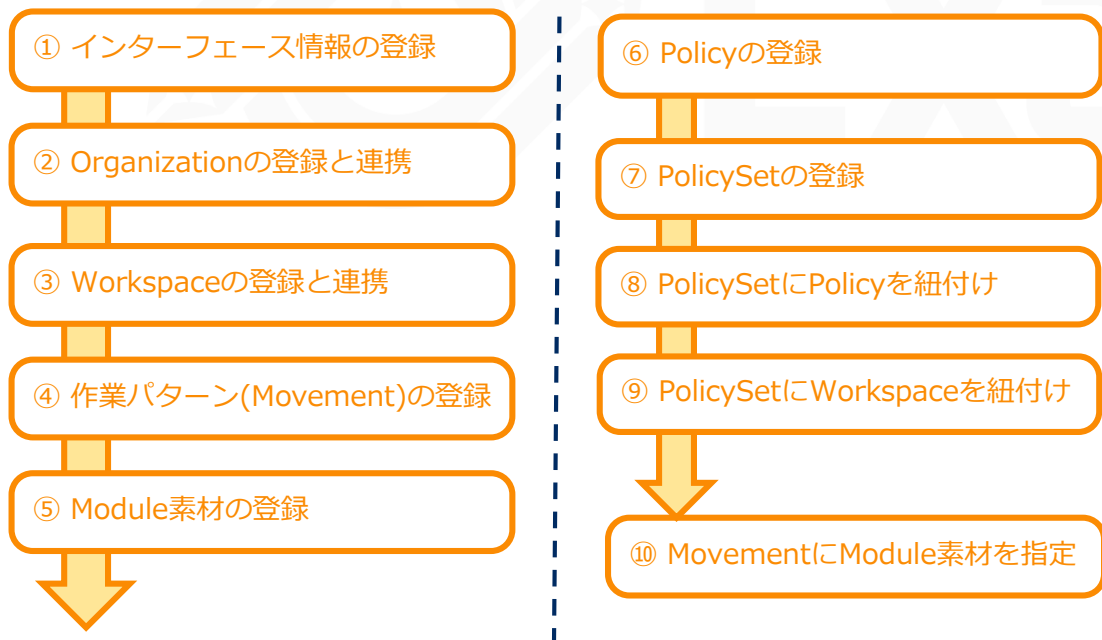
## 2.1 シナリオ

### シナリオについて

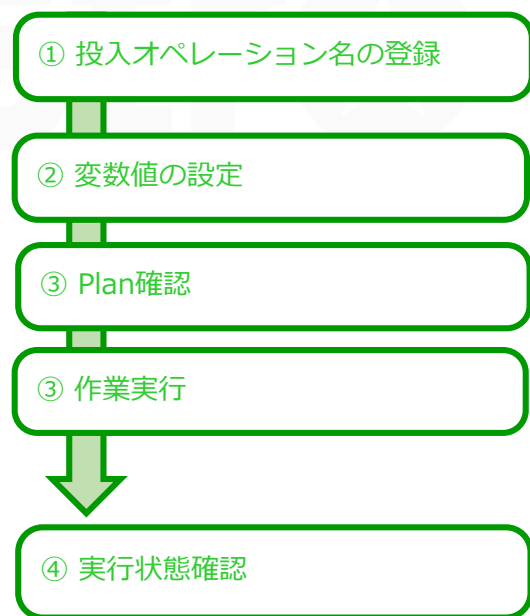
本シナリオは、ITAのTerraform Driverを利用して、パブリッククラウド上(AWS、Azure)にVMを作成するPlanを確認します。その後、**定義されたポリシー**に沿った設定を行い各クラウドにVMを作成します。

【**仕込み編**】までを一度登録・連携すると、以降の操作は【**実行編**】を繰り返し行うことで、対象の再設定・再登録を行うことができます。(自動化)

#### 【仕込み編】



#### 【実行編】



## 2.2 事前準備(1/7)

### Moduleの作成

- 本シナリオで使用する4つのModuleを作成します。

【注意】 文字コードは"UTF-8"、改行コードは"LF"、拡張子は"tf"で作成してください。

```
variable "access_key" {}
variable "secret_key" {}
variable "region" {}
variable "ami" {}
variable "key_name" {}
variable "security_group" {}
variable "tags_name" {}
variable "hello_tf_instance_count" {
  default = 2
}
variable "hello_tf_instance_type" {
  default = "t2.micro"
}
```

ファイル名 :

**aws\_create\_instance\_variables.tf**

AWSインスタンス作成用の変数定義ファイルです。  
変数には具体値変数が代入されます



## 2.2 事前準備(2/7)

### Moduleの作成

```
provider "aws" {  
  access_key = var.access_key  
  secret_key = var.secret_key  
  region = var.region  
}  
  
resource "aws_instance" "hello-tf-instance" {  
  ami          = var.ami  
  key_name     = var.key_name  
  security_groups = [var.security_group]  
  tags = {  
    Name = "${var.tags_name}-${count.index+1}"  
  }  
  count = var.hello_tf_instance_count  
  instance_type = var.hello_tf_instance_type  
}
```

**ファイル名 :**  
**aws\_create\_instance.tf**

AWSインスタンス作成用の  
リソース定義ファイルです。  
セキュリティグループ、キーペアは  
事前にAWSで作成・用意します。

## 2.2 事前準備(3/7)

### Moduleの作成

```
variable "subscription_id" {}
variable "tenant_id" {}
variable "client_id" {}
variable "client_secret" {}
variable "resource_group_name" {}
variable "security_group" {}
variable "location" {}
variable "Vnet_name" {}
variable "Vnet_address_space" {}
variable "subnet_name" {}
variable "address_prefixes" {}
variable "public_ip_name" {}
variable "allocation_method" {}
variable "domain_name_label" {}
variable "network_interface_name" {}
variable "NIC_name" {}
variable "VM_name" {}
variable "VM_size" {}
variable "publisher" {}
```

```
variable "offer" {}
variable "sku" {}
variable "source_image_version" {}
variable "admin_username" {}
variable "ssh_public_key" {}
variable "os_disk_name" {}
variable "caching" {}
variable "storage_account_type" {}
variable "VM_count" {}
```

