BPA 사용 설명서 RBAC 권한 v5.1

소개

RBAC(Role Based Access Control)는 기업 내 사용자의 역할에 따라 액세스를 제한하는 방법입니다. 기본 역할을 사용할 수 있으며 기본 권한을 선택하고 이러한 역할을 사용자 그룹에 매핑하여 새 역할을 만들 수 있습니다. 사용자 그룹에 역할을 매핑하면 해당 사용자 그룹의 모든 사용자가 역할과 관련된 작업을 수행할 수 있습니다.

새 권한

이 섹션에서는 마이크로서비스가 자체 RBAC 권한 집합을 정의하는 방법에 대해 설명합니다.

정적 권한 정의

정적 권한을 정의하려면

1. microservice/src/config/device-permissions-spec.json 경로의 모든 권한을 나열하는 JSON 파일을 만듭니다.
2. 다음 JSON 코드를 실행하여 권한을 나열합니다.

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

자격 증명 그룹 내에서 보기, 관리 및 사용 권한을 만들려면 다음 예를 참조하십시오.

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

권한 생성 예

BPA에서 권한 등록

권한을 등록하려면

1. @cisco-bpa-platform/mw-util-common-app에서 rbacSpecProcessorHelper를 가져옵니다.

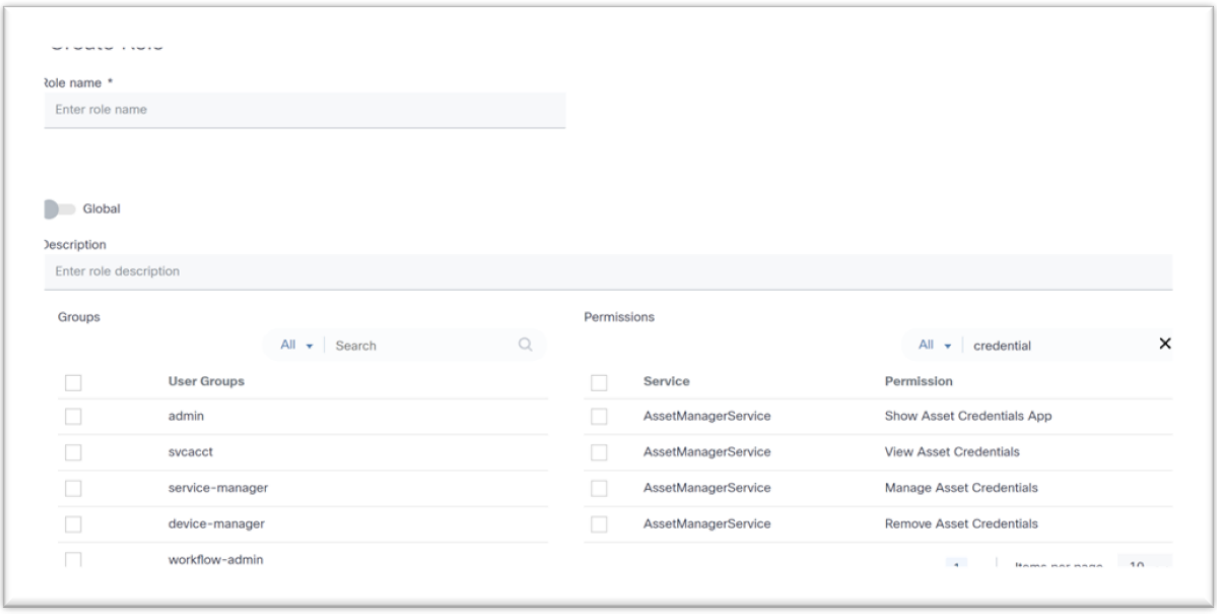
```
const { rbacHelper, rbacSpecProcessorHelper } = require('@cisco-bpa-platform/mw-util-common-app');
```

2. rbacSpecProcessorHelper를 사용하여 JSON 파일에 나열된 권한을 처리하려면 다음 명령을 실행합니다.

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

✎ 참고: 사용자는 @cisco-bpa-platform/mw-util-common-app에서 공유하는 메서드를 참조하여 보기, 관리 및 권한 제거를 등록할 수도 있습니다.

사용자는 Create Role(역할 생성) 및 Edit Role(역할 수정) 페이지에서 등록된 권한을 볼 수 있습니다.



역할 생성

매핑 권한

이전 섹션에서 만든 권한은 기본적으로 역할에 매핑하거나 역할과의 매핑을 정의하여 역할에 매핑할 수 있습니다.

권한 매핑 정의

정의된 권한을 기본 역할에 매핑하려면 microservice/src/config/device-role-permissions-mapping-spec.json 경로에 새 JSON 파일을 만듭니다.

아래 예에서 3개의 권한은 BPA 수퍼 관리자 역할에 매핑되고, 2개는 테넌트 관리자 역할에 매핑되며, 1개는 네트워크 운영자 역할에 매핑됩니다.

✎ 참고: 서비스 이름, 그룹 이름 및 작업 이름은 JSON 파일과 동일하게 정의되어야 합니다.

```
{
    "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"}
            ]
        }
    ]
}
```

권한 매핑 업데이트

권한 매핑을 갱신하려면

1. rbacHelper, rbacPermissionMappingHelper를 @cisco-bpa-platform/mw-util-common-app에서 가져옵니다.

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

2. 다음 명령을 실행하여 역할의 권한 매핑을 업데이트합니다.

```
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. 다음 명령을 실행하여 파일을 처리하고, 권한 목록을 만들고, 역할에 권한을 매핑합니다.

```
async function permissionsSpecProcessor() {
    let resources = require('./config/device-permissions-spec');
    let rbacSpecProcessorObj = rbacSpecProcessorHelper.getRbacSpecProcessorlltil();
    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.getRbacSpecProcessorUti
        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');
    }
}
```

참고: 사용자는 @cisco-bpa-platform/mw-util-common-app에서 공유하는 메서드를 참조하여 역할로 권한 매핑을 업데이트할 수도 있습니다.

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.