

Zowe API Integration

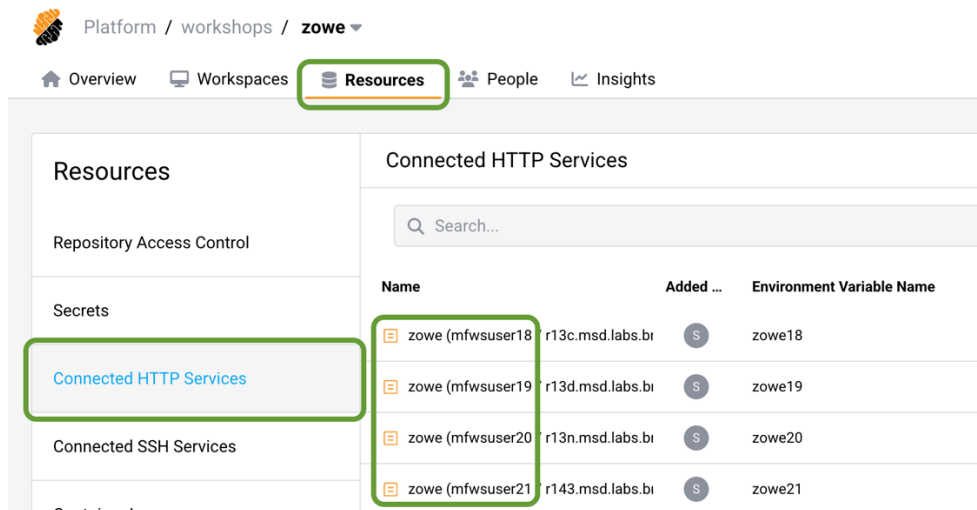
Getting Started

Imagine you're part of a team tasked with integrating the SYSVIEW application into Zowe's API Mediation Layer. Your goal is to make SYSVIEW's REST APIs available to external applications through Zowe's powerful API infrastructure.

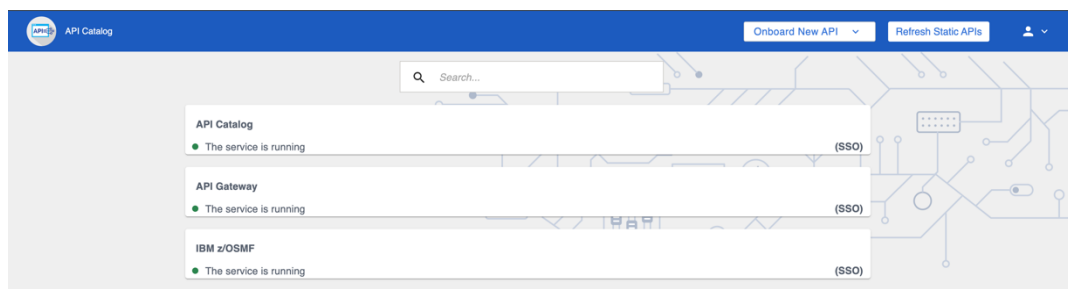
Browse the Zowe API Catalog UI

Before onboarding SYSVIEW, it's useful to review the services that are already integrated with Zowe APIML. This gives you an understanding of how existing services are listed and accessed in the API catalog.

1. Navigate to the Strong Network platform main page
2. Go to the **Resources** -> **Connected HTTP Services**
3. Click on the service corresponding to your user ID **zowe (mfwsuserXX...** to open the Zowe API Catalog UI



