HCL MT CH-MSP Product Documentation

Provisioning APIs

Manually fixing Blogs provisioning API issue

Provisioning APIs

API - https://apps.na.collabserv.com/wikis/home?lang=en-us#!/wiki/W0d07dd0b225e_410e_a5c4_1b9cfc43101d/page/Provisioning%20APIs

I. General information

When calling the provisioning APIs, use the credential for users with the 'bss-provisioning-admin' role defined in each Connections app.

All provisioning APIs have two phases to call:

- **prepare**: A prepare phase to check with the application whether it is ok to make the API call. No data are updated in the applications.
- **execute**: Once prepare calls return with an OK, go ahead make the actual call to save/update the provisioning data.

Although 'prepare' phase is optional, it is recommended to add it as part of the general provisioning logic.

II. Manage Organizations

The API allows admin users to create, and update organizations that are already defined in LDAP to Connections applications. The organization information, e.g. organization name, come from the LDAP. For instance, the organization name comes from the LDAP, not provided in the API payload.

Input:

Method	URI	Description
POST	/profiles/wdp/provisioning/profilesendpointmtprovisioning	Managing organizations.

Headers:

Header name	Value	Description
Content-type	application/json	Content-type header can only be set to application/json
x-message-id	Prepare phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4eprofiles/wdp/provisioning/profilesendpointmtprovisioningprepare Execute phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4eprofiles/wdp/provisioning/profilesendpointmtprovisioningexecute	This is an optional header to carry a message-id to the server. The general format of the message ID is: <UUID>profiles/wdp/provisioning/profilesendpointmtprovisioning<phase> where <UUID> is a randomly generate ID; <phase> is either prepare or execute

1. Samples payloads

a). Sample JSON input for: prepare, and add an organization:

```
{
  "Phase": "prepare",
  "Version": "1",
  "OperationId": "AddOrganization",
  "ReqId": "ff97bd83-c23b-43af-a8ae-769496686bda",
  "Payload": {
    "OrganizationId": "4000000004"
  },
  "RequestedBy": "",
  "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}
```

b). Sample JSON input for adding organization in prepare phase:

```
{
  "Phase": "execute",
  "Version": "1",
  "OperationId": "AddOrganization",
  "ReqId": "ff97bd83-c23b-43af-a8ae-769496686bda",
  "Payload": {
    "OrganizationId": "4000000004"
  },
  "RequestedBy": "",
  "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}
```

c). Sample JSON input for updating organization in prepare phase:

```
{
  "Phase": "prepare",
  "Version": "1",
  "OperationId": "AddOrganization",
  "ReqId": "ff97bd83-c23b-43af-a8ae-769496686bda",
  "Payload": {
    "OrganizationId": "4000000004"
  },
  "RequestedBy": "",
  "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}
```

d). Sample JSON input for updating organization in execute phase:

```
{
  "Phase": "execute",
  "Version": "1",
  "OperationId": "UpdateOrganization",
  "ReqId": "ff97bd83-c23b-43af-a8ae-769496686bda",
  "Payload": {
    "OrganizationId": "4000000004"
  },
  "RequestedBy": "",
  "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}
```

2. Input field description:

Field Name	Required	Description
Phase	Yes	Two choices: • For prepare phase - set the value to: prepare. • For execute phase - set the value to: execute
OperationId	Yes	Two choices: • For adding a new organization, set OperationId to: AddOrganization • For updating an organization, set OperationId to: UpdateOrganization
ReqId	No	Optional. But the ReqId can be used to link the prepare phase call and the execute call (described later). It is typically a randomly generated UUID.
Payload	Yes	Only one field is required: OrganizationId. This is the ID that is defined in LDAP for the organization you are trying to provision.

RequestedBy	No	Optional. You can set it to the user who is performing the API call, or leave it blank.
ServiceId	No	Always set it to this value for provisioning organzations: profiles/wdp/provisioning/profilesendpointmtprovisioning

3. Command line sample:

curl --insecure -u wasadmin:xbhyimln -d @entitle_org_execute.json -H "Content-Type:application/json" -X POST -v https://lcauto4.cnx.cwp.pnp-hcl.com/profiles/wdp/provisioning/profilesendpointmtprovisioning

II. Manage Users

Users need to be managed in each Connections application individually, using an API endpoint in each application. So to add/update/revoke a user, all endpoints to the following applications need to be called: Profiles, Files, Communities, News, Wikis, Activities, Blogs. Note that 'Bookmark/Dogear' is not supported for multi-tenant.

Note that the API calls are all very similar in each application, except the API endpoints.

