Brij Kishore Pandey 
Save For Later 
API Gateway 
In a Nutshell 
Clients 
Web 
Include & 
Exclude List 
Mobile 
API Gateway 
Request 
Validation 
protocol 
conversion 
1 
Microservices 
AuthN & 
AuthZ 
O 
O 
Rate 
Limit 
Dynamic 
O Routing 
APIs 
Exception 
Handling 
Monitoring 
1 
1 
Observability 
Tools Like Grafana 
Circuit 
Breaker 
cache 
1 
1 
1 
Redis 

API Gateway

API Gateway: A Deeper Technical Dive

1. Routing and Request Aggregation:

API Gateways play a crucial role in handling requests by routing them to the appropriate microservices. For example, if you have separate services for user authentication, product information, and order management, the API Gateway will direct requests to these services based on the request path or other criteria. Furthermore, it can aggregate results from multiple services into a single response to streamline client-side processing.

2.   Load Balancing and Service Discovery:

In a microservices architecture, a single service might be running on multiple instances for load distribution and redundancy. An API Gateway often includes a load balancer to distribute incoming requests evenly across these instances. It also integrates with service discovery mechanisms to route requests to available service instances dynamically.

3.   Security:

API Gateways enhance security in several ways:

* 𝗔𝘂𝘁𝗵𝗲𝗻𝘁𝗶𝗰𝗮𝘁𝗶𝗼𝗻 𝗮𝗻𝗱 𝗔𝘂𝘁𝗵𝗼𝗿𝗶𝘇𝗮𝘁𝗶𝗼𝗻: It can enforce authentication and authorization policies, ensuring that only valid and permitted requests reach backend services.
* 𝗧𝗵𝗿𝗲𝗮𝘁 𝗣𝗿𝗼𝘁𝗲𝗰𝘁𝗶𝗼𝗻: It acts as a shield against common security threats such as SQL injection, cross-site scripting (XSS), and others by validating request formats and payloads.
* 𝗥𝗮𝘁𝗲 𝗟𝗶𝗺𝗶𝘁𝗶𝗻𝗴 𝗮𝗻𝗱 𝗤𝘂𝗼𝘁𝗮𝘀: To protect backend services from being overwhelmed by too many requests, the API Gateway can impose rate limits and quotas on clients.

4. API Transformation and Orchestration:

API Gateways can transform requests and responses to and from backend services. This includes transforming protocols (e.g., from HTTP to WebSocket), rewriting request paths, or converting data formats. It can also orchestrate complex service interactions, making it easier to implement composite services or workflows.

5. Monitoring and Analytics:

By being the entry point for all service requests, API Gateways are in a prime position to monitor traffic and gather data. They can track metrics such as request count, response times, and error rates, providing valuable insights for performance optimization and capacity planning.

6. Caching:

To reduce the load on backend services and improve response times, API Gateways can cache responses. For instance, data that doesn’t change frequently (like product descriptions in a catalog) can be cached at the gateway level.

7. Version Management and Canary Releases:

API Gateways facilitate API versioning, allowing clients to use different versions of an API simultaneously. They also support canary releases (gradual rollouts of new features), directing a subset of traffic to new service versions for testing purposes.The key players in API Gateway Landscape which

would be competitive for DataPower gateway

1. AWS API Gateway
2. Microsoft Azure API Management
3. Google Cloud Endpoints
4. Kong
5. Apigee
6. MuleSoft Anypoint Platform
7. NGINX
8. Red Hat 3scale

My Learning Journey in Data Power

Deployment choice and flexibility 
CLOUD READY 
o. Signed & encrypted 
VIRTUAL 
O 
api 
Signed & encrypted 
Gateway Stack 
IBM Optimized 
Embedded OS 
REST & SOAP 
LINUX 
api 
Signed & encrypted 
Gateway Stack 
IBM Optimized 
Application layer 
REST & SOAP 
O 
C_90 
api 
Signed & encrypted 
Gateway Stack 
IBM Optimized 
Application layer 
REST & SOAP 
CLOUD PAK 
o —n Sigred & encrypted 
Gateway Stack 
IBM Optimized 
Application layer 
api 
Gateway Stack 
IBM Optimized 
Embedded OS 
REST & SOAP 
Trwted Platform & 
Hardware Security 
api 
o o • Crypto Acceleratim 
OpenShift Cmtair*-r 
REST & SOAP 
2 

How Data Power evovled:

There are many choices on how the user wants the datapower. There are many choices to deploy the datapower.

