# Infrastructure Power Management System “PowerMan”

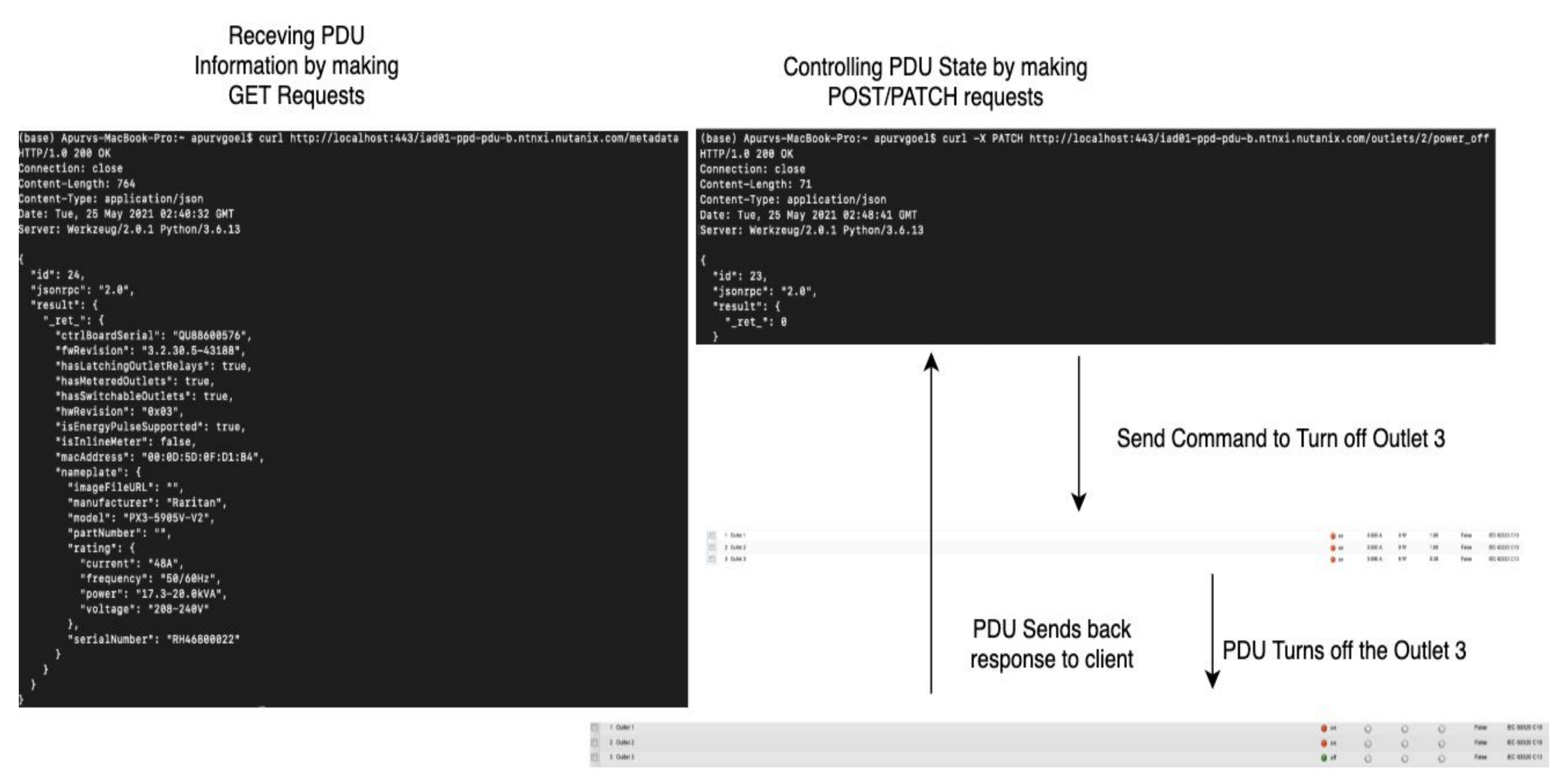
**STUDENTS:** Apurv Goel, Christian Lancaster, Wenxuan Yang

## Introduction and Background

- Nutanix is a cloud computing company that offers a variety of services for on-prem and hybrid cloud deployments.
- Nutanix offers a disaster recovery service, Xi Leap
- Xi Leap utilizes Nutanix own data centers to provide reliable storage for disaster recovery
- Presently, the Nutanix data centers do not have a central solution for power management in their data center
- To perform even a trivial power cycle operation, they must file an IT ticket, which is costly in terms of time and human labor.
- There are many equipment vendors with unique interfaces.

## System Requirements

- Nutanix is looking for a service that can remotely control and monitor power distribution in their many data centers.
- The proposed system will provide an API which is capable of performing various query and control operations on power distribution units (PDUs)
- The system shall provide a single API which is capable of interacting with PDUs from different vendors
- The system shall utilize an authentication mechanism to ensure that only authorized users can control the PDUs remotely
- The system should cache redundant queries to conserve bandwidth and to reduce PDU downtime
- The system should switch multiple outlets in a controlled manner, so as not to exceed the PDUs peak power limitation.
- The system should be able to scale to *thousands* of hosts.