**ファイル名 :**  
**azure\_create\_instance\_variables.tf**

Azureインスタンス作成用の変数定義ファイルです。  
変数には具体値変数が代入されます

## 2.2 事前準備(4/7)

### Moduleの作成

ファイル名 :


**azure\_create\_instance.tf(1/3)**

Azureインスタンス作成用のリソース  
定義ファイルです。


リソースグループと、そのネットワークセキュ  
リティグループ、仮想ネットワークの作成。

また、作成するVM台数分の仮想マシン本体、  
ディスク、ネットワークインターフェースの  
作成を行います。

```
provider "azurerm" {  
  features {}  
  subscription_id = var.subscription_id  
  client_id       = var.client_id  
  client_secret   = var.client_secret  
  tenant_id      = var.tenant_id  
}  
  
resource "azurerm_resource_group" "hoge" {  
  name = var.resource_group_name  
  location = var.location  
}
```




```
resource "azurerm_network_security_group" "hoge" {  
  name = var.security_group  
  location = azurerm_resource_group.hoge.location  
  resource_group_name = azurerm_resource_group.hoge.name  
  
  security_rule {  
    name           = "SSH"  
    priority       = 1001  
    direction      = "Inbound"  
    access         = "Allow"  
    protocol       = "Tcp"  
    source_port_range = "*"   
    destination_port_range = "22"  
    source_address_prefix = "*"   
    destination_address_prefix = "*"   
  }  
  
  security_rule {  
    name           = "HTTP"  
    priority       = 1002  
    direction      = "Inbound"  
    access         = "Allow"  
    protocol       = "Tcp"  
    source_port_range = "*"   
    destination_port_range = "80"  
    source_address_prefix = "*"   
    destination_address_prefix = "*"   
  }  
}
```



## 2.2 事前準備(5/7)

### Moduleの作成

ファイル名 :  
**azure\_create\_instance.tf(2/3)**




```
resource "azurerm_virtual_network" "hoge hoge" {
  name = var.Vnet_name
  address_space = [var.Vnet_address_space]
  location = azurerm_resource_group.hoge hoge.location
  resource_group_name = azurerm_resource_group.hoge hoge.name
}

resource "azurerm_subnet" "hoge hoge" {
  name = var.subnet_name
  resource_group_name = azurerm_resource_group.hoge hoge.name
  virtual_network_name = azurerm_virtual_network.hoge hoge.name
  address_prefixes = [var.address_prefixes]
}

resource "azurerm_public_ip" "hoge hoge" {
  count = var.VM_count
  name = "${var.public_ip_name}-${count.index}"
  location = azurerm_resource_group.hoge hoge.location
  resource_group_name = azurerm_resource_group.hoge hoge.name
  allocation_method = var.allocation_method
  domain_name_label = "${var.domain_name_label}-${count.index}"
}

resource "azurerm_network_interface" "hoge hoge" {
  count = var.VM_count
  name = "${var.network_interface_name}-${count.index}"
  location = azurerm_resource_group.hoge hoge.location
  resource_group_name = azurerm_resource_group.hoge hoge.name

  ip_configuration {
    name = var.NIC_name
    subnet_id = azurerm_subnet.hoge hoge.id
    private_ip_address_allocation = var.allocation_method
    public_ip_address_id = azurerm_public_ip.hoge hoge[count.index].id
  }
}
```



### Moduleの作成

ファイル名 :  
**azure\_create\_instance.tf(3/3)**



```
resource "azurerm_network_interface_security_group_association" "hoge hoge" {
  count = var.VM_count
  network_interface_id = azurerm_network_interface.hoge hoge[count.index].id
  network_security_group_id = azurerm_network_security_group.hoge hoge.id
}

resource "azurerm_linux_virtual_machine" "hoge hoge" {
  count = var.VM_count
  name = "${var.VM_name}-${count.index}"
  resource_group_name = azurerm_resource_group.hoge hoge.name
  location = azurerm_resource_group.hoge hoge.location
  size = var.VM_size
  admin_username = var.admin_username
  network_interface_ids = [azurerm_network_interface.hoge hoge[count.index].id]

  admin_ssh_key {
    username = var.admin_username
    public_key = var.ssh_public_key
  }

  os_disk {
    name = "${var.os_disk_name}-${count.index}"
    caching = var.caching
    storage_account_type = var.storage_account_type
  }

  source_image_reference {
    publisher = var.publisher
    offer = var.offer
    sku = var.sku
    version = var.source_image_version
  }
}
```

