Lab 1 - Design & Create an API Proxy with OpenAPI Specification

Use case

You have a requirement to create a reverse proxy to handle requests from the Internet and forward them to an existing service. You have decided to follow a design first approach and built a reusable component, a specification which can be used to describe the API contract, generate API documentation, generate API test cases, etc., using the OpenAPI Specification. You would like to generate an Apigee API Proxy by using the OpenAPI Specification (fka: Swagger) instead of building the API Proxy from scratch.

How can Apigee API Management help?

Apigee enables you to quickly expose services as APIs. You do this by creating an **API proxy**, which provides a facade for the service that you want to expose, such as existing API endpoints, generic HTTP services, or applications (such as Node.js). The API proxy decouples your service implementation from the API endpoint that developers consume. This shields developers from future changes to your services as well as implementation complexities. As you update services, developers, insulated from those changes, can continue to call the API, uninterrupted. On Apigee, the API Proxy is also where runtime policy configuration is applied for API Management capabilites. For further information, please see: <u>Understanding APIs and API Proxies</u>.



Apigee also supports the <u>OpenAPI specification</u> out of the box, allowing you to autogenerate API Proxies. Apigee has a built-in OpenAPI specification editor and store which you can use to design and maintain your OpenAPI specifications.



In this lab, we will learn how to

- design an OpenAPI specification for an existing HTTP service and store it within the Apigee platform, and
- create an API proxy that routes inbound requests to an existing HTTP service.

Pre-requisites

- Basic understanding of <u>OpenAPI Specification</u> (FKA: Swagger)
- Access to an HTTP client to test the API (eg. cURL, Postman, etc.). If you do not have access to one, you can use the <u>Apigee Trace Tool</u> or the <u>Apigee REST client</u>.

Instructions

Note: During this workshop, as you may be working within an <u>Apigee Organization (Org)</u> that is shared by multiple users. Please prefix all asset names within the Org with your initials. For example, Spec name = {your-initials}_{spec name}, API proxy name = {your-initials}_{proxy name}, etc.

Create an OpenAPI Specification

During the course of this lab, the sample HTTP service we will expose as an API endpoint is the Hipster Products service located at http://cloud.hipster.s.apigee.com/products. First,

we are going to design and create an OpenAPI specification for the different resource endpoints, i.e. /products and /products/{productId}.

- 1. Go to <u>https://apigee.com/edge</u> and log in. This is the Apigee management UI.
- 2. Select **Develop** → **Specs** in the side navigation menu

| a | ravindranv ~ ravindranvtrial-eval | Spe |
|-------------------|---|-------|
| $\langle \rangle$ | Develop | All 🗸 |
| | Specs | NAN |
| | API Proxies | |
| | Shared Flows | |
| | Offline Trace | |
| Ŧ | Publish | |
| ıh | Analyze | |
| ٩ | Admin | |
| 0 | Help | |
| Ŀ | Feedback | |
| | | |

3. As we have a pre-designed sample of the spec available for this lab, we will be importing it into your Apigee Org's Spec Store. Click **+Spec**. Click on **Import URL** to add a new spec from existing source.



- 4. Enter spec details. Replace **{your-initials}** with the initials of your name.
 - File Name: {your-initials}_hipster_products_api_spec
 - URL: <u>https://raw.githubusercontent.com/apigee/apijam/master/Module-</u> <u>1/Resources/products-catalog-spec.yaml</u>



5. Verify the values and click **Import**. Spec has been imported into Apigee and is ready to use. You should see your spec in the list. For example,

| a | ravindranv ~ ravindranvtrial-eval | Specs > Home | |
|-------------------|---|-----------------------------|-------------|
| $\langle \rangle$ | Develop | All • | + Spec + |
| | Specs | NAME A DESCRIPTION | MODIFIED |
| | API Proxies | Spec that you just created! | * ^ / @ |
| | Shared Flows | m | |
| | Offline Trace | Products Backend - | 1 month ago |
| Ŧ | Publish | | |
| ih | Analyze | | |
| 63 | Admin | | |

6. Click on **{your-initials}**_hipster_products_api_spec from the list to access the OpenAPI spec editor & interactive documentation that lists API details & API Resources.



Create an API Proxy

 It's time to create Apigee API Proxy from an OpenAPI Specification. Click on Develop → API Proxies from side navigation menu.



