**CI/CD Pipeline Management (Critical)**

**Criticality:** High – Ensures continuous deployment, development, and delivery are uninterrupted.

* **Pipelines** 
  + Monitor pipeline status for different countries.
  + Fix failures by rolling back or adjusting versions.
  + Verify build success and integrate SonarQube/Checkmarx for code quality/security.
  + Abort and rebuild stuck pipelines; investigate artifact handling issues.

**EKS Cluster & Infrastructure Management (Very Critical)**

**Criticality:** Very High – Any issues here affect production stability and scalability.

* **EKS Updates (v1.25 -> 1.28 upgrades)**
  + Ensure pre-checks before EKS version upgrades.
  + Apply updates incrementally, starting with non-Prod, then Production.
  + Test node groups, resource scaling, and PVC (EBS) functionality post-upgrade.
* **Terraform/Terragrunt Modifications**
  + Fix any deprecated Terraform arguments and Terraform script errors.
  + Run tests in non-Prod environments to confirm successful Terraform changes before pushing to Production.
* **Jenkins Upgrades (2.426.3 -> 2.440.1)**
  + Backup Jenkins configurations before major updates.
  + Create Jenkins images and Helm charts for new versions; push into AWS ECR.
  + Ensure plugin compatibility and performance monitoring post-upgrade.

**IP Whitelisting on WAF Using Terraform (Critical)**

**Criticality:** High – Essential for securing access and controlling traffic flow.

* **Define IP Sets:**  
  Configure IP sets in Terraform to specify the range of IPs for whitelisting.
* **Apply in Stages:**  
  Test changes in non-Prod before applying them in Production for security and stability validation.
* **Automated Updates:**  
  Integrate with CI/CD pipelines to automate and monitor updates to whitelisted IPs.

**SSL Certificate Management (Very Critical)**

**Criticality:** Very High – SSL management is vital for secure communications.

* **SSL Updates (NL/IE country)**
  + Create and validate SSL CSR files for Production and Preprod environments.
  + Roll out SSL updates CSR in non-Prod before applying them in Production.
  + Verify the impact on all dependent services like domain.

**Docker Customization & Debugging (Critical)**

**Criticality:** High – Impacts pipeline performance and deployment efficiency.

* **Docker Image Customization**
  + Use an existing image as the base.
  + Add custom scripts and configurations.
  + Build the custom image with necessary updates.
* **Testing & Debugging**
  + Test the image locally by running it interactively (docker run -it).
  + Check for errors, missing dependencies, and verify custom scripts.
  + Debug by checking logs (docker logs) and attaching to the container for real-time inspection.
* **Deployment to Pipeline**
  + Push the customized image to ECR.
  + Update pipeline configurations to use the new image.
  + Monitor performance and ensure compatibility with existing pipelines.

**Automation & DevOps Tools (Moderate Criticality)**

**Criticality:** Moderate – Automation impacts efficiency, but failures are less disruptive than infrastructure issues.

* **Helm/ECR Updates (package and driver updates)**
  + Update Helm charts and drivers in AWS ECR with new versions.
  + Test these updates in the non-Prod environment before pushing to Production.
* **Jenkins Automation (Pipeline tweaks and image updates)**
  + Modify Groovy scripts to accommodate branch input variables and create new automation pipelines.
  + Test image and automation creation in staging before applying in the live environment.

**Monitoring, Documentation, and Reporting (Lower Criticality)**

**Criticality:** Low – Important for transparency and long-term efficiency, but immediate failure does not impact critical services.

* **Monitoring and PR Reporting**
  + Continuously monitor pipeline statuses.
  + Submit PRs for any required updates and track failures reported to the team.
* **Documentation (pipelines)**
  + Maintain detailed documentation of all pipeline flows, Jenkins’s configurations, and error resolutions.
  + This serves as a backup for future troubleshooting and scaling efforts.

Before eks update