## 2.2 事前準備(7/7)

### Policyの作成

#### ファイル名 : **limit-proposed-monthly-cost.sentinel**

月額のコストを制限するポリシーです。

月額コストが\$50を上回る場合は  
Applyを行いません。

また、その月額コストの  
総見積りが出力されます。

AWS,Azureどちらのクラウドにおいても  
適用可能です。

```
import "tfrun"
import "decimal"

limit = decimal.new(50)

cost_limit_by_workspace = func() {
  if tfrun.cost_estimate else null is null {
    print("no cost estimates available")
    return false
  }

  workspace_name = tfrun.workspace.name

  proposed_cost = decimal.new(tfrun.cost_estimate.proposed_monthly_cost)

  if proposed_cost.less_than(limit) {
    print("Proposed monthly cost", proposed_cost.string,
      "of workspace", workspace_name,
      "is under the limit: $", limit)
    return true
  }

  if proposed_cost.greater_than(limit) {
    print("Proposed monthly cost", proposed_cost.string,
      "of workspace", workspace_name,
      "is over the limit: $", limit)
    return false
  }
}

cost_validated = cost_limit_by_workspace()

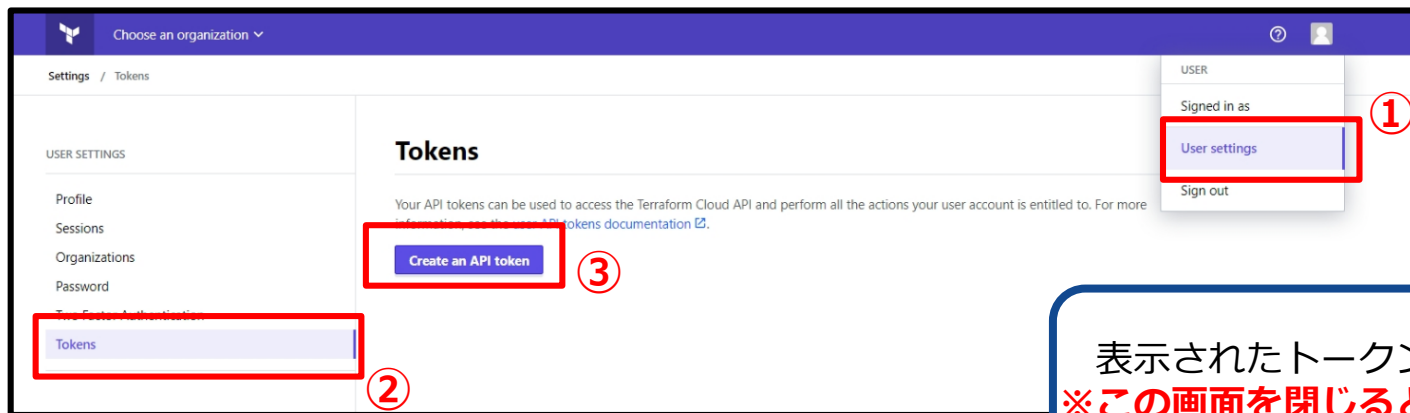
main = rule {
  cost_validated
}
```

### 3. 仕込み編

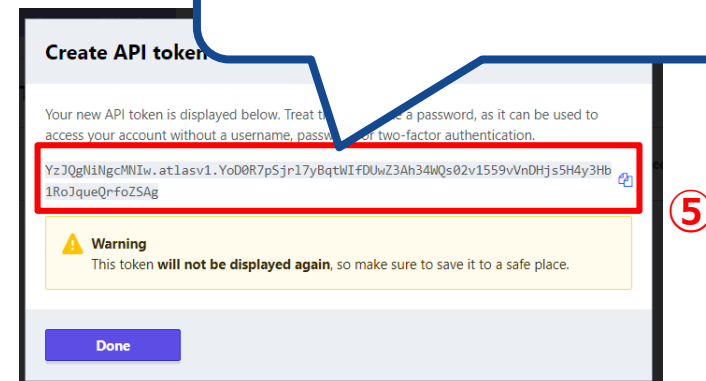
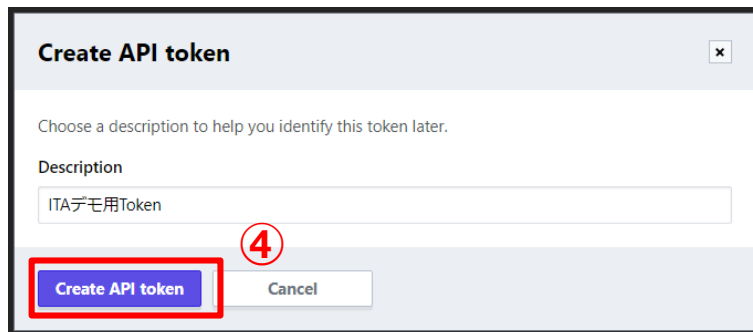
## 3.1 インターフェース情報の登録(1/2)

### User Tokenの発行

- Terraform DriverからTerraformに連携するために、Terraformからユーザートークンを発行する必要があります。
- ブラウザよりTerraformにログインし、[User Setting]→[Tokens]→[Creat an API token]の順に押下することで発行することができます。



表示されたトークンをメモしておく  
※この画面を閉じると再表示できません





## 3.1 インターフェース情報の登録(2/2)

### ■ インターフェース情報

- 連携するTerraformのHostnameと、発行したUserTokenを入力します

※ITAに連携できるTerraformは1つのみなので、インストール時に最初からある項目を「更新」して値を入力する必要があります。

メニュー：Terraform>インターフェース情報

- ① 一覧から登録済み項目の [更新] を押下する。
- ② 各項目へ下表のように入力し、[登録]を押下する。

メインメニュー

インターフェース情報

Organizations管理

Workspaces管理

Movement一覧

Module素材集

Policies管理

Policy Sets管理

PolicySet-Policy組付管理

フィルタ

オートフィルタ

一覧

No.	Hostname	User Token	Proxy Address	Proxy Port	状態監視周期(単位ミリ秒)	進行状態表示行数	NULL連携
1		.....			3000	1000	無効

※は必須項目です。

戻る 更新

Hostname

(Terraformのドメイン名)

User Token

(任意でご入力下さい)

## 3.2 Organizationの登録と連携(1/2)

### Organizationを登録する

Organizationを作成しましょう。

メニュー : Terraform > Organizations管理

- ① 登録 > 登録開始 を押下する。
- ② 各項目へ下表のように入力し、[登録]を押下する。

Organization ID	Organization Name*	Email address*	アクセス権	備考
自動入力			設定 アクセス許可ロール	

※\*は必須項目です。

[戻る](#) [登録](#)

