# **JRAF Project – Cloudbees**

# **USER GUIDE**

**Author: Jag Sethi**

***How to Request Access for cloudbees and Nexus:***

Please follow the below instruction to request access to cloudbees and new Nexus.

Cloudbee URL:

<https://oc.cvshealth.com/jenkins/>

JRAF project pipeline highlighted in yellow.

·  [Jenkins](https://oc.cvshealth.com/jenkins/)

·  ·  [ci-autoeng](https://ci-autoeng.cvshealth.com/jenkins/)

·  ·  [DBEngineering](https://ci-autoeng.cvshealth.com/jenkins/job/DBEngineering/)

·  ·  [Oracle](https://ci-autoeng.cvshealth.com/jenkins/job/DBEngineering/job/Oracle/)

·  ·  [ans\_ent\_java\_vulnerability\_remediation\_cloudbee](https://ci-autoeng.cvshealth.com/jenkins/job/DBEngineering/job/Oracle/job/ans_ent_java_vulnerability_remediation_cloudbee/)s

Direct link for JRAF Project:

<https://ci-autoeng.cvshealth.com/jenkins/job/DBEngineering/job/Oracle/>

look for [ans\_ent\_java\_vulnerability\_remediation\_cloudbee](https://ci-autoeng.cvshealth.com/jenkins/job/DBEngineering/job/Oracle/job/ans_ent_java_vulnerability_remediation_cloudbee/)s pipeline

Nexus URL:

<https://nexus-sdi.cvshealth.com/nexus/>

You should be part of the following group:

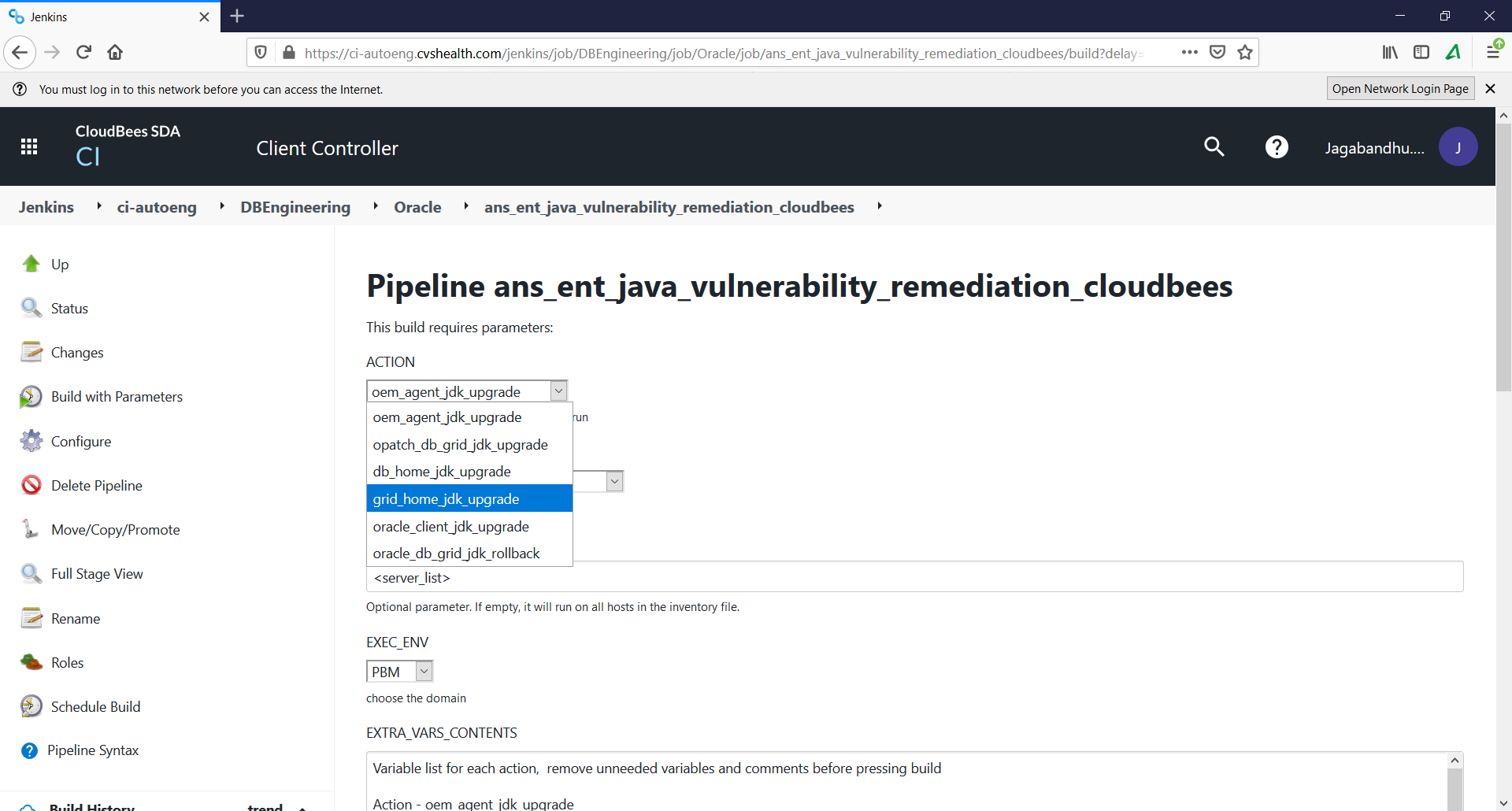
Cloudbee access to view/execute pipeline: ***dso\_ans-dbeng-dev***

Nexus access to upload: dso\_nxs-dbeng

To add access for your id, please use the below link. Click on “+” on right top corner and choose “Add Access” from drop down menu.