2. Click **+Proxy** The Build a Proxy wizard is invoked.



3. Select **Reverse proxy**, Click on **Use OpenAPI** below reverse proxy option.



4. You should see a popup with list of Specs. Select **{yourinitials}**_hipster_products_api_spec and click **Select.**

| a | kevinford ~ amer-poc12 | API Proxies > Create Proxy | Cancel |
|-----|---------------------------|---|--------|
| <> | Develop | API proxies safely expose backend services to API consumers. Choose a template to create your API proxy. | |
| | Specs | | |
| | API Proxies | Reverse proxy (most common) SOAP service | |
| | Shared Flows | A Text Create a Res Hu or pass through prove for a SCAP | |
| | Offline Trace | Use OpenAPI Spec | |
| Ŧ | Publish | My Spec Upload File Import from URL | |
| -th | Analyze | Home Is and noticeably addressing as | |
| ۲ | Admin | SPEC LAST MODIFIED | |
| 0 | Help | VR_hipster_products_api_spec a few seconds ago | |
| m | Feedback | | |
| | | Creel Stort | |
| | < | | |

- 5. Enter details in the proxy wizard. Replace **{your-initials}** with the initials of your name.
 - Proxy Name: {your_initials}_Hipster-Products-API
 - Proxy Base Path: /v1/{your_initials}_hipster-products-api
 - Existing API: Observe the field value which is auto filled from OpenAPI Spec.

| Proxy details | | | | |
|---|---------------------------|--------------------------|---|--|
| Name | {your initials} | _Hipster-Products-API | | |
| VR_Hipster-Products-API | | Available | | |
| Alphanumeric characters, dash (-) or un | derscore (_) | | | |
| Base path | /v1/{your initi | als}_hipster-products-ap | i | |
| /v1/vr_hipster-products-api | | | | |
| This proxy will handle requests on host | name/base-path Learn more | | | |
| Description | | | | |
| Products API for Hipster App | lication | | | |
| | | | | |
| | | | | |
| Target (Existing API) | | | | |
| http://cloud.hipster.s.apigee.c | om/ | | | |
| The URL of the backend service that the | is proxy invokes | | | |

- 6. Verify the values and click **Next**.
- 7. Select **Pass through (none)** for the authorization in order to choose not to apply any security policy for the proxy. Click Next.



8. You can select which API resources, from the list configured in the OpenAPI Spec, should be exposed. Select all & Click on **Next**

| Proxy details | Policies - | 3 Flows | 4 Virtual hosts 5 Summary | |
|---------------------------------|-------------------|-----------------------------------|---------------------------|--|
| OpenAPI operations | | | | |
| Select the operations to genera | ate condition flo | ows. You can update the operation | ns later. Learn more | |
| PATH | VERB | OPERATION | SUMMARY | |
| /products | GET | GetProducts | | |
| /products/{productId} | GET | GetProductDetails | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

9. Go with the **secure Virtual Host** configuration. Ensure that the **default** one is unchecked.

| Proxy details 🗸 | Policies Virtual hosts 5 Summary |
|---|--|
| /irtual hosts for incoming traffic, select the vi | rtual hosts this proxy will bind to when it is deployed. Learn more |
| VIRTUAL HOST | HOSTALIAS |
| secure 🗸 | https://ww-demo3-prod.apigee.net (Environment: prod) https://ww-demo3-test.apigee.net (Environment: test) |
| default | http://ww-demo3-prod.apjgee.net (Environment: prod) http://ww-demo3-test.apjgee.net (Environment: test) |
| Make su | re this is unchecked |
| | |
| | |
| | |
| | |

