**RELEASE PROCESS**

1.**Release Preparation Procedure:**

* We will have to filter issues through current milestone in powerpay-root.
* We will have to wait for QA signoff and prepare checklist for release in below format.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| SL.No. | Ticket ID | Assignee | Impacted Services | QA Verified | Merge Requests | Code Reviewed | Remarks |
| 01 | 1234 | ABC | Org,Pe,Ge,Config,UI,Cm,Report-gen,Report-process,Kernel,Dse,IAM | Done | Done | present | Good |

* If Ticket ID:1234 Has Powerpay-Config as impacted service ,then we have to check for MR’s for both showcase and Prod\_live and those MR’s have code review label and assigned to Anish Yesuraj.
* If any of the tickets from list missing MR’s or Code Review Lables ,we will have to inform respective developers.
* Once All tickets are good for release then post the status in release group and get confirmation from anish to proceed further .
* Once we got the confirmation from anish,We will have to cut the release branches.
* At first, Note down the all impacted services and start cutting the branches for each service.
* Go to the Each service Project in gitlab and compare develop and master branches,This should show all the commits going for today’s release.
* If yes,Create new release branch from develop and create MR to master.Before submitting the MR’s cross verify all the commits once.
* If not,We will have to get clarity from respective service team and anish/manoj/arun,whether we have to cherrypick or revert the changes,
* If cherrypick needed,then create new release branch from master and cherrypick the required commits to release branch and create MR from release barnch to master branch.
* IF revert needed,First create new release branch from develop and revert the unwanted commits from release to develop branch and compare both release and master branches whether required commits are presented for today’s release or not.
* If everything good,Create MR from release to master branch.
* Repeat the same procedure for all the services except powerpay-config and inform anish/manoj/arun to merge the MR’s for powerpay-config and release branch MR’s of all the impacted services.

**2.Release Implementation Procedure:**

* Once all the MR’s merged,We will have to start the release implementation.
* Login to Jenkins using the url,<https://dev-powerpay.ascent-online.com/jenkins>.
* **Showcase Env:**
* Goto 03-staging folder and Open the respective service folder and trigger build,after successful build it will automatically triggers deployment.
* If request comes for all services build and deploy follow this sequence,

UI>IAM>CM>GE>report-process>report-gen>pe>org.

* Powerpay-UI and CM services are standalone services,we can trigger them individually. Better approach to trigger cm service first then other services.
* **Note:**If there is powerpay-kernel changes present,just trigger kernel build then it will automatically triggers all other services in below format.
* kernal >ge

>org

>dse >pe

>rep-common>rep-proc

>rep-gen

* After all services Build/deploy,restart Org service.
* During deployment,serivce will restart automatically,check the logs of each service by logging into the respective servers and use the below commands to check the logs.
* **CM logs:** tail -f /mnt/ELK/showcase/cm/logs/cm.log
* **Ge logs:** tail -f /mnt/ELK/showcase/powerpay-generalization-service/logs/powerpay-generalization-service.log
* **Report-process logs:** tail -f /mnt/ELK/showcase/report-processing-service/logs/report-processing-service.log
* **report-generation logs:** tail -f /mnt/ELK/showcase/report-generation-service/logs/report-generation-service.log
* **Pe logs:** tail -f /mnt/ELK/showcase/logs/pe.log
* **IAM logs:** tail -f /mnt/ELK/showcase/powerpay-iam-service/logs/ powerpay-iam-service.log
* **Org logs:** tail -f /mnt/ELK/showcase/organization-service/logs/organization-service.log
* **Note:** For PE we have to load the tenants after completing the build/deployment for all services.
* The steps to be followed for loading the tenants are,
* login to the server where PE service is running,For Showcase(192.168.9.21)
* Load all the tenants using below command.

**curl --location --request GET '**[**https://showcase-powerpay.ascent-online.com/showcase/pe/manage/loadAllTenants'**](https://qa-powerpay.ascent-online.com/qa/pe/manage/loadAllTenants') **--header 'Authorization: Basic YWRtaW46c2VjcmV0'**

* After loading the tenants,check the pending compilations using below command,

**tail -f /mnt/ELK/showcase/pe/logs/pe.log | grep -E 'Loading scripts for tenant|Storage Configuration Files|Refreshing storage layouts for tenant|Pending Compilations'**

* Once the pending compilations comes to 0, check all the tenants loaded successfully or not using below commands,

**curl --location --request GET '**[**https://showcase-powerpay.ascent-online.com/showcase/pe/manage/getLoadedTenantList'**](https://qa-powerpay.ascent-online.com/qa/pe/manage/getLoadedTenantList') **--header 'Authorization: Basic YWRtaW46c2VjcmV0'**

* **Prod1&Prod2 Env:**
* Goto 04-production folder and Open the respective service folder and trigger build,after successful build it will automatically triggers deployment for both prod1&prod2.
* If request comes for all services build and deploy follow this sequence,

UI>IAM>CM>GE>report-process>report-gen>pe>org.