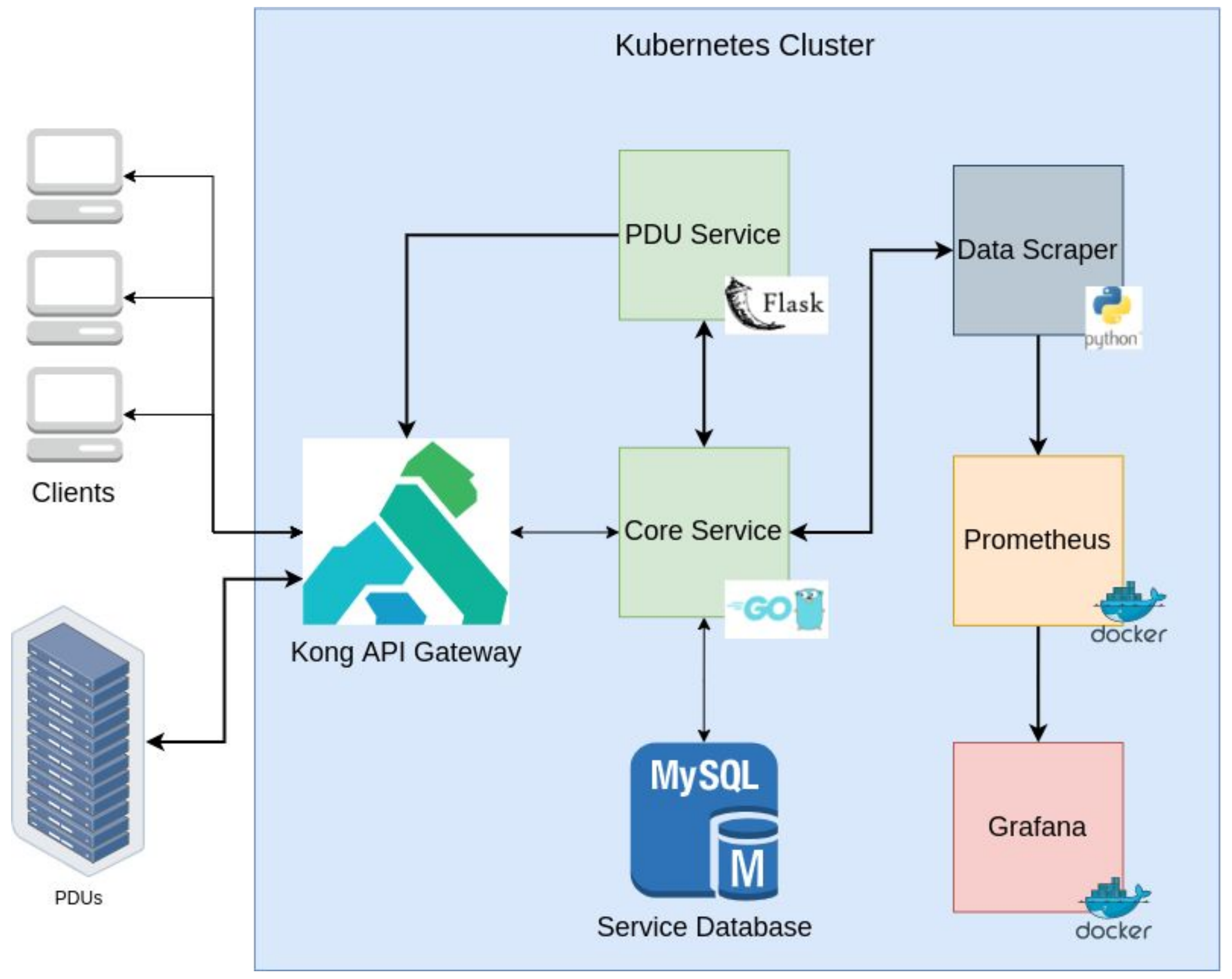

Receving PDU Information by making GET Requests
Controlling PDU State by making POST/PATCH requests
Send Command to Turn off Outlet 3
PDU Sends back response to client
PDU Turns off the Outlet 3

```
(base) Apurvs-MacBook-Pro:~ apurvgoel$ curl http://localhost:443/iad01-ppd-pdu-b.ntnxi.nutanix.com/metadata
HTTP/1.0 200 OK
Connection: close
Content-Length: 764
Content-Type: application/json
Date: Tue, 25 May 2021 02:40:32 GMT
Server: Werkzeug/2.0.1 Python/3.6.13

{
  "id": 24,
  "jsonrpc": "2.0",
  "result": {
    "_ret_": {
      "ctrlBoardSerial": "QU88600576",
      "fwRevision": "3.2.30.5-43188",
      "hasLatchingOutletRelays": true,
      "hasMeteredOutlets": true,
      "hasSwitchableOutlets": true,
      "hwRevision": "0x03",
      "isEnergyPulseSupported": true,
      "isInlineMeter": false,
      "macAddress": "00:0D:5D:0F:D1:B4",
      "nameplate": {
        "imageFileURL": "",
        "manufacturer": "Raritan",
        "model": "PX3-5905V-V2",
        "partNumber": "",
        "rating": {
          "current": "48A",
          "frequency": "50/60Hz",
          "power": "17.3-20.0kVA",
          "voltage": "208-240V"
        },
        "serialNumber": "RH46B00022"
      }
    }
  }
}
```