10. Ensure that only the **test** environment is selected to deploy to and click **Create and Deploy**

| Summary Proxy name VE, Hister-Products-API Proxy type Reverse Proxy Proxy base path /v1/vr_lipster-products-api Target http://cloud.hister.s.apigee.com/ Policies Authorization - Pass through (no authorization) OpenAPI Spec Hipster Products API Virtual hosts secure Optional Deployment prod verse protain portai The prexy will begin handling requests only aftar its deployed to an environment. | Proxy details — | Policies Slows | Virtual hosts | 5 Summary |
|--|------------------|---|---------------|-----------|
| Proxy name VR. Higster-Products-API Proxy type Reverse Proxy Proxy base part V/1/vr./Bigster products-api Target Mtp://Cloud.hipster.s.apigee.com/ Policies Authorization - Pass through (no authorization) OpenAPI Spee Hipster Products API Virtual hosts secure | Summary | | | |
| Prox type Reverse Proxy Prox base and //L/v_/lipiter-products-api Target http://cloud.hipiter.s.apigee.com/ Police Autorization - Pass through (no authorization) OpenAPI Spec Hipiter Products API Vitual hosts secure Optional Deployment | Proxy name | VR_Hipster-Products-API | | |
| Procy base path /v1/vr_flipster: products api Target http://cloud.hipster.sa.pigee.com/ Policies Authorization - Pass through (no authorization) OpenAPI Spec Hipster Products API Virtual hosts secure | Proxy type | Reverse Proxy | | |
| Target http://cloud.hipsters.apige.com/ Policies Authorization - Pass through (no authorization) OpenAPI Spee Hipster Products API Virtual hosts secure Optional Deployment Prod Prod | Proxy base path | /v1/vr_hipster-products-api | | |
| Policies Authorization - Pass through (no authorization) OpenAPI Spec Hipster Products API Virtual hosts secure Optional Deployment | Target | http://cloud.hipster.s.apigee.com/ | | |
| OpenAPI Spec Hipster Products API Virtual hosts secure Optional Deployment | Policies | Authorization - Pass through (no authorization) | | |
| Virtual hosts secure Optional Deployment | OpenAPI Spec | Hipster Products API | | |
| Optional Deployment prod prod protal portal The proxy will begin handling requests only after it's deployed to an environment. | Virtual hosts | secure | | |
| | Optional Deploym | ent | | |

11. Once the API proxy is created and deployed click **Edit Proxy** to view your proxy in the proxy editor.

| Created proxy: VR_Hipster-Products-API Deployed proxy to environment: test The proxy is now handling requests at the following URLs: Environment URL | Sreated proxy: VR_Hipster-Products-API Deployed proxy to environment: test xxy is now handling requests at the following URLs: nment URL bitse:financiamo 2 data paintse ent/of for biaster-products-asi | | | |
|--|--|----------------------------------|--|----|
| Deployed proxy to environment: test The proxy is now handling requests at the following URLs: Environment URL | Deployed proxy to environment: test xxy is now handling requests at the following URLs: " mment URL bits: "financiamo 2-last aplace out/of for bioster-producto-val | Created prox | y: VR_Hipster-Products-API | |
| The proxy is now handling requests at the following URLs: Environment URL | oxy is now handling requests at the following URLs: imment URL https://www.down-9.dot.t.psi.nov.out/of/or.hipstors.org/urls-yri | Deployed pro | xy to environment: test | |
| Environment URL | innent URL | The proxy is now hand | ing requests at the following URLs: | |
| | https://www.domp?ctast.anjago.pot/w?/wr.hinstor.products.anj | Environment | URL | |
| test https://ww-demo3-test.apigee.net/v1/vr_hipster-products-api | indian www.neuros-reacapigeeners.vit.vi_inbster-bioducts/abi | test | https://www-demo3-test.apigee.net/v1/wr_hipster-products-api | Co |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

12. *Congratulations!* ...You have now built a reverse proxy for an existing backend service. You should see the proxy **Overview** tab.

| API Proxies | > VR_ ations | Hipster-Products-A | PI > O | Deployment to environments | Proxy overview | W DEVELOP TRACE | Runtime Trace PERFORMANC |
|-----------------------------|-----------------|-----------------------------------|------------|---|------------------|-------------------------|--------------------------------|
| Revision 1 Su | immary | / | | | | | |
| Created: a few second | ls ago , Upd | lated: a few seconds ago . | | | | | |
| Description: Products | API for Hip | ister Application | | | | | |
| Deployments | | | | | | | |
| Dep | loyed en | vironment | | API | endpoint URL | | |
| Environment | REVIS | on a | catus | URL | | | |
| test | 1 | | • | https://ravindranvtrial-eval-test.apigee.net/v1/vr_hips | ter-products-api | Click to view API resou | rce paths |
| Proxy Endpoir | nts | API | base pat | th | Expand All Colla | apse All | |
| Name | | Base Path | | Target Endpoints | | | |
| default | L | /v1/vr_hipster-products-api | | default | | | |
| Endpoint Flow Nar | me | | Method | Path / Condition | | | |
| PreFlow | | | ALL | n/a | | | |
| GetProducts | | | GET | /products | | | |
| GetProductDetails | (| | GET | /products/* | | | |
| PostFlow | | | ALL | n/a | | | |
| Target Endpoi | ints | Ja | arget back | kend service endpoint | Expand All Colla | apse All | |
| Name | Target | | - | Used by Proxy Endpoints | | | |
| ▶ default | UR. | http://cloud.hipster.s.apigee.com | (| default | | | |

