Mopuan Roller ve Yetkiler

Mopuan entegrasyonu için kullanım kılavuzu

opsenseapi · Sürüm 1.0 · 28 Ekim 2026

1. Bu servisler ne işe yarar

OpSense portalında her çalışanın bir **rolü** vardır. Bir rolün:

- bir **adı** (örneğin "Supervisor" veya "Housekeeper"),
- çalışanın şirket verilerinin ne kadarını görebileceğini belirleyen bir **hiyerarşi seviyesi**,
- bir **yetki matrisi** vardır: her portal modülü için rolün görüntüleme, oluşturma, düzenleme ve silme yetkisi olup olmadığı.

Çalışanlar birbirine de bağlıdır: bir çalışanın bir veya daha fazla **yöneticisi** (supervisor) olabilir ve bir yöneticinin bir **ekibi** (ona bağlı çalışanlar) vardır.

Mopuan kendi rollerini tutmaz. Bu üç servis, aynı kuralları Mopuan'da uygulayabilmeniz için OpSense rollerini, yetkilerini ve hiyerarşisini size verir.

Servis	Ne için kullanılır
`SRVGetUserRoles`	Bir çalışanı (veya tüm çalışanları) rolü, yetkileri, yöneticisi, ekibi ve görünürlüğüyle tek bir nesnede almak için. Bir çalışan giriş yaptığında veya tek bir çalışanı gösterirken kullanın.
`SRVGetRoles`	Şirketin rol listesini, her rolün yetkileriyle birlikte almak için. Rolleri kurarken veya güncellerken kullanın.
`SRVGetCompanyRoleData`	Tüm şirketi tek çağrıda almak için: rol listesi ve her çalışanın yöneticisi, ekibi ve görünürlüğü. Tam senkronizasyon için kullanın, örneğin günde bir kez.

Üç servis de yalnızca okuma yapar. Çağırmak hiçbir şeyi değiştirmez.

2. Hiyerarşi seviyeleri

Seviye numarası **düştükçe** yetki **artar**.

Seviye	Ad	Görebildikleri
0	Super Admin	Şirketin tüm çalışanları
1	Admin	Şirketin tüm çalışanları
2	Supervisor	Kendisi, doğrudan bağlı çalışanları ve onlara doğrudan bağlı çalışanlar (iki seviye aşağı)
3	User	Yalnızca kendisi

Seviye 0 yalnızca yerleşik "Admin" rolü tarafından kullanılır. Herkesi görmesi gereken CEO veya genel müdür gibi roller portalda seviye 1 olarak oluşturulur.

Portalda **yalnızca atanan** (only assigned) olarak işaretlenmiş bir çalışan, rolünün seviyesi ne olursa olsun yalnızca kendisini görür.

Görünürlüğü kendiniz hesaplamanız gerekmez. Her çalışan nesnesinde hazır bir `visibility` alanı vardır (bkz. bölüm 6).

3. Kimlik doğrulama

Şirketin `api_key` değerini, diğer OpSense servislerinde olduğu gibi normal bir istek parametresi olarak, POST alanı şeklinde gönderin. HTTP başlığı (header) olarak veya URL içinde göndermeyin.

Anahtar, çağrının hangi şirket için olduğunu belirler. Yalnızca o şirketin rollerini ve çalışanlarını alırsınız.

> **Bu servisleri yalnızca kendi sunucunuzdan çağırın.** `api_key`, şirketin tüm çalışanlarına ve yetkilerine erişim sağlar. Anahtarı hiçbir zaman Mopuan mobil uygulamasına veya bir web sayfasına koymayın. Mobil uygulamanın bir çalışanın yetkilerine ihtiyacı varsa sizin sunucunuza sormalı; sunucunuz da giriş yapmış çalışan için `SRVGetUserRoles` servisini çağırmalıdır.