Its all about having this gateway available, where you need it? In fact there is a physical form factor available as well.

Physical form factor is (DMZ ready most secured data power, very popular and very secure)

IBM Data power documentation:

<https://www.ibm.com/docs/en/datapower-operator>

Data power Features Blog: <https://pronteff.com/category/ibm-datapower/>

Learning about : **Secure Pipeline services**: PlayBook

<https://pages.github.ibm.com/secure-pipelines-service/sps-playbook/sps-onboarding/adoption-overview>

Stepzen - GraphQL

Event automation- Connect the dots between business events to spot new trends, threats, and opportunities. Put events to work and respond in real-time.

, API connect, App connect, Aspera, Data power

Integration portfolio

75% - Hardware

25% - Virtual

9007, 9008 are the h/w variants.

9009 - Jan 2026

9006 - phased out

Understand where Datapower is:

Every has to do 1 slide what Datapower does.

Data Power Learning Link From WWW

IBM official DataPower learning resource:

[How to Create an MPGW Processing Policy on IBM DataPower Gateway](https://www.youtube.com/watch?v=YrtYmea68Jg&list=PL_4RxtD-BL5tDrXtsCvDNRsW3V0QlODGi)

A computer and a device

Description automatically generated with medium confidence

[How IBM DataPower Gateways Boosts Your Mobile, IoT & More](https://www.youtube.com/watch?v=sNHwZrVq6AA)

A blue and white logo

Description automatically generated

[IBM DataPower Tutorial | Explore IBM DataPower In Less Than An Hour [ Learn DataPower ] - MindMajix](https://www.youtube.com/watch?v=fjB_4lwHl3c)

Explore: 
IBM Datapower 
In about 70 minutes 

[Secure Gateway Transactions Faster Than Ever With Our New User Interface in IBM DataPower Gateway](https://www.youtube.com/watch?v=FiV16wy6EiY&list=PL_4RxtD-BL5tDrXtsCvDNRsW3V0QlODGi)

IBM 
DataPower 
Gateway 

[DataPower Probe tips & tricks](https://www.youtube.com/watch?v=ls0xmIdK3ts&t=1s)

How does it work? 
(externals) 
• on a 
• rt Of the 
• option tor MPWG: XML FW ws-proxy 
• Not 
• is a 
• Terrwar,' of service 

[IBM Datapower Installation on VMWare.](https://www.youtube.com/watch?v=lN3Od1A9Hqc&list=PLx76DzrwX6H3OOALmMT-BxOQJ398HEN2b&index=3)



<https://docs.traceable.ai/docs/ibm-datapower-1>

Useful learnings about Tenant

<https://www.ibm.com/docs/en/datapower-gateway/10.6?topic=tenants-creating-tenant-installing-its-firmware>

DataPower Commands list:

<https://www.ibm.com/docs/en/datapower-gateway/10.5.x?topic=reference-commands>

Configuring steps for Tenant connections:

<https://www.ibm.com/docs/en/datapower-gateway/10.5.0?topic=tenants-defining-connections-tenant>

**Serial port or Telnet port in DP:**

Shared by pchan

<https://www.ibm.com/support/pages/connecting-serial-console-datapower-gateway-or-mq-appliance>

Documentation link for Tenant:

<https://www.ibm.com/docs/en/datapower-gateway/10.5.0?topic=administration-tenants>

Shared by Karen: **To setup Failure Notification during Reboot.**

<https://www.ibm.com/support/pages/best-practices-failure-notification-settings-datapower>

CVE Fixing process overview:

<https://ibm.ent.box.com/folder/266640505676?s=8gej6qmaa9c8ihj1vth8nnozi3l6orf3>

Overview of CVE process:

We get vulnerabilities from cyber security team, Cyber security team usually scan our code.