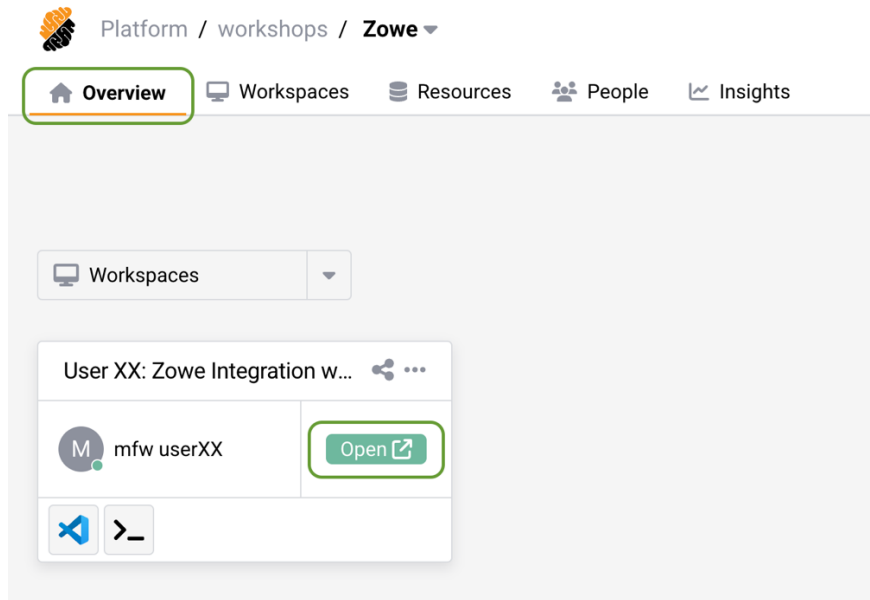
You can browse the list of currently onboarded services



Now that you've seen the catalog, it's time to switch to your development environment. You'll be working in a secure cloud IDE that runs VS Code, connected directly to the mainframe. This is where you'll do all your work. It has Zowe CLI, Zowe Explorer and TN3270 terminal emulator installed and configured.

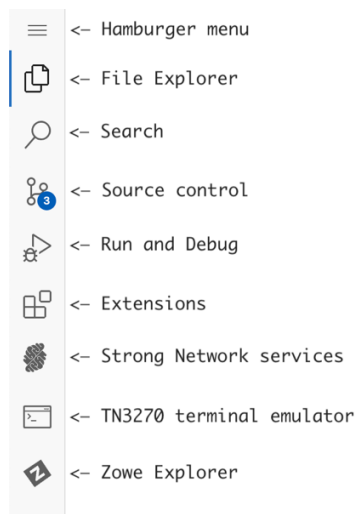
Open cloud IDE

1. Return to the Strong Network platform main page
2. Switch to the **Overview** tab
3. Click on *Running / Paused* button



The new window should pop-up, loading you into the secure cloud IDE.

Get familiar with the VSCode Activity Bar



With the IDE set up and ready, it's time to onboard the SYSVIEW REST API service to Zowe APIML. You'll use Zowe Explorer to locate configuration files and modify them to point to the correct host and IP. This will enable SYSVIEW's API to be onboarded to the mediation layer.

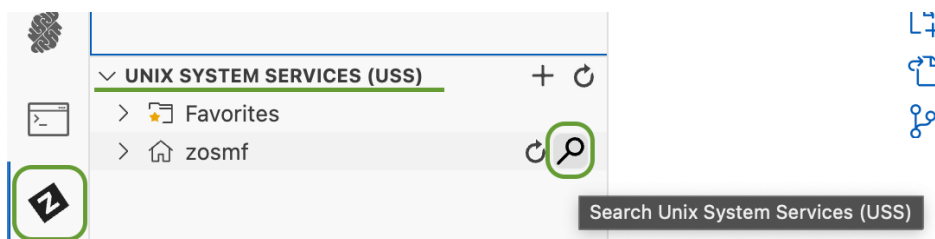
Onboard SYSVIEW to the APIML

Let's locate the SYSVAPPS (SYSVIEW Application Server) run-time directory using Zowe Explorer.