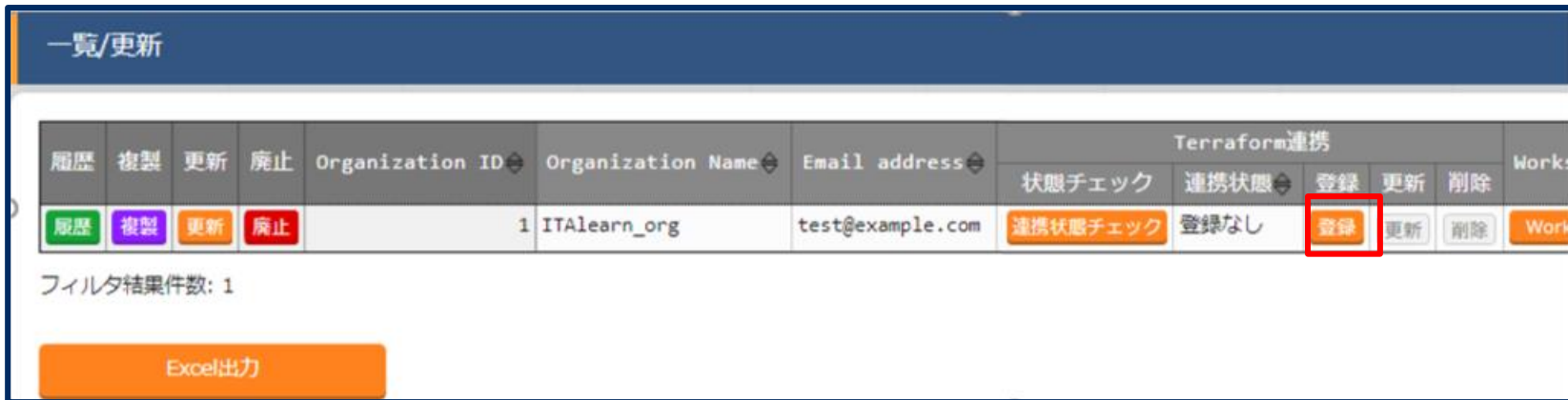
Organization Name	Email address
-------------------	---------------

ITAllearn_org	(任意でご入力下さい)
---------------	-------------

## 3.2 Organizationsの登録と連携(2/2)

### Organizationを連携する

- Organization管理からOrganizationの項目を作成した後、  
[連携状態チェック]で対象のTerraformに追加したOrganizationがあるかどうかをチェックすることができます。
- 「登録なし」であれば[登録]を押下することで対象のTerraformにOrganizationを作成できます。

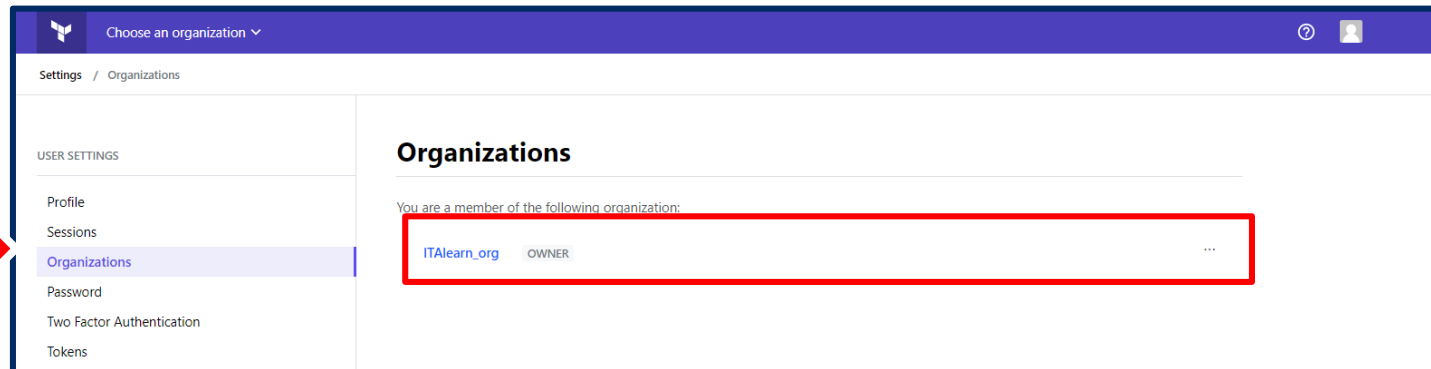


一覧/更新

履歴	複製	更新	廃止	Organization ID	Organization Name	Email address	Terraform連携					Work:
							状態チェック	連携状態	登録	更新	削除	
履歴	複製	更新	廃止	1	ITAlearn_org	test@example.com	連携状態チェック	登録なし	登録	更新	削除	Work

フィルタ結果件数: 1

Excel出力



Choose an organization

Settings / Organizations

USER SETTINGS

- Profile
- Sessions
- Organizations
- Password
- Two Factor Authentication
- Tokens

### Organizations

You are a member of the following organization:

ITAlearn_org	OWNER	...
--------------	-------	-----

## 3.3 Workspaceの登録と連携(1/2)

### Workspaceを登録する

Workspaceを作成しましょう。

メニュー：Terraform > Workspaces管理

- ① 登録 > 登録開始 を押下する。
- ② 各項目へ下表のように入力し、[登録]を押下する。

Workspace ID	Organization*	Workspace Name*	Terraform Version	アクセス権
自動入力	<input type="text"/>	<input type="text"/>	<input type="text"/>	設定 アクセス許可ロール

※\*は必須項目です

Organization	Workspace Name
ITAllearn_org	ITA-demo-AWS
ITAllearn_org	ITA-demo-Azure

## 3.3 Workspaceの登録と連携(2/2)

### Workspaceを連携する

- Workspaces管理からWorkspaceの項目を作成した後、  
[連携状態チェック]で対象のTerraformに追加したWorkspaceがあるかどうかをチェックすることができます。
- 「登録なし」であれば[登録]を押下することで対象のTerraformにWorkspaceを作成できます。

※WorkspaceはOrganization上に作成されるため、必ず先にOrganizationを対象のTerraformに作成しておく必要があります

履歴	複製	更新	廃止	Workspace ID	Organization	Workspace Name	Terraform Version	Terraform連携					Movement
								状態チェック	連携状態	登録	更新	削除	
履歴	複製	更新	廃止	1	ITAllearn_org	ITA-demo-AWS		連携状態チェック	登録済み	登録	更新	削除	Movement
履歴	複製	更新	廃止	2	ITAllearn_org	ITA-demo-Azure		連携状態チェック	登録なし	登録	更新	削除	Movement

フィルタ結果件数: 2

Excel出力

ITAllearn\_org / Workspaces

Workspaces 3 total

All 3 Success 0 Error 0 Needs Attention 0 Running 0

WORKSPACE NAME	RUN STATUS	RUN	REPO	LATEST CHANGE
ITA-demo-AWS				a few seconds ago
ITA-demo-Azure				9 minutes ago

