# Advosecure23 API list

Below is the list of required Advosecure API’s set for both Management as Tenant portal

**Management Portal**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Context** | **Action** | **Function** | **Parameters** | **result** | **Note** |
| Tenants | GET | /api/Tenants/List/{parentId} | parentId = UTID of management tenant  parentID = 0 or empty = rootparent | Child Collection{utid, parentId, tenantname, domain, contact, address, status} | Returns all child tenants.  If parentID=0 then return all child tenants of rootparent |
| Tenants | GET | /api/Tenants/Get/{utid} | Utid = UniquetenantId | utid, authprovider, domain, connectionstring, tenantname, contact, address, status | Returns 1 tenant detail |
| Tenants | POST | /api/Tenants/Add | {  "id": 0,  "schemaName": "string",  "name": "string",  "azureTenantId": "string",  "connectionString": "string",  "adminTenantId": 0 } | Error or UTID | Create a tenant |
| Tenants | GET | /api/Tenants/Exist/{utid, name} | Utid or name | True or False | Check if a tenant  exists |
| Tenants | POST | /api/Tenants/Enable/{datetime} | Datetime = when to enable if empty then immediate. Must be empty or in future | True = enabled success or False not success, reasoncode | Enable a disable tenant |
| Tenants | POST | /api/Tenants/Disable/{datetime},{reason} | Datetime = when to enable if empty then immediate. Must be empty or in future. Reason (0=new, 1=end of subscription, 2=cancelled by us, 3=other | True = enabled success or False not success, reasoncode | Disable a tenant |
| Tenants | POST | /api/Tenants/Subscribe |  |  | Used by subscribe form |
| Users | GET | /api/Users/List/{UTID} | UTID=Unique Tenant ID | Uuid, username, email, phone, status, UTID | Get list of users for management portal for current user. These are only management users, not tenant users. |
| Users | GET | /api/Users/Get/{UUID} | UUID=Unique User ID | User data | Get specific user |
| Users | POST | /api/Users/Add |  |  |  |
| Users | POST | /api/Users/Update/{UUID} |  |  |  |
| Users | POST | /api/Users/Enable/{UUID} |  |  |  |
| Users | POST | /api/Users/Disable/{UUID} |  |  |  |
| Users | POST | /api/Users/AddRole/{RID,UUID} | RoleID, UUID |  |  |
| Users | POST | /api/Users/RemoveRole/{RID,UUID} |  |  |  |
| Users | POST | /api/Users/ResetPW/{UUID} | UUID | True or False | Reset user password and MFA. Will send confirmation mail to user with password reset link. |
| Rbac | GET | /api/rbac/GetRoles |  |  |  |
| Rbac | POST | /api/rbac/AddRole | RoleName | RoleID, or 0 if failed | Add a role if not exist |
| Rbac | POST | /api/rbac/RemoveRole/{Rid} | RoleId | True -1, or False 0 | Will fail if Role has members |
| Rbac | GET | /api/rbac |  |  |  |
| Pages | GET | /api/pages/get/{contentid} | contentid | Content (html) from Content table | Get HTML content for page content to be included in html page layout. |
| Pages | POST | /api/pages/update/{contentid, content | Contentid, content | Result (false = failed, true is ok) | Update HTML content |
| Pages | POST | /api/pages/delete/{contentid} | Contentid | Result true or false | Delete page content |

