

# BPA使用手冊RBAC許可權v5.1

- [簡介](#)
- [新許可權](#)
  - [定義靜態許可權](#)
- [在BPA中註冊許可權](#)
- [對映許可權](#)
  - [定義許可權對映](#)
  - [更新許可權對映](#)

## 簡介

基於角色的訪問控制(RBAC)是一種根據使用者在企業中的角色來限制訪問的方法。預設角色可用，並且可以通過選擇首選許可權並將這些角色對映到使用者組來建立新角色。將角色對映到使用者組允許該使用者組中的所有使用者執行與該角色關聯的操作。

## 新許可權

本節介紹微服務如何定義其自己的RBAC許可權集。

### 定義靜態許可權

要定義靜態許可權，請執行以下操作：

1. 建立JSON檔案，列出microservice/src/config/device-permissions-spec.json路徑中的所有許可權。
2. 運行以下JSON代碼以列出許可權：

```
{
  "service": "service-name",
  "permissions": [
    {
      "group": "group-name",
      "actions": [
        {"name": "action-name", "displayName": "Display name to be shown in roles page"}
      ]
    }
  ]
}
```

請參閱以下示例，在憑據組中建立檢視、管理和刪除許可權。

```
{
  "service": "AssetManagerService",
  "permissions": [
    {
      "group": "credential",
      "actions": [
        {"name": "view", "displayName": "View Asset Credentials"},
        {"name": "manage", "displayName": "Manage Asset Credentials"},
        {"name": "remove", "displayName": "Remove Asset Credentials"}
      ]
    }
  ]
}
```

建立許可權的示例

## 在BPA中註冊許可權

要註冊許可權，請執行以下操作：

1. 從@ciseco-bpa-platform/mw-util-common-app匯入rbacSpecProcessorHelper。

```
const { rbacHelper, rbacSpecProcessorHelper } = require('@ciseco-bpa-platform/mw-util-common-app');
```

2. 運行以下命令以使用rbacSpecProcessorHelper處理JSON檔案中列出的許可權。

```
let resources = require('./config/asset-manager-permissions-spec');
let rbacSpecProcessorObj = rbacSpecProcessorHelper.getRbacSpecProcessorUtil();
rbacSpecProcessorObj.setSpec(resources);
await rbacSpecProcessorObj.process();
```

---

 附註：使用者還可以參考@ciseco-bpa-platform/mw-util-common-app中共用的方法來註冊檢視、管理和刪除許可權。

---

使用者可以在「建立角色」和「編輯角色」頁中檢視已註冊的許可權。

## 建立角色

## 對映許可權

預設情況下，在上一節中建立的許可權可以對映到角色，也可以通過定義與角色的對映來對映。

## 定義許可權對映

要將定義的許可權對映到預設角色，請在microservice/src/config/device-role-permissions-mapping-spec.json路徑中建立新的JSON檔案。

在下面的示例中，三(3)個許可權對映到BPA超級管理員角色，兩(2)個許可權對映到Tenant Admin角色，一(1)個許可權對映到Network Operator角色。

 附註：服務名稱、組名稱和操作名稱應與JSON檔案中的定義相同。

```
{
  "service": "AssetManagerService",
  "roles": [
    {
      "name": "BPA Super Admin",
      "permissions": [
        {"name": "view", "displayName": "View Asset Credentials", "group": "credential"},
        {"name": "remove", "displayName": "Remove Asset Credentials", "group": "credential"},
        {"name": "manage", "displayName": "Manage Asset Credentials Add, Update and Delete Asset C"}
      ]
    },
    {
      "name": "Tenant Admin",
```

```

    "permissions": [
      {"name": "view", "displayName": "View Asset Credentials", "group": "credential"},
      {"name": "remove", "displayName": "Remove Asset Credentials", "group": "credential"}
    ]
  },
  {
    "name": "Network Operator",
    "permissions": [
      {"name": "view", "displayName": "View Asset Credentials", "group": "credential"}
    ]
  }
]
}

```

## 更新許可權對映

要更新許可權對映，請執行以下操作：

1. 從@ciseco-bpa-platform/mw-util-common-app匯入rbacHelper、rbacPermissionMappingHelper。

```
const { rbacHelper, rbacPermissionMappingHelper } = require('@ciseco-bpa-platform/mw-util-common-app');
```

2. 運行以下命令以更新與角色的許可權對映：

```

let rbacUtilObj = rbacHelper.getRbacUtilObj();
let registryDetails = await rbacUtilObj.getRegistryByName({}, 'device-role-permissions-mapping-spec');
if (Array.isArray(registryDetails) && registryDetails.length === 0) {
  let rbacPermissionMappingSpecProcessorObj = rbacPermissionMappingHelper.getRbacSpecProcessorUtil();
  let rolePermissions = require('./config/device-role-permissions-mapping-spec');
  rbacPermissionMappingSpecProcessorObj.setSpec(rolePermissions);
  await rbacPermissionMappingSpecProcessorObj.process();
  let rbacUtilObjRegister = rbacHelper.getRbacUtilObj();
  await rbacUtilObjRegister.addRegistry({}, 'device-role-permissions-mapping-spec');
}

```

3. 運行以下命令以處理檔案、建立許可權清單以及對映許可權與角色：

```

async function permissionsSpecProcessor() {
  let resources = require('./config/device-permissions-spec');
  let rbacSpecProcessorObj = rbacSpecProcessorHelper.getRbacSpecProcessorUtil();
  rbacSpecProcessorObj.setSpec(resources);
  await rbacSpecProcessorObj.process();

  let rbacUtilObj = rbacHelper.getRbacUtilObj();

```

```
let registryDetails = await rbacUtilObj.getRegistryByName({} 'device-role-permissions-mapping-spec')
if (Array.isArray(registryDetails) && registryDetails.length === 0) {
  let rbacPermissionMappingSpecProcessorObj = rbacPermissionMappingHelper.getRbacSpecProcessorUtil()
  let rolePermissions = require('./config/device-role-permissions-mapping-spec');
  rbacPermissionMappingSpecProcessorObj.setSpec(rolePermissions);
  await rbacPermissionMappingSpecProcessorObj.process();
  let rbacUtilObjRegister = rbacHelper.getRbacUtilObj();
  await rbacUtilObjRegister.addRegistry({}, 'device-role-permissions-mapping-spec');
}
}
```

---

 附註：使用者還可以參考@[cisco-bpa-platform/mw-util-common-app](#)中共用的方法來更新角色許可權對映。

---

## 關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。