Create an APIM Instance

* Basics – call out Organization Name / Admin email
* Monitoring – Talk about performance hit

Import and publish a backend API

1. Go to APIs
2. Click on Open API tile
3. Open API spec -> <https://conferenceapi.azurewebsites.net?format=json>
   * Display Name -> Demo Conference API
   * Name -> demo-conference-api
   * API URL suffix -> conference
4. Versioning
5. Test new API in Azure portal -> Get Speakers
6. Test new API in Postman

Create and Publish a Product

1. Go to Products
2. Click Add
3. Talk through Add screen
4. Add API to a product

Transform and protect your API

*Strip Response Headers*

1. Go to Test tab of Demo Conference API
2. Execute GetSpeakers
3. Note X-ASPNet-Version + X-Powered-By
4. Go to design – all operations
5. Add outbound policy – set-header
6. Delete headers

*Replace URLs*

1. Rerun the GetSpeakers
2. Note the backend URLs
3. Design -> All Operations -> edit outbound
4. Transformation policies – Mask URLs in content

*Rate Limiting*

1. Design -> All operations -> inbound
2. Snipets -> Limit clal rate per key
3. Hit GetSpeakers multiple times

Monitor

1. Go to Metrics
2. Talk about Capacity / Requests
3. Add filter
4. Can set an alter
5. Resource Logs

Debug

1. Test GetSpeakers
2. Review Trace info

Revisions

1. Show Revision listing
2. Go to Revisions tab
3. Add a Revision
4. Add an operation
   1. Display name -> test
   2. Name -> test
   3. URL -> POST test
5. Back to Revisions
6. Make current

Add Cache Policy

Mock API Response

1. Create HTTP API
2. Add operation
   1. Display Name – Get Communities Listing
   2. Name – get-CommunityListing
   3. Url - /communities
   4. Add 200 Response
3. Add Policy – Mock Response
   1. API Management Response – 200 OK, application/json

Developer portal

developer.glennissolutions.com