For all input JSON for managing users,, there is an 'OperationId' field, set the value in the payload JSON accordingly:

OperationId	**Description**
AddSubscriber	Add a new user to a Connections application.
UpdateSubscriber	Update a user in a Connections application.
RevokeSubscriber	Revoke a user in a Connections application. **Note:** When a user is revoked, the user will be marked as inactive in Connections. Provisioning APIs do not update LDAP. So such user will still be able to login, and access Connections. In order to prevent users from logging in and access Connections, their LDAP account needs to be disabled.

1. Manage users in Profiles

Input:

Method	**URI**	**Description**
POST	/profiles/wdp/provisioning/profilesendpointmtprovisioning	Manage users for Profiles

Headers:

Header Name	**Value**	**Description**
Content-type	application/json	
	Prepare phase: 5773004f-8f7b-40d5-8b1c-	This is an optional header to carry a message-id to the server.

x-message-id	831abcc81d4eprofiles/wdp/provisioning/profilesendpointmtprovisioningprepare Execute phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4eprofiles/wdp/provisioning/profilesendpointmtprovisioningexecute	The general format of the message ID is: <UUID>profiles/wdp/provisioning/profilesendpointmtprovisioning<phase> where <UUID> is a randomly generate ID; <phase> is either prepare or execute

Note that API endpoint to manger users in Profiles is exactly the same as the API endpoint to manage Organizations. That is because organizations are only managed and used in Profiles, not other Connections applications.

a). Add a user to Profiles:

```
{
 "Phase": "prepare",
 "Version": "1",
 "OperationId": "AddSubscriber",
 "ReqId": "87d80a71-13a6-4846-942f-e36ee24fda38",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {},
  "SubscriberId": "1000000006"
 },
 "RequestedBy": "",
 "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}
```

execute phase body:

```
{
 "Phase": "execute",
 "Version": "1",
 "OperationId": "AddSubscriber",
 "ReqId": "87d80a71-13a6-4846-942f-e36ee24fda38",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {},
  "SubscriberId": "1000000006"
 },
 "RequestedBy": "",
 "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}
```

b). Revoke a user in Profiles:

```
{
 "Phase": "prepare",
 "Version": "1",
 "OperationId": "RevokeSubscriber",
 "ReqId": "0780b166-28e5-4e8e-96b7-8a0d5f04d7d2",
 "Payload": {
  "SubscriberId": "1000000006"
 },
 "RequestedBy": "",
```

```
  "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}

{
 "Phase": "execute",
 "Version": "1",
 "OperationId": "RevokeSubscriber",
 "ReqId": "0780b166-28e5-4e8e-96b7-8a0d5f04d7d2",
 "Payload": {
  "SubscriberId": "1000000006"
 },
 "RequestedBy": "",
 "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}
```


c). Update a user in Profiles:


```
{
 "Phase": "execute",
 "Version": "1",
 "OperationId": "UpdateSubscriber",
 "ReqId": "87d80a71-13a6-4846-942f-e36ee24fda38",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {},
  "SubscriberId": "1000000006"
 },
 "RequestedBy": "",
 "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}

{
 "Phase": "prepare",
 "Version": "1",
 "OperationId": "UpdateSubscriber",
 "ReqId": "87d80a71-13a6-4846-942f-e36ee24fda38",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {},
  "SubscriberId": "1000000006"
 },
 "RequestedBy": "",
 "ServiceId": "profiles/wdp/provisioning/profilesendpointmtprovisioning"
}
```


b). Sample cURL command

curl --insecure -u wasadmin:xxxxxx -d @entitle_user_profiles.json -H "Content-Type:application/json" -X POST -v https://lcauto4.cnx.cwp.pnp-hcl.com/profiles/wdp/provisioning/profilesendpointmtprovisioning

2. Manage users in Files

Input:

Method	URI	Description
POST	/files/wdp/provisioning/filesendpointmtprovisioning	Manage users for Files

Headers:

Header Name	Value	Description
Content-type	application/json	
x-message-id	Prepare phase: 95ac8887-6c38-4044-87c7-b416dd850e47files/wdp/provisioning/filesendpointmtprovisioningprepare Execute phase: 95ac8887-6c38-4044-87c7-b416dd850e47files/wdp/provisioning/filesendpointmtprovisioningexecute	This is an optional header to carry a message-id to the server. The general format of the message ID is: <UUID>profiles/wdp/provisioning/filesendpointmtprovisioning<phase> where <UUID> is a randomly generate ID; <phase> is either prepare or execute