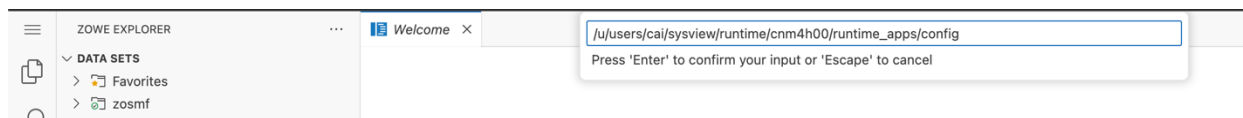
1. Copy path to the sysview configuration to the clipboard

```
/u/users/cai/sysview/runtime/cnm4h00/runtime_apps/config
```

2. Go to the *Zowe Explorer* extension by selecting it on the side panel.
3. Click the magnifying lens icon under USS section - zosmf profile.
Please note that first profile verification may take up to 20 seconds.



4. Once validated, paste the copied path to the SYSVAPPS configuration in the input field that appears:



5. Wait for the USS files to load, then click on the **application.yml** file to open it.
Note that the first file opening may take some time.
6. In the **application.yml** file, find the **apiml** section and update the dummy host names and IP addresses with your mainframe host and IP:

IP: **<zdnt-ip>**

Host: **<host>**

```
42  apiml:
43    enabled: true
44    service:
45      serviceId: casysview
46      hostname: <host.name.com>
47      ipAddress: <host.ip.address>
48      discoveryServiceUrls:
49        - https://<zowe.host.name>:7553/eureka
```

7. Save the file after making the necessary changes.

Start SYSVIEW Application Server

At this point, the changes to the configuration are made and SYSVIEW knows where to onboard. To see those changes take effect, you need to start the SYSVIEW Application Server.

1. Open an IDE terminal by using **Hamburger menu -> Terminal -> New Terminal**
2. Run the following Zowe CLI command to start the SYSVIEW Application Server:

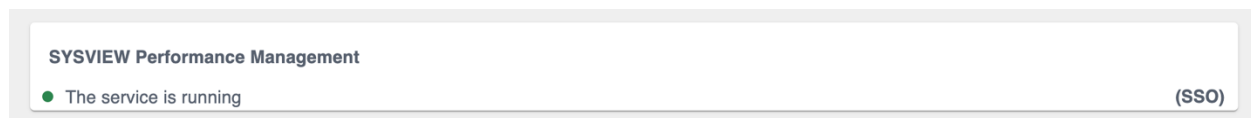
```
zowe zos-console issue command "S SYSVAPPS"
```

Verify onboarding

You've successfully updated the configuration and started the application server. Now, it's time to verify that SYSVIEW is properly onboarded and can be accessed via Zowe APIML.

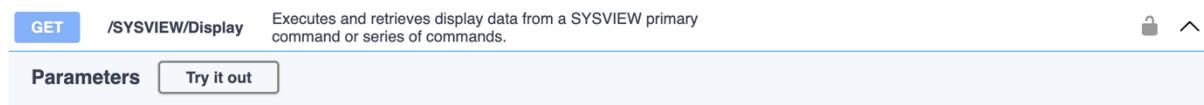
Browse the Zowe API Catalog UI once again

It may take 5 to 10 minutes for the SYSVAPPS to start and register to the Discovery Service. If the application was configured properly, we will see the **SYSVIEW** tile in the API Catalog.



Click on the tile to drilldown to the API documentation. You can verify the service is configured properly by trying out some endpoint.

Select endpoint, click **Try it out**, fill in parameters if needed and click **Execute** button below



Run the code snippet from IDE

Now you can reach the same endpoint from the cloud IDE using curl command. Navigate to the IDE terminal and paste following command there, replace host and pass values:

```
curl -X "GET" "https://  
<host>:7554/casysview/api/v1/SYSVIEW/Display?command=status" -H "accept:  
application/json" -u <tso_user_id>:<tso_user_password>
```

You successfully onboarded SYSVIEW to the Zowe API Mediation Layer and verified it by accessing the REST API endpoint.