

# ISE ID 그룹, REST API를 통한 사용자 생성 및 수정

## 목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[구성](#)

[1. ID 그룹 생성](#)

[2. ID 그룹 상세내역 검색](#)

[3. 사용자 생성](#)

[4. 사용자 세부사항 검색](#)

[5. 사용자 세부사항 수정](#)

[다음을 확인합니다.](#)

## 소개

이 문서에서는 ID 관리 자동화에 사용할 수 있는 Rest API를 사용하여 ID 그룹 및 사용자를 생성하고 수정하는 방법에 대해 설명합니다. 이 장에서 설명하는 절차는 샘플 독립형 ISE 구축 및 JSON 형식의 Rest API Firefox Client(REST)를 기반으로 합니다.

## 사전 요구 사항

### 요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- Cisco ISE(Identity Services Engine)
- REST API
- JSON

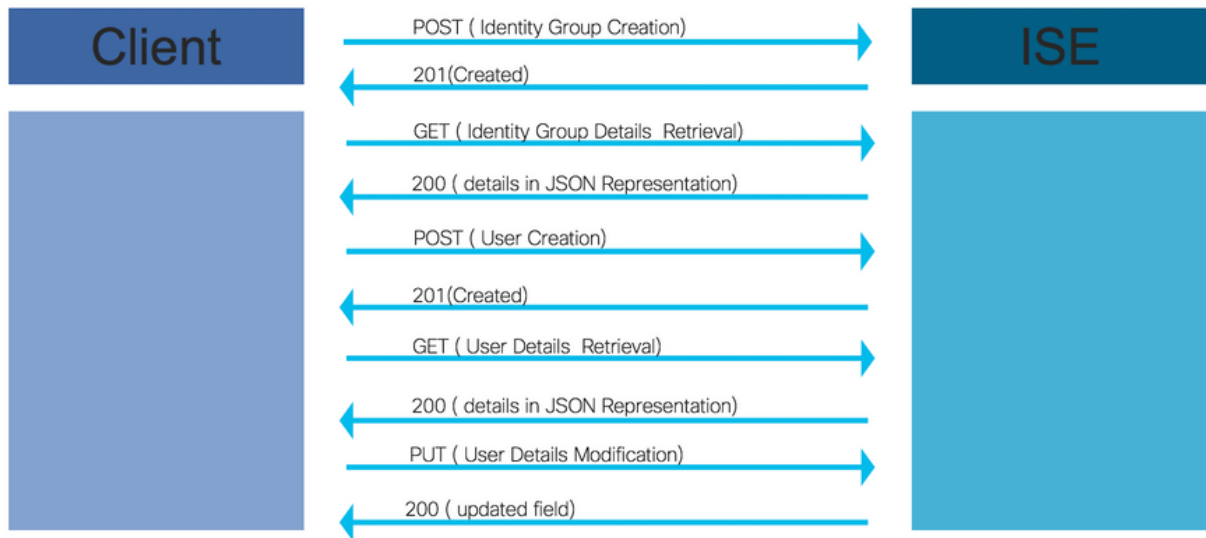
### 사용되는 구성 요소

이 문서는 특정 소프트웨어 및 하드웨어 버전으로 제한되지 않으며 REST API를 통한 샘플 컨피그레이션 가이드일 뿐입니다.

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 이해해야 합니다.

## 구성

## API Request and Response flow



다음은 API를 통한 작업의 샘플 단계로서, 고유한 호출을 작성하기 위한 참조로 사용할 수 있습니다.

### 1. ID 그룹 생성

POST 방법의 도움을 받아 ID 그룹을 생성합니다.

API 호출 URL:

<https://<ISE IP>:9060/ers/config/identityGroup>

API 호출의 헤더:

HTTP 'Content-Type' 헤더: application/json

HTTP 'Accept' 헤더: application/json

ID 그룹 생성을 위한 JSON 코드

```
{ "IdentityGroup": { "name": "
```

예:

Request ↗ ⚙ +

POST  Send request

**Headers** ▾

Content-Type	application/json	🗑
Accept	application/json	🗑

[+Add header](#)

**Basic auth** ▾

ersadmin	.....	<input type="checkbox"/> Show password?
----------	-------	---

**Request body** ▾

Type

```

{
  "IdentityGroup": {
    "name": "testusergroup",
    "description": " User group for testing",
    "parent": "NAC Group:NAC:IdentityGroups:User Identity Groups"
  }
}
```

Response (0.711s) - https://10.71.244.140:9060/ers/config/identitygroup

**201** Created

[Headers >](#)

## 2. ID 그룹 상세내역 검색

GET 메서드의 도움말을 사용하여 ID 그룹 세부 정보를 가져옵니다.

API 호출 URL:

https://<ISE IP>:9060/ers/config/identityGroup?filter=name.CONTAINS.<ID 그룹 이름>

API 호출의 헤더:

HTTP 'Content-Type' 헤더: application/json  
 HTTP 'Accept' 헤더: application/json

예:

GET

**Headers** ▾

Content-Type	application/json	🗑
Accept	application/json	🗑

[+Add header](#)

**Basic auth** ▾

ersadmin	.....	<input type="checkbox"/> Show password?
----------	-------	---

Response (0.786s) - https://10.71.244.140:9060/ers/config/identitygroup?filter=name.CONTAINS.testusergroup

**200** OK

**Headers** >

```
{
  "SearchResult": {
    "total": 1,
    "resources": [
      {
        "id": "b6ae7220-4289-11ea-a840-cee7c3fe1b0d",
        "name": "testusergroup",
        "description": " User group for testing",
        "link": {
          "rel": "self",
          "href": "https://10.71.244.140:9060/ers/config/identitygroup/b6ae7220-4289-11ea-a840-cee7c3fe1b0d",
          "type": "application/xml"
        }
      }
    ]
  }
}
```

**참고:**이 ID 그룹의 사용자를 만들려면 ID(ID 그룹 세부 정보에서 수신)가 필요합니다.

### 3. 사용자 생성

POST 메서드의 도움말을 사용하여 사용자를 생성합니다.