a). Add a user to Files:


```
{
 "Phase": "prepare",
 "Version": "1",
 "OperationId": "AddSubscriber",
 "ReqId": "5ca4b324-bb32-4dd6-b09d-0f7741034cb4",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {
  "FilesService": {
     "files_allow_quota_overage": "true",
     "files_quota": "524288000"
   },
    "ShareService": {
     "share_allow_quota_overage": "true",
     "share_quota": "524288000"
    }
  },
  "SubscriberId": "1000000029"
 },
 "RequestedBy": "",
 "ServiceId": "files/wdp/provisioning/filesendpointmtprovisioning"
}

{
 "Phase": "execute",
 "Version": "1",
 "OperationId": "AddSubscriber",
 "ReqId": "5ca4b324-bb32-4dd6-b09d-0f7741034cb4",
 "Payload": {
```

```
   "Locale": "en_US",
   "ServiceOfferingAttributeValues": {
   "FilesService": {
      "files_allow_quota_overage": "true",
      "files_quota": "524288000"
    },
    "ShareService": {
      "share_allow_quota_overage": "true",
      "share_quota": "524288000"
    }
   },
   "SubscriberId": "1000000029"
 },
 "RequestedBy": "",
 "ServiceId": "files/wdp/provisioning/filesendpointmtprovisioning"
}
```


Note: Both '*files_quota*' and '*share_quota*' attributes are quota limits for the entire organization, in MB. Please set them to the same value for all users in the same organization.

b). Revoke a user in Files:


```
{
 "Phase": "prepare",
 "Version": "1",
 "OperationId": "RevokeSubscriber",
 "ReqId": "45e77448-3055-4457-b9f8-dd46b0a8681d",
 "Payload": {
  "SubscriberId": "1000000004"
 },
 "RequestedBy": "",
 "ServiceId": "files/wdp/provisioning/filesendpointmtprovisioning"
}

{
 "Phase": "execute",
 "Version": "1",
 "OperationId": "RevokeSubscriber",
 "ReqId": "45e77448-3055-4457-b9f8-dd46b0a8681d",
 "Payload": {
  "SubscriberId": "1000000004"
 },
 "RequestedBy": "",
 "ServiceId": "files/wdp/provisioning/filesendpointmtprovisioning"
}
```


Note: Both '*files_quota*' and '*share_quota*' attributes are quota limits for the entire organization, in MB. Please set them to the same value for all users in the same organization.

c). Update a user in Files:


```
{
 "Phase": "prepare",
 "Version": "1",
 "OperationId": "UpdateSubscriber",
 "ReqId": "5ca4b324-bb32-4dd6-b09d-0f7741034cb4",
 "Payload": {
  "Locale": "en_US",
```

```
  "ServiceOfferingAttributeValues": {
   "FilesService": {
   "files_allow_quota_overage": "true",
   "files_quota": "524288000"
   },
   "ShareService": {
    "share_allow_quota_overage": "true",
    "share_quota": "524288000"
   }
  },
  "SubscriberId": "1000000029"
 },
 "RequestedBy": "",
 "ServiceId": "files/wdp/provisioning/filesendpointmtprovisioning"
}
```

```
{
  "Phase": "execute",
  "Version": "1",
  "OperationId": "UpdateSubscriber",
  "ReqId": "5ca4b324-bb32-4dd6-b09d-0f7741034cb4",
  "Payload": {
    "Locale": "en_US",
    "ServiceOfferingAttributeValues": {
      "FilesService": {
        "files_allow_quota_overage": "true",
        "files_quota": "524288000"
      },
      "ShareService": {
        "share_allow_quota_overage": "true",
        "share_quota": "524288000"
       }
    },
    "SubscriberId": "1000000029"
  },
  "RequestedBy": "",
  "ServiceId": "files/wdp/provisioning/filesendpointmtprovisioning"
}
```


Note: Both '*files_quota*' and '*share_quota*' attributes are quota limits for the entire organization, in MB. Please set them to the same value for all users in the same organization.

b). Sample cURL command

```
curl --insecure -u wasadmin:xbhyimln -d @entitle_user_files.json -H "Content-Type:application/json" -X POST -v https://lcauto4.cnx.cwp.pnp-hcl.com/files/wdp/provisioning/filesendpointmtprovisioning
```

3. Manage users in Communities

Input:

Method	URI	Description
POST	/communities/wdp/provisioning/communitiesendpointmtprovisioning	Managing users for Communities.

