You are working as a DevOps Engineer at XYZ Corp, an organization that sells health & wellness products to end-users. Recently you have received feedback from your QA team about “Internal Server Errors” while browsing the product page. To troubleshoot the issue, the developers have requested to see the logs of the Production environment. Briefly explain how would you enable your developers to browse the Production environment logs?

**Solution:**

* **Centralized Logging**
* **Using PM2 as a Monitoring tools in the servers**
* **Providing the Developers a role Based Access (IAM roles) in AWS by limiting the specific Permission.**

**Centralized Logging :( Taking Example as One of EC2-usecase)**

* If the application is running on the Ubuntu EC2-Instance, AWS has a Ec2 cloud watch log agent which will be installed on the Ec2 instance, where the Ec2 server logs can Be sent to cloud watch Logs based on the Activity we can Check the Cloud Logs.
* Cloud Watch Logs can be sent to the S3 bucket based on which Developers can have the access for the S3 bucket Which In turn can be used for Debugging Purpose.
* Note: It allows you to set up alarms and notifications based on specific log events

**Using PM2 as a Monitoring tools in the servers (MERN/MEAN application)**

* Pm2 is a real time monitoring system in which it can be used for monitoring the Backend based application such as the Nest js, Node js.
* Pm2 provides a list of Commands for the listing of the Real time logs
* Pm2 start app.js or for nest js application we can have dist folder created as well as pm2 list shows the list of the backend applications that are actually running.
* Pm2 log loglist number, gives the actual logging details in the real time servers

**Providing the Developers a role Based Access (IAM roles) in AWS by limiting the specific Permission.**

**Below Steps are for an AWS Production**

* Creating an IAM group for the developers and Providing the Limited access such as logging into Servers and Checking the logs for the application
* Use and IAM based policies can be used to provide access for the Developers group as well as Guiding On How to use the EC2 servers.

* The application I worked was a Angular+Nest js application, A scenario to above was happened and As a devops Engineer I created a user with EC2 service access only.
* Nest js Applications (Internal Server Issues has been Handled by checking the PM2 logs).
* By Restarting the Backend Applications using pm2 services as well as Providing Specific Port for the Specific Application we can Avoid the Above scenarios in Prod
* Mismatch of Port Numbers for the Application has caused a Major Issue in one of the applications in real time servers
* By Checking the Available Ports as well as the providing specific Ports ,Internal Server Erros Can be Avoided

Note :

* Usage of Nginx Webservers without reverse Proxy can also happen, instead check the /etc/nginx/sites-available/filename in Ubuntu.
* In Windows server, usage of IIS for the web application using the reverse proxy