Test the API Proxy

Let us test the newly built API proxy. You can use any HTTP client like cURL or Postman, or the <u>Apigee Trace Tool</u>.

Using cURL

org = Organization name env = Environment where API is deployed

curl -X GET "https://{{org}}-{{env}}.apigee.net/{{your initials}}_hipster-products-api/products"

Using Trace Tool:

- Navigate to your proxy's **Trace** tab.
- Ensure that the deployed API revision is selected.
- Hit Start Trace Session.

| | | | 1. N | lavigate to Trace ta |
|--|----------------------------------|--|------------------|----------------------|
| 2. Select the deployed rev. | | | OVERVIEW DEVELOP | TRACE PERFORMANCE |
| eployment to Trace Environment test, Revision 1 - | tart Trace Session | Download Trace Session | | |
| Transactions | Send Requests 3. Start Trace | | | |
| Status Method UR Elapore | Method LIRI | | | State |
| | https://ravindranvtrial-eval-tes | t.spigee.net/v1/vr_hipster-products-api/products | | Send |
| | Transaction Map | | | |
| | 6 | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | Phase Details | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| View Options | 2 | | | |
| Transaction Man | | | | |
| Show Disabled Policies | | | | |
| Show Skipped Phases | | | | |
| Show All FlowInfos | | | | |
| Phase Details | | | | |
| Automatically Compare Selected Phase | | | | |
| - Version de la companya de la compa | | | | |

- Wait for Trace session to start.
- Modify the URL to send to a valid API resource **append a '/products' to the end of the URL.**
- Hit 'Send'.

| | 1. W | ait fo | Trace to start | | OVERVIEW DEVELOP | TRACE | PERFOR | MANCE |
|---------------------|------------------------------|--------|--|---|------------------|-------|---------|--------|
| Deployment to Trace | ndronmert test, Revision 1 🕶 | 55 | p Trace Section Remaining Time: 🔘 05:47 | Download Trace Seasion | | | | |
| g Transactions | | | Send Requests | | | | | |
| 월 Sielan Mell | ed UN | theord | Method URL CBT https://ravindranvtrial-eval-tes | st.apigee.net/v1/vr_hipster-products-api/products | | - 1 | Send | Status |
| | | | Transaction Man | 2. Append '/products' to URL | | 3. H | it Send | |

• You will see that the API proxy recieved the request and sent back a HTTP status 200 response which was logged by the Trace session. You can click on the step shown below to view the response body.

| API Proxies > VR_Hipster-Pro | ducts-API > Trace > 1 | | | | |
|---|--|--|---|---|--|
| | | | OVER | RVIEW DEVELOP TRACE | PERFORMANCE |
| eployment to Trace Environment test, Revision 1 - | Stop Trace Session Remaining Time: 📀 08: | Download Trace Session | | | |
| Transactions | < Send Requests | | | | |
| Status Method URI Elapse | Method URL | | | | Status |
| 1 200 GET /v1/vr_hipster- products- api/products | ans GET https://ravindranvtrial | I-eval-test.apigee.net/v1/vr_hipster-products-a | pi/products | | Send 200 |
| 1. Successiui response logged | Phase Details server x-envoy-upstream-service-time | enov 2 3. Response b o | enwy dy | 2. Select to v | view response (Back Next) |
| View Options Transaction Map Show Databased Patiente Show All Patientes (2) Show All Patientes (2) Phase Details G. Anoma Solido Consee Selected Pase | Response Content Body | ("products":[("id":"0LJCSPC72","n ame":"Vilage Typevriter","descrip tion":"This typevriter looks good in your Living room.", "pitture:"Y static/img/roducts/typevriter.jp g", "pricedust": ("virtersc/ode":"US 0), "categories": ("virtersc"), ("id 0), "categories": ("virtersc"), ("id 0), "categories": ("virtersc"), ("id 0), "categories": ("virtersc"), "us 0," categories": ("virtersc"), "us 0," categori | <pre>("products":[("Ld":"OLJCEPCZ2","nam """This typeriter looks good in yo g/roducts/yoperiter.jop","pricedsd anos":p90000000%)."categories":["Viut e Camera Lears","description:""Too wo bably deesn't work anyway."."picture g","pricedsd":["CurrencyCode":"USD","</pre> | <pre>e":"Vintage Typewriter", ur living room.","pictur ":{"currencyCode":"USD", age"}),("id":"66VCHSJNUP on't have a camera to us ":"/static/img/products/ "units":"12","nanos"</pre> | "descriptio e":"/static/im "units":"67","n ","name":"Vinta e it and it pro camera-lens.jp |
| Show variables | | etural." Intatic lima laradurte leamar | | | |

