

ProxCloud: A Portal for Proxmox in NextCloud

Student: Robertson Velez

Instructor/Faculty: *Seyedmasoud Sadjadi*, Florida International University

PROBLEM

Integration of Nextcloud and Proxmox.

The coexistence of powerful tools such as Nextcloud and Proxmox poses a challenge due to the absence of native compatibility.

This capstone project aims to address this gap by developing a seamless integration solution using APIs.

CURRENT SYSTEM

Web development using Javascript.

Creation of a custom third-party app within the Nextcloud ecosystem.

Leveraging the capabilities of the Nextcloud and Proxmox API, this app will act as a bridge, facilitating communication between Nextcloud instances and Proxmox servers.

REQUIREMENTS



Web Server:

- Apache or Nginx for hosting the Nextcloud instance.



Proxmox Server:

- Version 6.0 or above.

Programming Languages:

- PHP for developing the Nextcloud app.
- JavaScript for creating dynamic and interactive user interfaces.



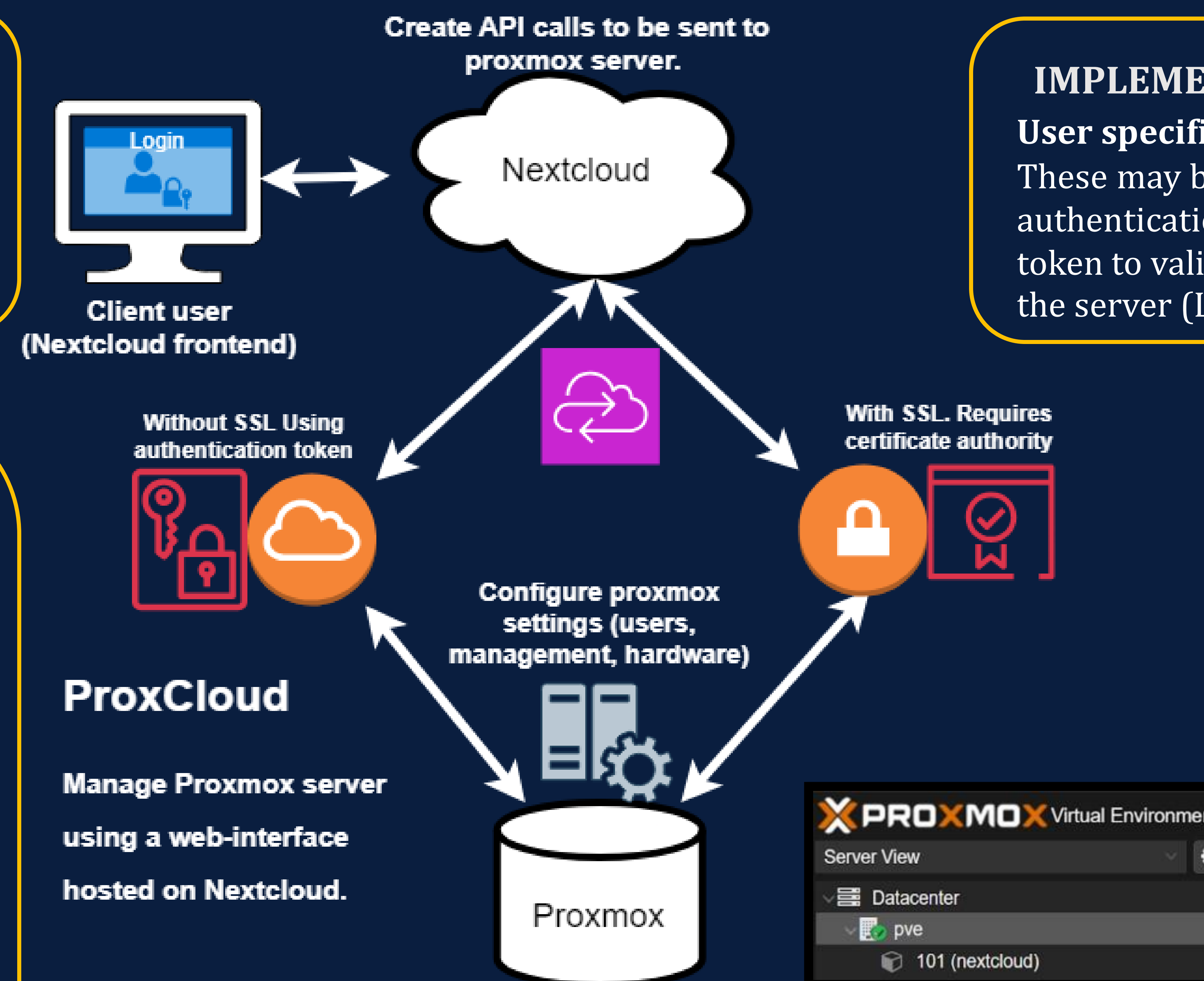
SYSTEM DESIGN

Nextcloud client and Proxmox server.

Manage Proxmox server using a web-interface hosted on Nextcloud. Integrated API calls within Proxmox allow the user to authenticates and manage resources.

SUMMARY & CHALLENGES

Throughout the course of this integration project, a few challenges have emerged. Notably, issues related to certificate authorities and SSL have surfaced. Additionally, Cross-Origin Resource Sharing (CORS) limitations, inherent in web browsers security policies. These introduced obstacles in facilitating communication between the Nextcloud app and the Proxmox server. Despite these challenges, the project has striven to develop a comprehensive solution, considering not only the technical intricacies but also the critical aspects of security and user experience.