Headers:

Header Name	Value	Description

Content-type	application/json	
x-message-id	Prepare phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4ecommunities/wdp/provisioning/communitiesendpointmtprovisioningprepare Execute phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4ecommunities/wdp/provisioning/communitiesendpointmtprovisioningexecute	This is an optional header to carry a message-id to the server. The general format of the message ID is: <UUID>communities/wdp/provisioning/communitiesendpointmtprovisioning<phase> where <UUID> is a randomly generate ID; <phase> is either prepare or execute

Sample payload:


```
{
 "Phase": "prepare",
 "Version": "1",
 "OperationId": "AddSubscriber",
 "ReqId": "3d186157-e03c-4406-b7ca-521e6c159541",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {},
  "SubscriberId": "1000000001"
 },
 "RequestedBy": "",
 "ServiceId": "communities/wdp/provisioning/communitiesendpointmtprovisioning"
}
```


For other phase and operations, set the fields 'Phase' and 'OperationId' accordingly.

Sample cURL command

curl --insecure -u wasadmin:xbhyimln -d @entitle_user_communities.json -H "Content-Type:application/json" -X POST -v https://lcauto4.cnx.cwp.pnp-hcl.com/profiles/wdp/provisioning/communities/wdp/provisioning/communitiesendpointmtprovisioning

4. Manage users in News

Input:

Method	URI	Description
POST	/news/wdp/provisioning/newsendpointmtprovisioning	Managing users for News/Homepage

Headers:

Header Name	Value	Description

Content-type	application/json	
x-message-id	Prepare phase: 6aa858bd-fbcc-4351-a6e6-0efeb6ac7151news/wdp/provisioning/newsendpointmtprovisioningexecute Execute phase: 6aa858bd-fbcc-4351-a6e6-0efeb6ac7151news/wdp/provisioning/newsendpointmtprovisioningexecute	This is an optional header to carry a message-id to the server. The general format of the message ID is: <UUID>news/wdp/provisioning/newsendpointmtprovisioning<phase> where <UUID> is a randomly generate ID; <phase> is either prepare or execute

Sample payload:


```
{
 "Phase": "execute",
 "Version": "1",
 "OperationId": "AddSubscriber",
 "ReqId": "cd075459-8e82-44fc-a690-0b4f08704f2f",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {},
  "SubscriberId": "1000000001"
 },
 "RequestedBy": "",
 "ServiceId": "news/wdp/provisioning/newsendpointmtprovisioning"
}
```


For other phase and operations, set the fields 'Phase' and 'OperationId' accordingly.

Sample cURL command

curl --insecure -u wasadmin:xbhyimln -d @entitle_user_profiles.json -H "Content-Type:application/json" -X POST -v https://lcauto4.cnx.cwp.pnp-hcl.com/profiles/wdp/provisioning/profilesendpointmtprovisioning

5. Manage users in Wikis

Input:

Method	URI	Description
POST	/wikis/wdp/provisioning/wikisendpointmtprovisioning	Managing users for Wikis.

Headers:

Header Name	Value	Description
Content-type	application/json	
x-message-id	Prepare phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4ewikis/wdp/provisioning/wikisendpointmtprovisioningprepare Execute phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4ewikis/wdp/provisioning/wikisendpointmtprovisioningexecute	This is an optional header to carry a message-id to the server. The general format of the message ID is: <UUID>wikis/wdp/provisioning/wikisendpointmtprovisioning<phase> where <UUID> is a randomly generate ID; <phase> is either prepare or execute

Sample payload:

```
{
 "Phase": "execute",
 "Version": "1",
 "OperationId": "AddSubscriber",
 "ReqId": "11b04405-18ac-47d0-9b48-3736b918501e",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {},
  "SubscriberId": "1000000001"
 },
 "RequestedBy": "",
 "ServiceId": "wikis/wdp/provisioning/wikisendpointmtprovisioning"
}
```

For other phase and operations, set the fields 'Phase' and 'OperationId' accordingly.

Sample cURL command

curl --insecure -u wasadmin:xbhyimln -d @entitle_user_activities.json -H "Content-Type:application/json" -X POST -v https://lcauto4.cnx.cwp.pnp-hcl.com/wikis/wdp/provisioning/wikisendpointmtprovisioning

6. Manage users in Activities

Input:

Method	URI	Description
POST	/activities/wdp/provisioning/activitiesendpointmtprovisioning	Managing users for Activities

Headers:

Header Name	Value	Description
Content-type	application/json	
x-message-id	Prepare phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4eactivities/wdp/provisioning/activitiesendpointmtprovisioningprepare Execute phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4eactivities/wdp/provisioning/activitiesendpointmtprovisioningexecute	This is an optional header to carry a message-id to the server. The general format of the message ID is: <UUID>activities/wdp/provisioning/activitiesendpointmtprovisioning<phase> where <UUID> is a randomly generate ID; <phase> is either prepare or execute

Sample payload:

```
{
 "Phase": "execute",
 "Version": "1",
 "OperationId": "AddSubscriber",
 "ReqId": "2e2da18a-3535-4d07-8347-f464b5093b21",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {},
  "SubscriberId": "1000000001"
 },
 "RequestedBy": "",
 "ServiceId": "activities/wdp/provisioning/activitiesendpointmtprovisioning"
}
```

For other phase and operations, set the fields 'Phase' and 'OperationId' accordingly.