Save the API Proxy

- 1. Let's save the API Proxy locally as an API Bundle so that we can reuse it in other labs.
- 2. Save the API Proxy by downloading the proxy bundle, See screenshot below for instructions.

| - | • | | | | | | |
|---|--------------------------------|-----------|-----------------|---|-------------------------|--|--|
| Trainet T Save Revision 1 T | | | Denloyment - | | | | |
| iave as New Revision iave as New API Proxy Undo All | 1ary dated: 3 minutes ago . | | | | | | |
| Jelete API Proxy Jelete Current Revision | or Hipster Application | | | | | | |
| Download Revision Upload a New Revision | Revision | Status | URL | | | | |
| NPI Proxy History | 1 | • | https://amer-po | c12-test.apigee.net/v1/vr_hipster-produ | cts-api | | |
| Proxy Endpoir | nts | | | | Expand All Collapse All | | |
| Name | Base Path | | | Target Endpoints | | | |
| default | /v1/vr_hipster-products | i-api | | default | | | |
| larget Endpoi | nts | | | | Expand All Collapse All | | |
| Name | Target | | | Used by Proxy Endpoints | | | |
| dofault | http://cloud.hipster.s.ar | igee.com/ | | default | | | |

Lab Video

If you like to learn by watching, here is a short video on creating a reverse proxy using OpenAPI Specification - <u>https://www.youtube.com/watch?v=3XBG9QOUPzg</u>

Earn Extra-points

Now that you have created a reverse proxy using an OpenAPI spec, click on the Develop tab and explore the flow conditions populated from the OpenAPI spec. Further expore the trace tab in the API Proxy editor which supports filtering traces as well as downloading trace data for offline use. Also, explore the OpenAPI Spec editor which allows you to edit an OpenAPI specification and supports bidirectional navigation between the yaml/json and a live view of the rendered OpenAPI spec. See how you can generate an API Proxy using the "Generate proxy" action in the Specs view.

| a | kevinford ~ | Specs > Home | | | | | | |
|-------------------|---------------|------------------------------|----------------|---|---|---|--|--|
| $\langle \rangle$ | Develop | Alt | | | | | | |
| | Specs | NAME. | Generate proxy | | | | | |
| | API Proxies | VR_hipster_products_api_spec | * | ٥ | 0 | ſ | | |
| | Shared Flows | | | | | | | |
| | Offline Trace | | | | | | | |
| Ŧ | Publish | | | | | | | |

Quiz

- 1. How do you import the proxy bundle you just downloaded?
- 2. How does Apigee handle API versioning?
- 3. Are there administrative APIs to create, update, or delete API proxies in Apigee?

Summary

That completes this hands-on lesson. In this simple lab you learned how to use Apigee to proxy an existing backend using an OpenAPI Specification and the Apigee proxy wizard.

References

- Useful Apigee documentation links on API Proxies -
 - Build a simple API Proxy <u>http://docs.apigee.com/api-</u> services/content/build-simple-api-proxy
 - Best practices for API proxy design and development -<u>http://docs.apigee.com/api-services/content/best-practices-api-proxy-design-and-development</u>
- Watch this "4 Minute Video 4 Developers" (4MV4D) on "Anatomy of an API Proxy" https://youtu.be/O5DJuCXXIRg