## 3.4 作業パターン(Movement)の登録

### Movementを作成する

先のplaybookを関連付けるMovementを登録しましょう。

メニュー：Terraform > Movement一覧

- ① 登録 > 登録開始 を押下する。
- ② 各項目で下表のように選択または入力し、[登録]を押下する。

Movement ID	Movement名*	遅延タイマー	Terraform利用情報 Organization:Workspace*	アクセス権
自動入力				設定 アクセス許可ロー

※\*は必須項目です。

戻る 登録

Movement名	Terraform利用情報 Organization:Workspace
VM作成(AWS)	ITA-demo-AWS
VM作成(Azure)	ITA-demo-Azure

## 3.5 Module素材の登録

### Moduleを登録する

作成したModuleをITAに登録しましょう。

メニュー：Terraform > Module素材集

- ① 登録 > 登録開始 を押下する。
- ② [参照] からModuleを選択し、「事前アップロード」を行う。
- ③ 各項目へ下表のように入力し、「登録」を押下する。

Module素材名	Module素材
aws_create_instance_variables	aws_create_instance_variables.tf
aws_create_instance_body	aws_create_instance.tf
azure_create_instance_variables	azure_create_instance_variables.tf
azure_create_instance_body	azure_create_instance.tf

## 3.6 Policy素材の登録

### Policyを登録する

作成したPolicyをITAに登録しましょう。

メニュー：Terraform > Policies管理

- ① 登録 > 登録開始 を押下する。
- ② [参照] からPolicyを選択し、「事前アップロード」を行う。
- ③ 各項目へ下表のように入力し、「登録」を押下する。



登録

Policy ID	Policy名*	Policy素材*	アクセス権
自動入力		ファイルの選択 ファイルが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:	設定 アクセス許可ロール

※\*は必須項目です。

戻る 登録

Policy名	Policy素材
limit-proposed-monthly-cost	limit-proposed-monthly-cost.sentinel



## 3.7 Policy Setの登録

### Policy Setを登録する

Policy SetをITAに登録しましょう。

メニュー：Terraform > Policy Sets管理

- ① 登録 > 登録開始 を押下する。
- ② 各項目へ下表のように入力し、「登録」を押下する。

登録

PolicySet ID	PolicySet名*	アクセス権	
		設定	アクセス許可ロール
自動入力	<input type="text"/>	<input type="button" value="設定"/>	

※\*は必須項目です。

戻る

登録

PolicySet名

PolicySet\_demo

## 3.8 Policy SetとPolicyの紐付け

### Policy SetとPolicyを紐付ける

作成したPolicy SetとPolicyを紐付けましょう。

メニュー： Terraform > PolicySet-Policy紐付管理

- ① 登録 > 登録開始 を押下する。
- ② 各項目へ下表のように入力し、「登録」を押下する。

登録

PolicySet-Policy紐付ID	Policy Set*	Policy*
自動入力	<input type="text" value="自動入力"/>	<input type="text"/>

※\*は必須項目です。

戻る 登録

Policy Set	Policy
1:PolicySet_demo	1:limit-proposed-monthly-cost

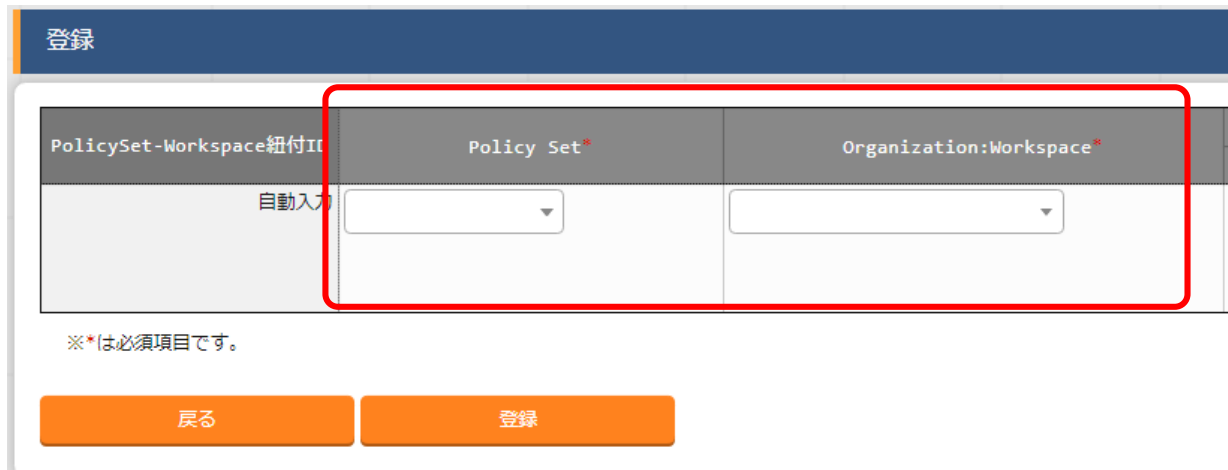
## 3.9 Policy SetとWorkspaceの紐付け

### Policy SetとWorkspaceを紐付ける

作成したPolicy SetとWorkspaceを紐付けましょう。

メニュー：Terraform > PolicySet-Workspace紐付管理

- ① 登録 > 登録開始 を押下する。
- ② 各項目へ下表のように入力し、「登録」を押下する。



登録

PolicySet-Workspace紐付ID	Policy Set*	Organization:Workspace*
自動入力	<input type="text"/>	<input type="text"/>

※\*は必須項目です。

戻る 登録

Policy Set	Organization:Workspace
1:PolicySet_demo	ITAllearn_org:ITA-demo-AWS
1:PolicySet_demo	ITAllearn_org:ITA-demo-Azure

## 3.10 MovementにModule素材を指定

### MovementにModuleを紐付ける

作成したMovementとModule素材を関連付けましょう。

メニュー: Terraform > Movement-Module紐付

- ① 登録 > 登録開始 を押下する。
- ② 各項目で下表のように選択または入力し、[登録]を押下する。

登録

紐付項目	Movement*	Module素材*	設定
自動入力	<input type="text"/>	<input type="text"/>	設定

※\*は必須項目です。

戻る 登録

Movement	Module素材
VM作成(AWS)	aws_create_instance_variables
VM作成(AWS)	aws_create_instance_body
VM作成(Azure)	azure_create_instance_valiables
VM作成(Azure)	azure_create_instance_body