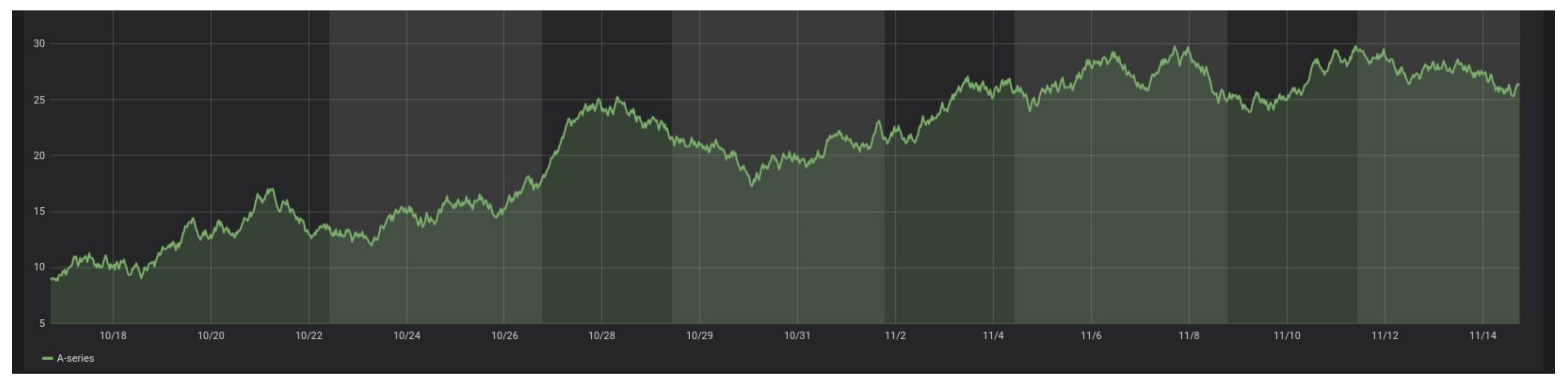
```
(base) Apurvs-MacBook-Pro:~ apurvgoel$ curl -X PATCH http://localhost:443/iad01-ppd-pdu-b.ntnxi.nutanix.com/outlets/2/power_off
HTTP/1.0 200 OK
Connection: close
Content-Length: 71
Content-Type: application/json
Date: Tue, 25 May 2021 02:48:41 GMT
Server: Werkzeug/2.0.1 Python/3.6.13

{
  "id": 23,
  "jsonrpc": "2.0",
  "result": {
    "_ret_": 0
  }
```

## Implementation

- Our system is designed as a microservices type application, which utilizes Docker and Kubernetes to provide workload isolation and scalability.



- The Kong API Gateway serves as an Ingress Controller to our Kubernetes cluster, and handles request authentication and validation
- The Core Service is a Golang application which is responsible for maintaining the state of the system using a MySQL database, and forwarding requests to the appropriate handler.
- The PDU Service is a Python Flask application which is responsible for translating the API request from the uniform API to the vendor specific API, sending the request to the PDU, and processing the response.



- We have also included a visualization service, which utilizes Prometheus and Grafana to continually poll the Core Service and retrieve time-series data about power consumption.

## Technologies Used

Flask
GO
Kong
POSTMAN
kubernetes
docker
Grafana
MySQL
Prometheus

## Future Development

- Prometheus Alertmanager can be used to notify site engineers of any issues with power distribution, such as excessive load.
- The PowerMan API can be used in future data center automations.
- For example, the power consumption information can be used to intelligently scale applications by switching racks or single nodes.
- The data can also be used to generate a heatmap for visual analysis.
- PowerMan can be extended to support additional PDU vendors.
- Okta integration is planned, which would allow for role-based access.

## Conclusion

- Our team was successfully able to implement the proposed system by developing on a local Kubernetes cluster. The application is able to process incoming requests, securely communicate with devices in the Nutanix data center, and return a response to the user.

## Acknowledgments

- We would like to thank our industry partners Aman Nijhawan, Brian Finn, Ben Nordan, Kevin Grady, Damon Cole, and others at Nutanix for their mentorship and support.
- We would also like to thank our course staff and our faculty mentor for their wisdom and encouragement throughout this project.
- Finally, we want to thank the open source community for making these projects available for our team to use and build upon.

ELECTRICAL & COMPUTER ENGINEERING
UNIVERSITY *of* WASHINGTON

**ADVISOR:** Dr. Baosen Zhang

**SPONSOR:** Nutanix, Inc.