Durum	Yanıt
`api_key` eksik	`error_code` 10015
`api_key` geçersiz veya pasif	`error_code` 10016

Her yanıt aynı şekilde sarılır:

```
{"result": { ... başarılı yanıt veya hata ... }}
```

4. İstekler

```
POST https://<api-host>/v1/<servis>
Content-Type: application/x-www-form-urlencoded
```

Parametreleri **form alanları** olarak gönderin. JSON istek gövdesi okunmaz.

SRVGetUserRoles

Alan	Zorunlu	Notlar
`api_key`	**Zorunlu**	Bkz. Kimlik doğrulama.
`mopuan_user_id`	İsteğe bağlı	Yalnızca bu çalışanı döndürür. Sizin Mopuan çalışan id'niz.
`user_id`	İsteğe bağlı	Yalnızca bu çalışanı OpSense kullanıcı id'si ile döndürür. `mopuan_user_id` gönderilirse dikkate alınmaz.

Filtre gönderilmezse tüm aktif çalışanlar döner.

```
curl -X POST "https://<api-host>/v1/SRVGetUserRoles" \
 -d "api_key=<api_key>" \
 -d "mopuan_user_id=345"
```

SRVGetRoles

Alan	Zorunlu	Notlar
`api_key`	**Zorunlu**	Bkz. Kimlik doğrulama.
`reference_id`	İsteğe bağlı	Yalnızca bu rolü, rol id'si ile döndürür.

```
curl -X POST "https://<api-host>/v1/SRVGetRoles" \
 -d "api_key=<api_key>"
```

SRVGetCompanyRoleData

Alan	Zorunlu	Notlar
`api_key`	**Zorunlu**	Bkz. Kimlik doğrulama.

```
curl -X POST "https://<api-host>/v1/SRVGetCompanyRoleData" \
 -d "api_key=<api_key>"
```

5. Rol nesnesi

`SRVGetRoles` ve `SRVGetCompanyRoleData` tarafından `roles` içinde, `SRVGetUserRoles` tarafından ise her çalışanın içinde `role` olarak döner.

```
{
 "id": 5,
 "name": "Supervisor",
 "hierarchy_level": 2,
 "hierarchy_name": "Supervisor",
 "is_global": false,
 "active": true,
 "user_count": 4,
 "permissions": {
   "work-orders": {"view": true, "create": true, "edit": true, "delete": false},
   "users-list":  {"view": false, "create": false, "edit": false, "delete": false}
 }
}
```

Alan	Açıklama
`id`	Rol id'si. Çalışanlar bu role `role_id` ile bağlanır.
`name`	Portalda tanımlanan rol adı.

`hierarchy_level`	0 ile 3 arası, bkz. bölüm 2.
`hierarchy_name`	Seviyenin adı: Super Admin, Admin, Supervisor veya User.
`is_global`	Yalnızca her şirkette bulunan yerleşik "Admin" rolü için `true`.
`active`	Rol portalda pasife alınmışsa `false`.
`user_count`	Bu role sahip aktif çalışan sayısı. `SRVGetUserRoles` tarafından döndürülmez.
`permissions`	`sections` içinde listelenen her modül için, modül anahtarıyla bir kayıt. Her kayıtta `true` veya `false` olarak `view` (görüntüleme), `create` (oluşturma), `edit` (düzenleme) ve `delete` (silme) bulunur. Rolün erişimi olmayan bir modül de listelenir; dördü de `false` olur.

`SRVGetRoles` ve `SRVGetCompanyRoleData` ayrıca şunları döndürür:

Alan	Açıklama
`hierarchy_levels`	Bölüm 2'deki tablo: `level`, `name` ve `scope` (`all`, `team` veya `self`).
`sections`	Tüm portal modülleri: `key` (`permissions` içinde kullanılır) ve `name`.

Modülün adını değil, `key` değerini kullanın. Anahtar değişmez; ad değişebilir.