## 4. 実行編

## 4.1 オペレーションの登録

### オペレーションを新規登録する

オペレーションを作成しましょう。

メニュー：基本コンソール > 投入オペレーション一覧

- ① 登録 > 登録開始 を押下する。
- ② 各項目へ下表のように入力し、[登録]を押下する。

登録

No.	オペレーションID	オペレーション名*	実施予定日時*	備考
自動入力	自動入力	<input type="text"/>	<input type="text"/>	

※\*は必須項目です。

戻る

登録

オペレーション名	実施予定日時
Terraform_demo	(任意でご入力下さい)

※ 「実施予定日時」は管理用の項目です。自動的に処理が実行されるわけではありません。

## 4.2 変数値の設定(1/4)

### 変数に数値を設定する

Moduleの変数に具体的な数値を設定しましょう。

メニュー: **Terraform > 代入値管理**

- ① 登録 > 登録開始 を押下する。
- ② 各項目で下表のように選択または入力し、[登録]を押下する。

登録

項番	オペレーション*	Movement*	変数名*	HCL設定*	Sensitive設定*	
自動入力	<input type="text"/>	<input type="text"/>	Movementを選択して下さい	OFF <input type="text"/>	OFF <input type="text"/>	<input type="text"/>

※\*は必須項目です。

戻る登録

## 4.2 変数値の設定(2/4)

### 変数に数値を設定する(1/3)

代入値の登録は以下の表を参考に行ってください。

オペレーション	Movement	変数名	具体値
Terraform_demo	VM作成(AWS)	security_group	ita-demo-sg※
Terraform_demo	VM作成(AWS)	key_name	ita-demo-key※
Terraform_demo	VM作成(AWS)	access_key	(AWSアクセスキー)
Terraform_demo	VM作成(AWS)	secret_key	(AWSシークレットキー)
Terraform_demo	VM作成(AWS)	region	(任意のリージョン)
Terraform_demo	VM作成(AWS)	tags_name	ita-demo-instance
Terraform_demo	VM作成(AWS)	hello_tf_instance_type	t2.large
Terraform_demo	VM作成(AWS)	hello_tf_instance_count	3
Terraform_demo	VM作成(AWS)	ami	(任意のAMI)

※セキュリティグループ、キーペアは事前に作成しておく必要があります。



## 4.2 変数値の設定(3/4)

### 変数に数値を設定する(2/3)

代入値の登録は以下の表を参考に行ってください。

オペレーション	Movement	変数名	具体値
Terraform_demo	VM作成(Azure)	subscription_id	(Azure認証情報)
Terraform_demo	VM作成(Azure)	tenant_id	
Terraform_demo	VM作成(Azure)	client_id	
Terraform_demo	VM作成(Azure)	client_secret	
Terraform_demo	VM作成(Azure)	resource_group_name	ita-demo-rg
Terraform_demo	VM作成(Azure)	location	ita-demo-web-azure
Terraform_demo	VM作成(Azure)	security_group	ita-demo-security-group
Terraform_demo	VM作成(Azure)	Vnet_name	ita-demo-vnet
Terraform_demo	VM作成(Azure)	Vnet_address_space	10.0.0.0/16
Terraform_demo	VM作成(Azure)	subnet_name	ita-demo-subnet
Terraform_demo	VM作成(Azure)	address_prefixes	10.0.2.0/24
Terraform_demo	VM作成(Azure)	public_ip_name	ita-demo-public-ip
Terraform_demo	VM作成(Azure)	allocation_method	Dynamic
Terraform_demo	VM作成(Azure)	domain_name_label	ita-demo-domain
Terraform_demo	VM作成(Azure)	network_interface_name	ita-demo-nwif

## 4.2 変数値の設定(4/4)

### 変数に数値を設定する(3/3)

代入値の登録は以下の表を参考に行ってください。

オペレーション	Movement	変数名	具体値
Terraform_demo	VM作成(Azure)	NIC_name	ita-demo-NIC
Terraform_demo	VM作成(Azure)	VM_name	ita-demo-web-azure
Terraform_demo	VM作成(Azure)	publisher	OpenLogic
Terraform_demo	VM作成(Azure)	offer	CentOS
Terraform_demo	VM作成(Azure)	sku	8_2
Terraform_demo	VM作成(Azure)	source_image_version	latest
Terraform_demo	VM作成(Azure)	os_disk_name	ita-demo-os-disk
Terraform_demo	VM作成(Azure)	storage_account_type	Standard_LRS
Terraform_demo	VM作成(Azure)	caching	ReadWrite
Terraform_demo	VM作成(Azure)	admin_username	ita-demo
Terraform_demo	VM作成(Azure)	ssh_public_key	(任意のSSH公開鍵)
Terraform_demo	VM作成(Azure)	VM_size	Standard_B2MS
Terraform_demo	VM作成(Azure)	VM_count	3

## 4.3 Planを確認

### Planを確認する

前項までの操作で、実行するMovementの作成と代入値の登録が終了しました。次にPlanを確認し、Moduleが定義されたポリシーに沿っているか確認します。

#### メニュー： Terraform > 作業実行

Menu

- メインメニュー
- インターフェース情報
- Organizations管理
- Workspaces管理
- Movement一覧
- Module素材集
- Policies管理
- Policy Sets管理
- PolicySet-Policy紐付管理
- PolicySet-Workspace紐付管理
- Movement-Module紐付
- 代入値自動登録設定
- 代入値管理
- 作業実行
- 作業状態確認
- 作業管理

Movement[フィルタ]

Movement[一覧]

