



## Tenant Partitioning

---

- Understanding Tenant, on page 1
- Tenant Provisioning using APIs, on page 1
- Working with Tenant Partitioning, on page 2
- Modifying the Objects after Tenant Creation, on page 3

## Understanding Tenant

Each tenant has its own set of partition, schedule set, schedule detail, search space, search space member, phone system, class of service, user template, distribution list, distribution list membership, user operator, call handler template, directory handler, interview handler, call handlers (Operator, Opening Greeting, Goodbye), and routing rules. The configuration objects are shared across multiple tenants such as, all the system settings like mailbox store, custom keypad, restriction tables etc. are shared across the multiple tenants.

Tenant partitioning also introduces the concept of using the corporate email addresses as their alias, which enables alias uniqueness across tenants. To achieve this, the aliases assigned to the users (in multi-tenancy environment) must be in the form of alias@smtpdomain, which would be the corporate email address of the tenant's user. To login into the Unity Connection Web interfaces like Cisco PCA, Web Inbox, VMREST API, users have to use this alias format. Two different tenants configured on the same Unity Connection can never have the same alias for the users as the smtp domain always differ.



---

**Note** There are no changes done for touchtone conversation and voice-recognition users.

---

## Tenant Provisioning using APIs

Cisco Unity Connection Provisioning Interface (CUPI) provides set of APIs for Unity Connection that can be used for tenant provisioning. To support the Tenant Partitioning feature using APIs in Unity Connection, few new APIs have been developed and few existing APIs have been modified.

- For more information about provisioning APIs, see the documentation wiki: [http://docwiki.cisco.com/wiki/Cisco\\_Unity\\_Connection\\_Provisioning\\_Interface\\_%28CUPI%29\\_API](http://docwiki.cisco.com/wiki/Cisco_Unity_Connection_Provisioning_Interface_%28CUPI%29_API)

# Working with Tenant Partitioning

## Steps to List the Basic Configuration Settings to Provision a Tenant

### Step 1

Create Tenant: A new API has been introduced for tenant provisioning. All its associated objects are created by executing a single API request. If the operation fails at any point in time, the entire process is rolled back. To create a tenant, an HTTP POST request should be sent to the URI: `https://<connection-server>/vmrest/tenants`

For the optional parameters, if no value is specified then system default values be used.

### Step 2

Create Port Group: To create a port group, an HTTP POST request should be sent to the URI:

`https://<connection-server>/vmrest/portgroups`

Tenant partitioning supports SIP integration only.

### Step 3

Create Ports: To create a port, an HTTP POST request should be sent to the URI:

`https://<connection-server>/vmrest/ports`

### Step 4

Integrate with Call Manager: To integrate with Call Manager, do the following:

- Create SIP Security Profile
- Create SIP Profile
- Create Trunk to Publisher and Subscriber
- Create Route Group
- Create Route List
- Create Route Pattern
- Create Voicemail Pilot and Voicemail Profile

### Step 5

Create User for the Tenant: To create a user, an HTTP POST request should be sent to the URI:

`https://<connection-server>/vmrest/users?templateAlias=<TenantUserTemplate>`

User template specified in the URI should belong to the tenant for which you are creating the user.

### Step 6

Record Greetings: To record the greetings, do the following:

- Create a temporary place-holder WAV file on Unity Connection via POST operation on the URI:  
`https://<connection-server>/vmrest/voicefiles`
- Use the temporary file name created before to upload the new audio. The HTTP content type is "audio/wav" and the payload content is the audio data.  
PUT `https://<connection-server>/vmrest/voicefiles/<TemporaryFileName>`
- Assign temporary audio WAV file to the target resource. Perform a POST operation on the following URI:  
`https://<connection-server>/vmrest/handlers/callhandlers/<TenantCallHandlerObjectId>/greetings/Standard/greetingstreamfiles/1033`

### Step 7

Call Pilot Number: Dial the pilot number. You hear the recorded opening greeting.

For more information about provisioning APIs, see the documentation wiki: [http://docwiki.cisco.com/wiki/Cisco\\_Unity\\_Connection\\_Provisioning\\_Interface\\_%28CUPI%29\\_API](http://docwiki.cisco.com/wiki/Cisco_Unity_Connection_Provisioning_Interface_%28CUPI%29_API).

## Phone System Integration

For multiple phone systems integration with Unity Connection, see the Multiple Phone System Integration Guide for Cisco Unity Connection Release 11.x available at [https://www.cisco.com/c/en/us/td/docs/voice\\_ip\\_comm/connection/11x/integration/guide/multiple\\_integration/b\\_cuc11xintmultiple.html](https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/connection/11x/integration/guide/multiple_integration/b_cuc11xintmultiple.html).

---

# Modifying the Objects after Tenant Creation

The administrator can modify the objects associated with a tenant after its creation.

### Steps to Add a New Routing Rule for a Tenant

#### Before you begin

---

**Step 1** Create a new routing rule, by sending the below POST request:

POST https://<connection-server>/vmrest/routingrules

Set the search space for the routing rule as per the tenant's search space for which this routing rule is getting created.

**Step 2** Create a new routing rule condition, by sending the below POST request:

POST https://<connection-server>/vmrest/routingrules/<routingrulesobjectId>/routingruleconditions

Add the tenant's phone system as the routing rule condition.

---

## Modifying the Objects after Tenant Creation