* Powerpay-UI and CM services are standalone services,we can trigger them individually. Better approach to trigger cm service first then other services.
* **Note:**If there is powerpay-kernel changes present,just trigger kernel build then it will automatically triggers all other services in below format.
* kernal >ge

>org

>dse >pe

>rep-common>rep-proc

>rep-gen

* After all services Build/deploy,restart Org service.
* During deployment,serivce will restart automatically,check the logs of each service by logging into the respective servers and use the below commands to check the logs.
* **Prod\_live\_1:**
* **CM logs:** tail -f /mnt/ELK/prod\_live/cm/logs/cm.log
* **Ge logs:** tail -f /mnt/ELK/prod\_live/powerpay-generalization-service/logs/powerpay-generalization-service.log
* **Report-process logs:** tail -f /mnt/ELK/prod\_live/report-processing-service/logs/report-processing-service.log
* **report-generation logs:** tail -f /mnt/ELK/prod\_live/report-generation-service/logs/report-generation-service.log
* **Pe logs:** tail -f /mnt/ELK/prod\_live logs/pe.log
* **IAM logs:** tail -f /mnt/ELK/prod\_live/powerpay-iam-service/logs/ powerpay-iam-service.log
* **Org logs:** tail -f /mnt/ELK/prod\_live/organization-service/logs/organization-service.log
* **Prod\_live\_2:**
* **CM logs:** tail -f /mnt/ELK/prod\_live\_2/cm/logs/cm.log
* **Ge logs:** tail -f /mnt/ELK/prod\_live\_2/powerpay-generalization-service/logs/powerpay-generalization-service.log
* **Report-process logs:** tail -f /mnt/ELK/prod\_live\_2/report-processing-service/logs/report-processing-service.log
* **report-generation logs:** tail -f /mnt/ELK/prod\_live\_2/report-generation-service/logs/report-generation-service.log
* **Pe logs:** tail -f /mnt/ELK/prod\_live\_2/logs/pe.log
* **Org logs:** tail -f /mnt/ELK/prod\_live\_2/organization-service/logs/organization-service.log

**03-Load Sample Tenants:**

* After All services Build and deployement,If there is PE changes,we have to load the sample tenants provided by QA team and will have to get the confimation from QA team After loading those sample tenants to load all the tenants.
* **Note:** If Any request comes to run the maintainance API,After All the tenants are loaded and pending compilations reached to 0 then only execute mainatainace API commands otherwise skip all the mainatainace API related sections from “03-Load Sample Tenants and 04-Load All the Tenants”
* **Commands to load sample tenants:**
* **Prod\_live\_1:**
* In terminal,Login the 192.168.3.180 server and execute below commands and replace the IN0000 with your Tenant ID.

**curl --location --request GET 'http://192.168.3.182/prod\_live/pe/manage/getLoadedTenantList' --header 'powerpayTenantId: IN0000' --header 'Authorization: Basic YWRtaW46c2VjcmV0'**

* Check whether the tenant has been loaded or not using below command,

**curl --location --request GET 'http://192.168.3.182/prod\_live/pe/manage/getLoadedTenantList' --header --header 'Authorization: Basic YWRtaW46c2VjcmV0'**

* **Command to run maintainance api for CTC-NET-GRVY:**
* Replace the TenantID IN0000 with your Tenant ID.

**curl -X POST http://192.168.3.182/prod\_live/ge/manage/initTenantConfigJob -H 'Authorization: Basic YWRtaW46c2VjcmV0' -H 'Content-Type: application/json' -d '{"tenants": ["IN0000"], "entityCode":"CTC-NET-GRVY"}'**

* By running the above curl cmd we will get job code which we need to copy and paste it in checkTenantConfigJobStatus to check the status of the activity.

**curl -X GET 'http://192.168.3.182/prod\_live/ge/manage/checkTenantConfigJobStatus?jobCode='xxxxxxxxxxxxx' -H 'Authorization: Basic YWRtaW46c2VjcmV0**

* Similarly for which maintainance api we need to run we will get that confirmation from dev team or qa team and corrosponding to that we will have to run the maintainance api.
* For more details and for other maintanance api details please refer

<https://gitlab.ascent-online.com/pnbhat/power-pay-root/wikis/02_Library_GE/GE-Maintenance-APIs>

* **Prod\_live\_2:**
* In terminal,Login the 192.168.3.180 server and execute below commands and replace the IN0000 with your Tenant ID.