選択	Movement ID	Movement名	オーケストラータ	遅延タイム	Terraform利用情報	アクセス権	備考	最終更新日時	最終更新者
					Organization:Workspace	アクセス許可ロー			
<input checked="" type="radio"/>	1	VM作成(AWS)	Terraform		ITAllearn_org:ITA-demo-AWS			2021/07/20 15:18:26	システム管理者
<input type="radio"/>	2	VM作成(Azure)	Terraform		ITAllearn_org:ITA-demo-Azure			2021/07/20 15:32:42	システム管理者

フィルタ結果件数: 2

オペレーション[フィルタ]

オペレーション[一覧]

選択	No.	オペレーションID	オペレーション名	実施予定日時	最終実行日時	アクセス権	備考	最終更新日時	最終更新者
						アクセス許可ロー			
<input checked="" type="radio"/>	1	1	Terraform_demo	2021/07/22 15:36				2021/07/20 15:36:52	システム管理者

フィルタ結果件数: 1

MovementID 1  
Movement名 VM作成(AWS)

**Plan確認** 実行

1 確認するMovementを選択する。

2 オペレーションを選択する。

3 [Plan確認]を押下する。

**Tips**  
[Plan確認]はPlan/PolicyCheckまでを行い、処理を停止します。

## 4.4 PolicyCheckログを確認

### PolicyCheckログを確認する

押下後の画面遷移先で、ステータスが「完了(異常)」となっていることが確認できます。  
画面下に移動し、PolicyCheckログを確認しましょう。

説明

対象作業

項目			値
作業No.			61
実行種別			Plan確認
ステータス			完了(異常)
呼出元Symphony			
呼出元Conductor			
実行ユーザ			システム管理者
Movement	ID		1
	名称		VM作成(AWS)
	遅延タイム(分)		
	Terraform利用情報	Organization:Workspace	ITAllearn_org:ITA-demo-AWS
		RUN-ID	run-GJFbJfkfFREKUwDb
オペレーション	No.		1
	名称		Terraform_demo
	ID		1
代入値			確認
入力データ	投入データ		<a href="#">InputData_0000000061.zip</a>
出力データ	結果データ		<a href="#">ResultData_0000000061.zip</a>
作業状況	予約日時		
	開始日時		2021/09/06 07:51:09
	終了日時		2021/09/06 07:51:41

進行状況(PolicyCheckログ)	
フィルタ: <input type="text"/> <input type="checkbox"/> 該当行のみ表示	
<pre>Sentinel Result: false  Sentinel evaluated to false because one or more Sentinel policies evaluated to false. This false was not due to an undefined value or runtime error.  1 policies evaluated.  ## Policy 1: limit-proposed-monthly-cost (hard-mandatory)  Result: false  Print messages: Proposed monthly cost 200.448 of workspace ITA-demo-AWS is over the limit: \$ {"coefficient": "50", "exponent": 0, FALSE - limit-proposed-monthly-cost.sentinel:34:1 - Rule "main"</pre>	

**Tips**

コストが\$50を大幅に  
上回っていることが確認できる

## 4.5 VMのサイズを変更して再度確認

### VMのサイズを変更して再度実行する。

最後に、デプロイするVMのサイズを変更して同様に実行します。

**Terraform > 代入値管理**から、下表を参考に具体値を変更し、

4.3同様にPlan確認をしましょう。

#### 変更前

オペレーション	Movement	変数名	具体値
Terraform_demo	VM作成(AWS)	hello_tf_instance_type	t2.large
Terraform_demo	VM作成(Azure)	VM_size	Standard_B2MS



デプロイするインスタンスサイズ  
AWS: **t2.large** → **t2.micro**  
Azure: **Standard\_B2MS** → **Standard\_B1LS**

#### 変更後

オペレーション	Movement	変数名	具体値
Terraform_demo	VM作成(AWS)	hello_tf_instance_type	<b>t2.micro</b>
Terraform_demo	VM作成(Azure)	VM_size	<b>Standard_B1LS</b>

**Tips**

## 4.6 再度PolicyCheckログを確認

### 再びPolicyCheckログを確認する

押下後の画面遷移先で、ステータスが「完了」となっていることが確認できます。  
画面下に移動しPolicyCheckログを確認後、Movementを実際に実行しましょう

説明			
対象作業			
項目		値	
作業No.		65	
実行種別		Plan確認	
ステータス		完了	
呼出元Symphony			
呼出元Conductor			
実行ユーザ		システム管理者	
Movement	ID	1	
	名称	VM作成(AWS)	
	遅延タイム(分)		
	Terraform利用情報	Organization:Workspace ITAlearn_org:ITA-demo-AWS RUN-ID run-vdyP2TiJP2xvgSFT	
オペレーション	No.	1	
	名称	Terraform_demo	
	ID	1	
代入値		確認	
入力データ	投入データ	<a href="#">InputData_0000000065.zip</a>	
出力データ	結果データ	<a href="#">ResultData_0000000065.zip</a>	
作業状況	予約日時		
	開始日時	2021/09/06 08:31:07	
		終了日時	
		2021/09/06 08:31:43	

#### 進行状況(PolicyCheckログ)

フィルタ:  ☐ 該当行のみ表示

```
Sentinel Result: true  
  
This result means that Sentinel policies returned true and the protected  
behavior is allowed by Sentinel policies.  
  
1 policies evaluated.  
  
## Policy 1: limit-proposed-monthly-cost (hard-mandatory)  
  
Result: true  
  
Print messages:  
proposed monthly cost 25.056 of workspace ITA-demo-AWS is under the limit: $ {"coefficient": "50", "exponent": 0, "flo  
TRUE - limit-proposed-monthly-cost.sentinel:34:1 - Rule "main"
```

**Tips**

変更後、コストが\$50を  
下回っていることが確認できた。

## 4.7 作業実行

### Movementを実行する

実行するModuleが定義したポリシーを適用していることが確認できました。  
最後にMovementを実行し、結果を対象ホストで確認してください。

メニュー： Terraform > 作業実行

**1** 実行するMovementを選択する。

選択	Movement ID	Movement名	オーケストラータ	遅延タイマー	Terraform利用情報	アクセス権	備考	最終更新日時	最終更新者
<input type="radio"/>	1	VM作成(AWS)	Terraform		Organization:Workspace	アクセス許可ロー		2021/07/20 15:18:26	システム管理者
<input type="radio"/>	2	VM作成(Azure)	Terraform		ITAllearn_org:ITA-demo-AWS			2021/07/20 15:32:42	システム管理者