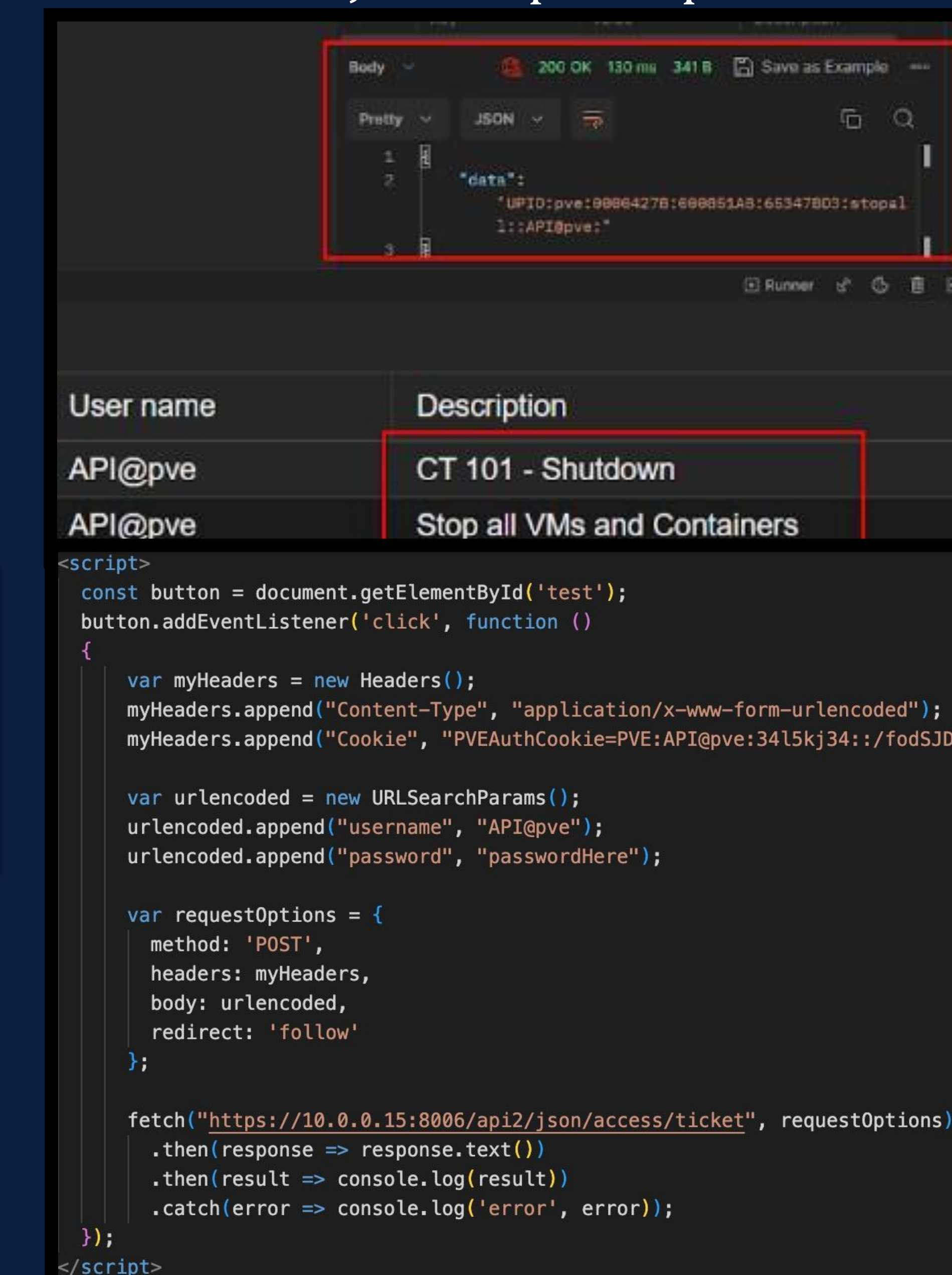


IMPLEMENTATION & VERIFICATION

User specifies login Credentials.

These may be used to obtain a temporary ticket and authentication token (Valid for 24 hours). Using the token to validate the user, they may communicate to the server (Limited access).

API call with JavaScript sample code:



```

const button = document.getElementById('test');
button.addEventListener('click', function () {
  var myHeaders = new Headers();
  myHeaders.append("Content-Type", "application/x-www-form-urlencoded");
  myHeaders.append("Cookie", "PVEAuthCookie=PVE:API@pve:3415kj34:~/fodSJD");

  var urlencoded = new URLSearchParams();
  urlencoded.append("username", "API@pve");
  urlencoded.append("password", "passwordHere");

  var requestOptions = {
    method: 'POST',
    headers: myHeaders,
    body: urlencoded,
    redirect: 'follow'
  };

  fetch("https://10.0.0.15:8006/api2/json/access/ticket", requestOptions)
    .then(response => response.text())
    .then(result => console.log(result))
    .catch(error => console.log('error', error));
});
  
```

User name	Description
API@pve	CT 101 - Shutdown
API@pve	Stop all VMs and Containers

REFERENCES

- Proxmox VE API. https://pve.proxmox.com/wiki/Proxmox_VE_API
- i12bretro. *Quick overview of Proxmox VE REST API*. YouTube. <https://www.youtube.com/watch?v=Iuyd2TXe8SA>
- Develop for Nextcloud: App development. <https://nextcloud.com/developer/>

ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by XXXXX (name of the project sponsor: federal, state or local government, or corporate partners, where applicable). We/I thank XXXX for his/her/their assistance and mentorship that I/we received throughout the senior design project.