API 호출 URL:

**https://<ISE IP>:9060/ers/config/internaluser/**

API 호출의 헤더:

HTTP 콘텐츠 유형 헤더: application/json  
 HTTP Accept 헤더: application/json

사용자 생성을 위한 JSON 코드:

```
{ "InternalUser": { "name": "
```

예:

Request ↗ ⚙ +

POST  Send request

**Headers** ▾

Content-Type	application/json	🗑
Accept	application/json	🗑

[+Add header](#)

**Basic auth** ▾

ersadmin	.....	<input type="checkbox"/> Show password?
----------	-------	---

**Request body** ▾

Type

```

{
  "InternalUser": {
    "name": "test123",
    "email": "test@cisco.com",
    "enabled": true,
    "password": "Letmein11",
    "firstName": "test",
    "lastName": "test",
    "changePassword": false,
    "identityGroups": "b6ae7220-4289-11ea-a840-cee7c3fe1b0d",
  }
}

```

Response (80.364s) - https://10.71.244.140:9060/ers/config/internaluser/

**201** Created

[Headers >](#)

## 4. 사용자 세부사항 검색

GET 메서드의 도움말을 사용하여 사용자 세부 정보를 가져옵니다.

API 호출 URL:

https://<ISE IP>:9060/ers/config/internaluser

**참고:** 이 URL을 사용하여 사용자를 필터링할 수 있습니다. firstName, lastName, identityGroup, name, description, email, enabled를 사용하여 사용자를 필터링할 수 있습니다.

이메일 ID는 각 사용자에게 대해 고유하므로 이메일 ID로 사용자 세부사항을 필터링하는 것이 좋습니다.

https://<ISE IP>:9060/ers/config/internaluser?filter=<필터링에 사용되는 필드의 이름>.CONTAINS.<필터링에 사용되는 필드의 값>

API 호출의 헤더:

HTTP 콘텐츠 유형 헤더:	application/json
HTTP Accept 헤더:	application/json

예:

Request

GET 140:9060/ers/config/internaluser?filter=email.CONTAINS.test@cisco.com Send request

Headers

Content-Type	application/json	🗑
Accept	application/json	🗑

+Add header

Basic auth

ersadmin	.....	<input type="checkbox"/> Show password?
----------	-------	---

Response (1.348s) - https://10.71.244.140:9060/ers/config/internaluser?filter=email.CONTAINS.test@cisco.com

200 OK

Headers >

```
{
  "SearchResult": {
    "total": 1,
    "resources": [
      {
        "id": "be9aee27-22f2-4a77-a8e9-3092ba3b636e",
        "name": "test123",
        "link": {
          "rel": "self",
          "href": "https://10.71.244.140:9060/ers/config/internaluser/be9aee27-22f2-4a77-a8e9-3092ba3b636e",
          "type": "application/xml"
        }
      }
    ]
  }
}
```

참고: 여기에 수신된 IDandName은 사용자의 암호나 기타 정보를 업데이트하는 데 필요합니다. hrefURL은 사용자 정보를 업데이트하는 데 사용됩니다.

## 5. 사용자 세부사항 수정

PUT 메서드의 도움말을 사용하여 사용자 암호를 수정합니다.

API 호출 URL:

https://<ISE IP>:9060/ers/config/internaluser/<4단계에서 설명한 프로세스를 사용하여 사용자 ID를 받았습니>

위 내용은 4단계에서 설명한 프로세스를 사용하여 받은 href URL입니다.

API 호출의 헤더:

HTTP 콘텐츠 유형 헤더: application/json  
HTTP Accept 헤더: application/json

사용자 자격 증명 수정에 대한 JSON 코드:

```
{ "InternalUser": { "id": "
```

예:

The screenshot shows a REST client interface with the following details:

- Request:** Method: PUT, URL: 140:9060/ers/config/internaluser/be9aee27-22f2-4a77-a8e9-3092ba3b636e, Send request button.
- Headers:** Content-Type: application/json, Accept: application/json.
- Basic auth:** Username: ersadmin, Password: [redacted], Show password? checkbox.
- Request body:** Type: Custom, JSON body: 

```
{  "InternalUser": {    "id": "be9aee27-22f2-4a77-a8e9-3092ba3b636e",    "name": "test123",    "password": "Letmein!3",    "enablePassword": "Letmein!3"  } }
```
- Response:** (2.177s) - https://10.71.244.140:9060/ers/config/internaluser/be9aee27-22f2-4a77-a8e9-3092ba3b636e, 200 OK.

## 다음을 확인합니다.

ID 그룹을 확인하려면 ISE GUI에서 **Administration(관리) > Identity Management(ID 관리) > Groups(그룹) > Identity Groups(ID 그룹) > User Identity Group(사용자 ID 그룹)**으로 이동합니다.

Verify users(사용자 확인)는 **Administration(관리) > Identity Management(ID 관리) > Identities(ID) > Users(사용자)**를 ISE GUI로 이동합니다.