They scan our code using the following tools (OSS Scan/ Mend scan).

Based on the scan result the tool will report the CVE's or vulnerabilities

CVE Process Overview

===================

**Which issues should be present in JIRA and which ones should be in GIT?**

There's a lot of automation built up around scan assessments.  Those will stay in Git for the foreseeable future.  If we need to do work to address a scan finding, that work should be done in a Jira issue (and linked back to the scan finding), but the finding itself (and its assessment) will remain in Git

**When should we update JIRA tickets for equivalent CVE?**

Right - you only need to open a Jira issue if you need to update a library or make a code-change to address a "Vulnerable" or "Not-Vulnerable-But-Visible" findingJust do the assessment labeling in Git (remove the SPbD-Assessment-Needed" label and replace it with "Vulnerable" or "Not-Vulnerable" or "Not-Vulnerable-but-visible", and possibly add "not-shipped" for findings against e.g. code that's only for test base, or that's built but never included in the firmware).  Also add a comment explaining the reasoning behind the assessment.

**What are the sequence of steps to follow when processing CVE related issue?**

If any findings do need remediation, then you'll need to open a Jira issue to actually make that change.  Once the change is made and finds its way into release branches, the scan should notice, and close the finding in Git.  The only time they won't is if we have to back-port a patch to address something (or create our own patch).  In that case, the scanner won't know what we've done, so you'll have to go back to the finding issue and mark it as a false-positive (and explain it's been addressed by a patch, and link the Jira issue that did the patch).

Throttler Learning

Throttler configurations: <https://www.ibm.com/docs/en/datapower-gateway/10.5.x?topic=commands-throttle-settings>

Throttler configurations: <https://www.ibm.com/docs/en/datapower-gateway/10.5.x?topic=commands-throttle-settings>

How to setup Docker for Throttler;

Below are the steps I followed to setup throttler in docker.

After running the NO\_MCP=1 build commands below set of commands will copy the tar file and unzip it and install the docker.

  448  NO\_MCP=1 make -j10 -C datapower/distro-ng debug\_datapower-ubi.tar.gz | tee ubiCopy.txt

  449  cd \_no\_mcp\_build/x86\_64/datapower/distro-ng/

  450  ls

  451  cd Desktop/DataPowerCode/dp-dev-branch-throttler/

  452  set -o vi

  453  docker images

  454  docker run --platform linux/amd64 -d -v ~/Desktop/dp-ui/scripts/Travis/config:/opt/ibm/datapower/drouter/config -e DATAPOWER\_ACCEPT\_LICENSE=true --name datapower -p 443:443 -p 9090:9090 -p 5554:5554 -p 6200-6210:6200-6210 -p 13003:13003 -p 5080:5080 -p 3334:3334 -p 35312:35312 -p 4100:4100 -p 10001:10001 -p 2053:2053 d524323aa75e

  455  pwd

  456  mkdir config

  457  cd config/

  458  pwd

  459  ls

  460  unzip \*

  461  pwd

  462  cd dp-ui-main/scripts/Travis/config

  463  ls

  464  pwd

  465  docker images

  466  docker run --platform linux/amd64 -d -v /home/venkatim/config/dp-ui-main/scripts/Travis/config:/opt/ibm/datapower/drouter/config -e DATAPOWER\_ACCEPT\_LICENSE=true --name datapower -p 443:443 -p 9090:9090 -p 5554:5554 -p 6200-6210:6200-6210 -p 13003:13003 -p 5080:5080 -p 3334:3334 -p 35312:35312 -p 4100:4100 -p 10001:10001 -p 2053:2053 d524323aa75e

  467  docker images

  468  /scripts/Travis/config:/opt/ibm/datapower/drouter/config -e DATAPOWER\_ACCEPT\_LICENSE=true --name datapower -p 9090:9090 -p 5554:5554 -p 6200-6210:6200-6210 -p 13003:13003 -p 5080:5080 -p 3334:3334 -p 35312:35312 -p 4100:4100 -p 10001:10001 -p 2053:2053 d524323aa75e