**Tenant Portal**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Context | Action | Function | Parameters | result | Note |
| Tenant | GET | /api/Tenants/GetCurrent | None | Get Current tenant data (utid, name, contact, etc) | Returns current tenant data |
| Tenant | GET | /api/Tenant/GetSettings | Get settings |  | Get tenant settings. On tenant creation these are prefilled with data from MGMT portal |
| Tenant | GET | /api/Tenant/SetSettings | Updates settings |  |  |
| Users | GET | /api/Users/List/{UTID} | UTID=Unique Tenant ID | Uuid, username, email, phone, status, UTID | Get list of users for app portal. |
| Users | GET | /api/Users/Get/{UUID} | UUID=Unique User ID | User data | Get specific user |
| Users | POST | /api/Users/Add |  |  |  |
| Users | POST | /api/Users/Update/{UUID} |  |  |  |
| Users | POST | /api/Users/Enable/{UUID} |  |  |  |
| Users | POST | /api/Users/Disable/{UUID} |  |  |  |
| Users | POST | /api/Users/AddRole/{RID,UUID} | RoleID, UUID |  |  |
| Users | POST | /api/Users/RemoveRole/{RID,UUID} |  |  |  |
| Users | POST | /api/Users/ResetPW/{UUID} | UUID | True or False | Reset user password and MFA. Will send confirmation mail to user with password reset link. |
| Users | POST | /api/Users/UpdateProfile |  |  |  |
| Users | GET | /api/Users/GetProfile |  |  |  |
| Users | POST | /api/Users/GetWidgets/{uuid} | Uuid | Array of widgets with data/config | Gets all user defined widgets with dasboard config (location, size etc), user saved settings,. As input for FE dashboard generation |
| Contacts | GET | /api/Contacts/List/ | none | Collection{ucid, contacttype, contactname, parentId, address, status} | Returns all contacts |
| Contacts | GET | /api/Contacts/Get/{ucid} | Ucid = UniquecontactId | All columns | Returns 1 contact |
|  |  |  |  |  |  |
| Contacts | POST | /api/Contacts/Add | {  "id": 0,  "schemaName": "string",  "name": "string",  "azureTenantId": "string",  "connectionString": "string",  "adminTenantId": 0 } | Error or UTID | Create a contact |
| Contacts | POST | /api/Contacts/Delete/{ucid} | ucid |  | Soft deletes a contact record and clears contact data |
| Contacts | POST | /api/Contacts/Update/{ucid} | ucid |  | Updates a Contact record |
| Contacts | GET | /api/contacts/GetSData/{ucid} | Ucid | 0 not found, -1 ok, 1 not authorized | Get contact sensitive data |
| Contacts | POST | /api/contacts/SetSData/{ucid} | Ucid + data | 0 not found, -1 ok, 1 not authorized | Set contact sensitive data |
| Contacts | POST | /api/contacts/RemoveSData/{ucid} | Ucid | 0 not found, -1 ok, 1 not authorized | Removes contact sensitive data |
| Contacts | GET | /api/contacts/GetUser/{ucid} | Ucid | Uuid (linked Userid if exist) | Get linked user id if it exists |
| Contacts | POST | /api/contacts/ |  |  |  |
| Contacts |  | Contact\_civilstatus |  |  |  |
| Contacts |  | Contact\_idtype |  |  |  |
| Contacts |  | Contact\_onlinestatus |  |  |  |
| Contacts |  | Contact\_titles |  |  |  |
| Matter |  | Matter\_area |  |  |  |
| Matter |  | matter |  |  |  |
| Matter |  | Matter\_contacts |  |  |  |
| Matter |  | Matter\_type |  |  |  |
| Matter |  | Matter\_notes |  |  | Links notes to matters |
| Countries |  | Countries |  |  |  |
| Courts |  | Court\_geographical\_jurisdiction |  |  |  |
| Courts |  | Court\_sitting\_in\_city |  |  |  |
| Courts |  | Court\_type |  |  |  |
| tasks |  | CRUD |  |  |  |
| tasks |  | templates |  |  |  |
| tasks |  | time |  |  |  |
| Tasks |  | Type (get) |  |  | Task types |
| Time |  | CRUD |  |  | Spent time on something |
| Time |  | Category |  |  | Like meeting, travel, admin,investigation |
| Assets |  |  |  |  | documents |
| Phone |  |  |  |  |  |
| Profile |  |  |  |  |  |
| Roles |  |  |  |  |  |
| Sessions |  |  |  |  |  |
| Activities |  |  |  |  |  |
| Assets |  |  |  |  |  |
| Billing |  |  |  |  |  |
| Expense |  |  |  |  |  |
| Fee |  |  |  |  |  |
| invoice |  |  |  |  |  |
| languages |  |  |  |  |  |
| lead |  |  |  |  |  |
| Note |  | Note\_Task |  |  | Links tasks to notes |
| Note |  | Note\_notification |  |  | Links Notifications to notes |
| Note |  | Note\_Matter |  |  | Links matters to notes |
| Note |  | Note |  |  | Holds notes |
| Crm |  | Leads (source, sourcetype, status) |  |  |  |
| Crm |  | Opportunities (contact, stage) |  |  |  |
| Voip |  | Settings |  |  |  |
| Voip |  | Extensions (list,get,set,delete,status) | Extension, uuid, password, status |  |  |
| Widgets | Get | Get |  |  | Available dashboard widgets |
| Voip | POST | Event (type,ext,direction, source, dest) |  |  | Receives phone/voip event. Will manage status of employee and triggers update contact, matter event |