6. Çalışan nesnesi

`SRVGetCompanyRoleData` tarafından `users` içinde, `SRVGetUserRoles` tarafından `data` içinde döner.

```
{
 "user_id": 12,
 "mopuan_user_id": "345",
 "name": "Jane Doe",
 "role_id": 5,
 "only_assigned": false,
 "supervisors": [{"user_id": 3, "mopuan_user_id": "101"}],
 "direct_reports": [
   {"user_id": 20, "mopuan_user_id": "410"},
   {"user_id": 21, "mopuan_user_id": null}
 ],
 "visibility": {
   "scope": "team",
   "user_ids": [12, 20, 21, 30],
   "mopuan_user_ids": ["345", "410", "512"]
 }
}
```

Alan	Açıklama
`user_id`	OpSense kullanıcı id'si.

`mopuan_user_id`	Sizin çalışan id'niz; çalışan henüz Mopuan'a bağlanmamışsa `null`. Bu çalışanlar da hiyerarşide boşluk kalmaması için listelenir.
`name`	Ad ve soyad.
`role_id`	Çalışanın rolü (yalnızca `SRVGetCompanyRoleData`).
`role`	Bölüm 5'teki rol nesnesinin tamamı (yalnızca `SRVGetUserRoles`, `role_id` yerine).
`only_assigned`	Portal bu çalışanı yalnızca kendi verileriyle sınırlıyorsa `true`.
`supervisors`	Çalışanın bağlı olduğu yöneticiler. Boş olabilir veya birden fazla kayıt içerebilir. Kendi yöneticisi olan bir supervisor'ın da kendi `supervisors` listesi vardır.
`direct_reports`	Bu çalışana doğrudan bağlı çalışanlar. Boş olabilir.
`visibility.scope`	`all`: şirketin tüm çalışanlarını görür. `team`: `user_ids` içindeki çalışanları görür. `self`: yalnızca kendisini görür.
`visibility.user_ids`	Çalışanın görebildiği OpSense id'leri, kendisi dahil. `scope` değeri `all` ise gönderilmez.
`visibility.mopuan_user_ids`	Aynı liste, sizin çalışan id'lerinizle. Mopuan'a bağlı olmayan çalışanlar listede yer almaz. `scope` değeri `all` ise gönderilmez.

Yalnızca aktif çalışanlar döner. Pasif veya silinmiş çalışanlara giden yönetici ve ekip bağlantıları listelenmez.

Görünürlük kuralları portalın kullandığı kurallarla aynıdır ve bu sırayla uygulanır:

1. `only_assigned` `true` ise: `self`.
2. Seviye 0 veya 1: `all`.
3. Seviye 2: `team`. Çalışanın kendisi, doğrudan bağlı çalışanları ve onlara doğrudan bağlı çalışanlar. Daha alt seviyeler dahil değildir.
4. Seviye 3: `self`. Kendisine bağlı biri olan seviye 3 çalışan için de geçerlidir.

7. Hata kodları

error_code	Anlamı	Tekrar denenmeli mi?
10015	`api_key` eksik.	Hayır. İsteği düzeltin.
10016	`api_key` geçersiz veya pasif.	Hayır. Anahtarı kontrol edin.
10020	`SRVGetRoles`: `reference_id` bu şirketin rolü değil. `SRVGetUserRoles`: çalışan bu şirketin aktif çalışanı değil.	Hayır.
10099	İstek bizim tarafımızda tamamlanamadı.	Evet

```
{"result": {"error_code": "10020", "error_text": "User not found."}}
```

8. Ne zaman çağrılmalı

- **Çalışan Mopuan'a giriş yaptığında:** `mopuan_user_id` ile `SRVGetUserRoles` çağırın. Tek çağrı, çalışanın yetkilerini ve kimlerin verilerini görebileceğini verir.
- **Tam senkronizasyon:** `SRVGetCompanyRoleData` servisini günde bir kez veya çalışan senkronizasyonundan sonra çağırın. Önceden sakladığınız veriyi yeni yanıtla değiştirin.
- **Rol ayar ekranları:** `SRVGetRoles` çağırın.

Roller, yetkiler ve yöneticiler OpSense portalında değiştirilir. Bu servisler değişiklik olduğunu size bildirmez; bu yüzden bir kopyayı bir günden uzun süre saklamayın.