フィルタ結果件数: 2

**2** オペレーションを選択する。

選択	No.	オペレーションID	オペレーション名	実施予定日時	最終実行日時	アクセス権	備考	最終更新日時	最終更新者
<input type="radio"/>	1	1	Terraform_demo	2021/07/22 15:36		アクセス許可ロー		2021/07/20 15:36:52	システム管理者

フィルタ結果件数: 1

**3** [実行]を押下する。

MovementID 1  
Movement名 VM作成(AWS)

Plan確認 実行

**Tips**  
実行後、自動で【作業状態確認】へ画面遷移します。

## 4.8 実行状態確認

### Movementの詳細結果を確認する

実行後の画面遷移先で、実行ステータスやログを確認することができます。  
投入データや出力データを確認することも可能です。

Menu

メインメニュー

インターフェース情報

Organizations管理

Workspaces管理

Movement一覧

Module素材集

Policies管理

Policy Sets管理

PolicySet-Policy紐付管理

PolicySet-Workspace紐付管理

Movement-Module紐付

代入値自動登録設定

代入値管理

作業実行

作業状態確認

作業管理

説明

対象作業

項目	値
作業No.	2
実行種別	通常
ステータス	完了
呼出元Symphony	
呼出元Conductor	
実行ユーザ	システム管理者
ID	1
名称	VM作成(AWS)
遅延タイム(分)	
Terraform利用情報	Organization:Workspace ITAlearn_org:ITA-demo-AWS RUN-ID run-Dc9RkRGdQekkCrHc
No.	1
名称	Terraform_demo
ID	1
代入値	確認
入力データ	投入データ
出力データ	結果データ
予約日時	
作業状況	開始日時 2021/07/21 14:00:00 終了日時 2021/07/21 14:00:00

Tips

投入データや結果データをまとめたZipファイルをダウンロードできます

進行状況(Planログ)

フィルタ:  ☐ 該当行のみ表示

```
+ device_index = (known after apply)
+ network_interface_id = (known after apply)
}

+ root_block_device {
+ delete_on_termination = (known after apply)
+ device_name = (known after apply)
+ encrypted = (known after apply)
+ iops = (known after apply)
+ kms_key_id = (known after apply)
+ tags = (known after apply)
+ throughput = (known after apply)
+ volume_id = (known after apply)
+ volume_size = (known after apply)
+ volume_type = (known after apply)
}

}

Plan: 3 to add, 0 to change, 0 to destroy.
```

Tips

結果はAWS、Azureアカウントにアクセスし各3台のVMが新たに作成されているかを確認してください。



## 4.9 数値を変更して再度実行(1/2)

### ■ インスタンス数を変更して再度実行する。

最後に、デプロイするインスタンス数を変更して同様に実行します。

**Terraform > 代入値管理**から、下表を参考に具体値を変更し、

4.3同様に作業実行しましょう。

#### 変更前

オペレーション	Movement	変数名	具体値
Terraform_demo	VM作成(AWS)	hello_tf_instance_count	3
Terraform_demo	VM作成(Azure)	VM_count	3



#### 変更後

オペレーション	Movement	変数名	具体値
Terraform_demo	VM作成(AWS)	hello_tf_instance_count	5
Terraform_demo	VM作成(Azure)	VM_count	1

デプロイするインスタンス数を  
AWS: 3台→5台に増設  
Azure: 3台→1台に減設

**Tips**

## 4.9 数値を変更して再度実行(2/2)

### ■ インスタンスの増減を確認

AWS・Azureにブラウザから接続し、

VMインスタンスの数が増減した通りに増減しているか確認しましょう。

#### AWS

<input type="checkbox"/>	Name	インスタンス ID	イン
<input type="checkbox"/>	ita-demo-instance-1	i-01ce909628624f4fc	✓
<input type="checkbox"/>	ita-demo-instance-3	i-0a9937555aca7be96	✓
<input type="checkbox"/>	ita-demo-instance-2	i-0d34ba73c42144d54	✓



<input type="checkbox"/>	Name	インスタンス ID	イン
<input type="checkbox"/>	ita-demo-instance-1	i-01ce909628624f4fc	✓
<input type="checkbox"/>	ita-demo-instance-3	i-0a9937555aca7be96	✓
<input type="checkbox"/>	ita-demo-instance-2	i-0d34ba73c42144d54	✓
<input type="checkbox"/>	ita-demo-instance-5	i-004f22d98cf7f6303	✓
<input type="checkbox"/>	ita-demo-instance-4	i-0f42e3212f538c8d2	✓

#### Azure

<input type="checkbox"/>	名前 ↑↓
<input type="checkbox"/>	ita-demo-nwif-0
<input type="checkbox"/>	ita-demo-nwif-1
<input type="checkbox"/>	ita-demo-nwif-2
<input type="checkbox"/>	ita-demo-os-disk-0
<input type="checkbox"/>	ita-demo-os-disk-1
<input type="checkbox"/>	ita-demo-os-disk-2
<input type="checkbox"/>	ita-demo-public-ip-0
<input type="checkbox"/>	ita-demo-public-ip-1
<input type="checkbox"/>	ita-demo-public-ip-2
<input type="checkbox"/>	ita-demo-security-group
<input type="checkbox"/>	ita-demo-vnet
<input type="checkbox"/>	ita-demo-web-azure-0
<input type="checkbox"/>	ita-demo-web-azure-1
<input type="checkbox"/>	ita-demo-web-azure-2



<input type="checkbox"/>	名前 ↑↓
<input type="checkbox"/>	ita-demo-nwif-0
<input type="checkbox"/>	ita-demo-os-disk-0
<input type="checkbox"/>	ita-demo-public-ip-0
<input type="checkbox"/>	ita-demo-security-group
<input type="checkbox"/>	ita-demo-vnet
<input type="checkbox"/>	ita-demo-web-azure-0



**Exastro**