**curl --location --request GET 'http://192.168.3.182/prod\_live\_2/pe/manage/getLoadedTenantList' --header 'powerpayTenantId: IN0000' --header 'Authorization: Basic YWRtaW46c2VjcmV0'**

* Check whether the tenant has been loaded or not using below command,

**curl --location --request GET 'http://192.168.3.182/prod\_live\_2/pe/manage/getLoadedTenantList' --header --header 'Authorization: Basic YWRtaW46c2VjcmV0'**

* **Command to run maintainance API for CTC-NET-GRVY:**
* Replace the TenantID IN0000 with your Tenant ID.

**curl -X POST http://192.168.3.182/prod\_live\_2/ge/manage/initTenantConfigJob -H 'Authorization: Basic YWRtaW46c2VjcmV0' -H 'Content-Type: application/json' -d '{"tenants": ["IN0000"], "entityCode":"CTC-NET-GRVY"}'**

* By running the above curl cmd we will get job code which we need to copy and paste it in checkTenantConfigJobStatus to check the status of the activity.

**curl -X GET 'http://192.168.3.182/prod\_live\_2/ge/manage/checkTenantConfigJobStatus?jobCode='xxxxxxxxxxxxx' -H 'Authorization: Basic YWRtaW46c2VjcmV0**

* Similarly for which maintainance api we need to run we will get that confirmation from dev team or qa team and corrosponding to that we will have to run the maintainance API.
* For more details and for other maintanance api details please refer

<https://gitlab.ascent-online.com/pnbhat/power-pay-root/wikis/02_Library_GE/GE-Maintenance-APIs>

* After loading the sample tenants and maintainance API ,get the confirmation from QA team to load all the tenants.

**04-Load All the Tenants:**

* In terminal,Login the 192.168.3.180 server and execute below Commands.
* **PROD\_LIVE:**
* To load all the tenants in prod\_live,

curl --location --request GET '<http://192.168.3.182/prod_live/pe/manage/loadAllTenants'> --header 'Authorization: Basic YWRtaW46c2VjcmV0'

* To check whether the all the tenants have been loaded or not

curl --location --request GET '<http://192.168.3.182/prod_live/pe/manage/getLoadedTenantList'> --header --header 'Authorization: Basic YWRtaW46c2VjcmV0'

* **PROD\_LIVE\_2:**
* To load all the tenants in prod\_live\_2,

curl --location --request GET '<http://192.168.3.182/prod_live_2/pe/manage/loadAllTenants'> --header 'Authorization: Basic YWRtaW46c2VjcmV0'

* To check whether the all the tenants have been loaded or not

curl --location --request GET '<http://192.168.3.182/prod_live_2/pe/manage/getLoadedTenantList'> --header --header 'Authorization: Basic YWRtaW46c2VjcmV0'

* **Note:**After All the tenants are loaded and pending compilations reached to 0,If needed we will have to run maintainance API using below Commands.
* **Prod\_live:**

Command to load the CTC-NET-GRVY maintenance api. We need to get the jobCode after running this cmd.

**curl -X POST http://192.168.3.182/prod\_live/ge/manage/initTenantConfigJob -H 'Authorization: Basic YWRtaW46c2VjcmV0' -H 'Content-Type: application/json' -d '{"tenants": [], "entityCode":"CTC-NET-GRVY"}'**

We need to enter that jobCode over here to check the status of the maintenance api.

**curl -X GET 'http://192.168.3.182/prod\_live/ge/manage/checkTenantConfigJobStatus?jobCode=xxxxxxxxxxxx' -H 'Authorization: Basic YWRtaW46c2VjcmV0'**

* **prod\_live\_2:**

Command to load the CTC-NET-GRVY maintenance api. We need to get the jobCode after running this cmd.

**curl -X POST http://192.168.3.182/prod\_live\_2/ge/manage/initTenantConfigJob -H 'Authorization: Basic YWRtaW46c2VjcmV0' -H 'Content-Type: application/json' -d '{"tenants": [], "entityCode":"CTC-NET-GRVY"}'**

We need to enter that jobCode over here to check the status of the maintenance api.

**curl -X GET 'http://192.168.3.182/prod\_live\_2/ge/manage/checkTenantConfigJobStatus?jobCode=xxxxxxxxxx' -H 'Authorization: Basic YWRtaW46c2VjcmV0'**

* Similarly for which maintainance api we need to run we will get that confirmation from dev team or qa team and corrosponding to that we will have to run the maintainance API.
* For more details and for other maintanance api details please refer

<https://gitlab.ascent-online.com/pnbhat/power-pay-root/wikis/02_Library_GE/GE-Maintenance-APIs>