Sample cURL command

curl --insecure -u wasadmin:xbhyimln -d @entitle_user_activities.json -H "Content-Type:application/json" -X POST -v https://lcauto4.cnx.cwp.pnp-hcl.com/activities/wdp/provisioning/activitiesendpointmtprovisioning

7. Manage users in Blogs

Input:

Method	URI	Description
POST	/blogs/wdp/provisioning/blogsendpointmtprovisioning	Managing users for Blogs

Headers:

Header Name	Value	Description
Content-type	application/json	
x-message-id	Prepare phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4eblogs/wdp/provisioning/blogsendpointmtprovisioningprepare Execute phase: 5773004f-8f7b-40d5-8b1c-831abcc81d4eblogs/wdp/provisioning/blogsendpointmtprovisioningexecute	This is an optional header to carry a message-id to the server. The general format of the message ID is: <UUID>blogs/wdp/provisioning/blogsendpointmtprovisioning<phase> where <UUID> is a randomly generate ID; <phase> is either prepare or execute

Sample payload:


```
{
 "Phase": "prepare",
 "Version": "1",
 "OperationId": "AddSubscriber",
 "ReqId": "476978dd-37ad-4eb8-89b7-bfdd6841292c",
 "Payload": {
  "Locale": "en_US",
  "ServiceOfferingAttributeValues": {},
  "SubscriberId": "1000000002"
 },
 "RequestedBy": "",
 "ServiceId": "blogs/wdp/provisioning/blogsendpointmtprovisioning"
}
```


For other phase and operations, set the fields 'Phase' and 'OperationId' accordingly.

b). Sample cURL command

curl --insecure -u wasadmin:xbhyimln -d @entitle_user_blogss.json -H "Content-Type:application/json" -X POST -v https://lcauto4.cnx.cwp.pnp-hcl.com/blogs/wdp/provisioning/blogsendpointmtprovisioning

Manually fixing Blogs provisioning API issue

It is very likely that you get 403 return code when calling Blogs provisioning APIs. It will be fixed in an iFix later (targeting July iFix). But you can follow the following steps to fix it manually.

Essentially, the web.xml needs to be updated in the Blogs enterprise application so that proper security filter can be applied for the provisioning APIs.

I. Use WAS Admin console

Thanks to Deniele, here is how to update the deployment using WAS admin console: Download the patch (thanks

to Daniele!): blogs-bssprovisioning-basicauth-patch.zip 1). Login to WAS Admin console, and check on Blogs

application checkbox;

2). Click 'Update' button and choose: Replace, add, or delete multiple files;

3). Specify the path to the patch zip file downloaded, and follow the screens to finish the update.

II. Manual update

Edit the web.xml in the following locations, and restart the server where Blogs runs on, it should work. a). Find the Blogs web.xml on

Dmgr node, e.g.:

```
/opt/IBM/WebSphere/AppServer/profiles/Dmgr01/config/cells/ip-10-190-161-
119Cell01/applications/Blogs.ear/deployments/Blogs/blogs.war/WEB-INF/web.xml
```

b). Find web.xml on all nodes where Blogs are deployed, e.g., i). At deployment:

```
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/config/cells/ip-10-190-161-
119Cell01/applications/Blogs.ear/deployments/Blogs/blogs.war/WEB-INF/web.xml
```

ii). At EAR:

```
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/installedApps/ip-10-190-161-
119Cell01/Blogs.ear/blogs.war/WEB-INF/web.xml
```

c). Edit web.xml as follows:

Find the following sections in web.xml. Delete the line as stroked out below, and replace it with the line in blue:

. . . . . .

```
  <filter-mapping>
    <filter-name>LCBasicAuthenticationFilter</filter-name>
    <servlet-name>BlogsMTProvisioningServlet</servlet-name>
    <url-pattern>/wdp/provisioning/*</url-pattern>
    <dispatcher>REQUEST</dispatcher>
    <dispatcher>FORWARD</dispatcher>
  </filter-mapping>

  ... ...


  <filter-mapping>
    <filter-name>BSSProvisioningSecurityFilter</filter-name>
    <servlet-name>BlogsMTProvisioningServlet</servlet-name>
    <url-pattern>/wdp/provisioning/*</url-pattern>
……
```