  469  ls /home/venkatim/config/dp-ui-main/scripts/Travis/config

  470  pwd

  471  ls /home/venkatim/config/dp-ui-main/scripts/Travis/config

  472  docker run --platform linux/amd64 -d -v /home/venkatim/config/dp-ui-main/scripts/Travis/config:/opt/ibm/datapower/drouter/config -e DATAPOWER\_ACCEPT\_LICENSE=true --name datapower -p 443:443 -p 9090:9090 -p 5554:5554 -p 6200-6210:6200-6210 -p 13003:13003 -p 5080:5080 -p 3334:3334 -p 35312:35312 -p 4100:4100 -p 10001:10001 -p 2053:2053 d524323aa75e

  473  docker ps

  474  docker container

  475  docker containers

  476  run

  477  sest -o vi

  478  set -o vi

  479  docker run --platform linux/amd64 -d -v /home/venkatim/config/dp-ui-main/scripts/Travis/config:/opt/ibm/datapower/drouter/config -e DATAPOWER\_ACCEPT\_LICENSE=true --name datapower -p 443:443 -p 9090:9090 -p 5554:5554 -p 6200-6210:6200-6210 -p 13003:13003 -p 5080:5080 -p 3334:3334 -p 35312:35312 -p 4100:4100 -p 10001:10001 -p 2053:2053 d524323aa75e

  480  docker ps

  481  docker images

  482  docker rm datapower

  483  docker run --platform linux/amd64 -d -v /home/venkatim/config/dp-ui-main/scripts/Travis/config:/opt/ibm/datapower/drouter/config -e DATAPOWER\_ACCEPT\_LICENSE=true --name datapower -p 443:443 -p 9090:9090 -p 5554:5554 -p 6200-6210:6200-6210 -p 13003:13003 -p 5080:5080 -p 3334:3334 -p 35312:35312 -p 4100:4100 -p 10001:10001 -p 2053:2053 d524323aa75e

  484  docker rm datapower

  485  docker run --platform linux/amd64 -d -v /home/venkatim/config/dp-ui-main/scripts/Travis/config:/opt/ibm/datapower/drouter/config -e DATAPOWER\_ACCEPT\_LICENSE=true --name datapower -p 9090:9090 -p 5554:5554 -p 6200-6210:6200-6210 -p 13003:13003 -p 5080:5080 -p 3334:3334 -p 35312:35312 -p 4100:4100 -p 10001:10001 -p 2053:2053 d524323aa75e

When running the docker it is very important to make it interactive. Only then we'll be able to configure the port that listens for GUI.

docker run -it -e DATAPOWER\_ACCEPT\_LICENSE=true -e DATAPOWER\_INTERACTIVE=true -p 9090:9090

By entering the above command when running the docker it becomes interactive.

To ebnable the web gui run the below commands.

configure; web-mgmt 0 9090; exit;  enables the webgui.

To enable throttler via CLI run below commands

Idg# top; co; throttle;

Inside throttler below are the commands to be used.

<https://www.ibm.com/docs/en/datapower-gateway/10.5.x?topic=commands-throttle-settings>

Thank you for quoting example of OOMKiller... So throttler acts as a OS service, when memory is less it reacts with trying to release some memory...

you mentioned about user space.. Please explain what are the apps in user space and the ones in kernel space if there is one…

Container means IDG or Linux installs

Good Write up in CV using DP experience

- Engage hundreds of federal clients at the technical and executive levels to create and sell multi-million dollar software solutions involving cloud integrations, software development, and cutting edge operations techniques.

- Discover and understand business problems, translate to technical solutions, and create solutions that lead to software and services sales.

- Adapt to rapidly changing assignments, market demands, and customers, by technically mastering cutting edge software portfolios that included high speed data transport, Kubernetes, Open Shift, message queues, Kafka, security gateways, API lifecycle management, DevSecOps, multi-cloud management, and AIOps technologies.