Accessnow link: https://aetna.tuebora.com/

***Pipeline for JRAF project:***



**The Action Drop down show the different playbooks.**

Action - oem\_agent\_jdk\_upgrade (Upgrades JDK version in OEM agent home)

Action - opatch\_db\_grid\_jdk\_upgrade (Upgrades JDK version inside OPATCH dir for DB and GRID homes)

Action - oracle\_client\_jdk\_upgrade (Upgrades JDK version in oracle client home)

Action - db\_home\_jdk\_upgrade (Upgrades JDK version inside DB home in JDK dir)

Action - grid\_home\_jdk\_upgrade (Upgrades JDK version inside GRID home in JDK dir)

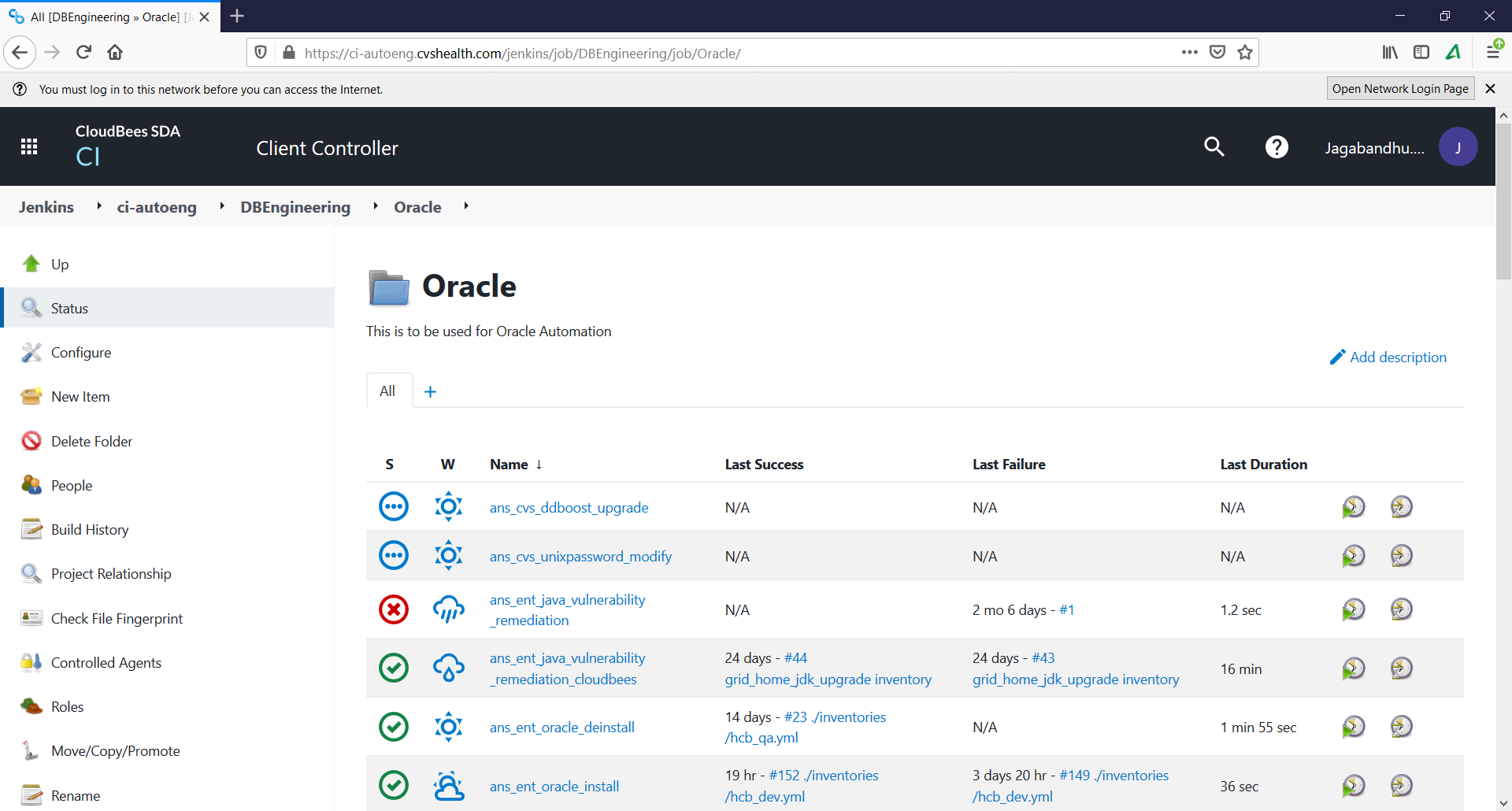
Action: oracle\_db\_grid\_jdk\_rollback (Rollback JDK patch inside DB or GRID homes)

**How to run a playbook:**

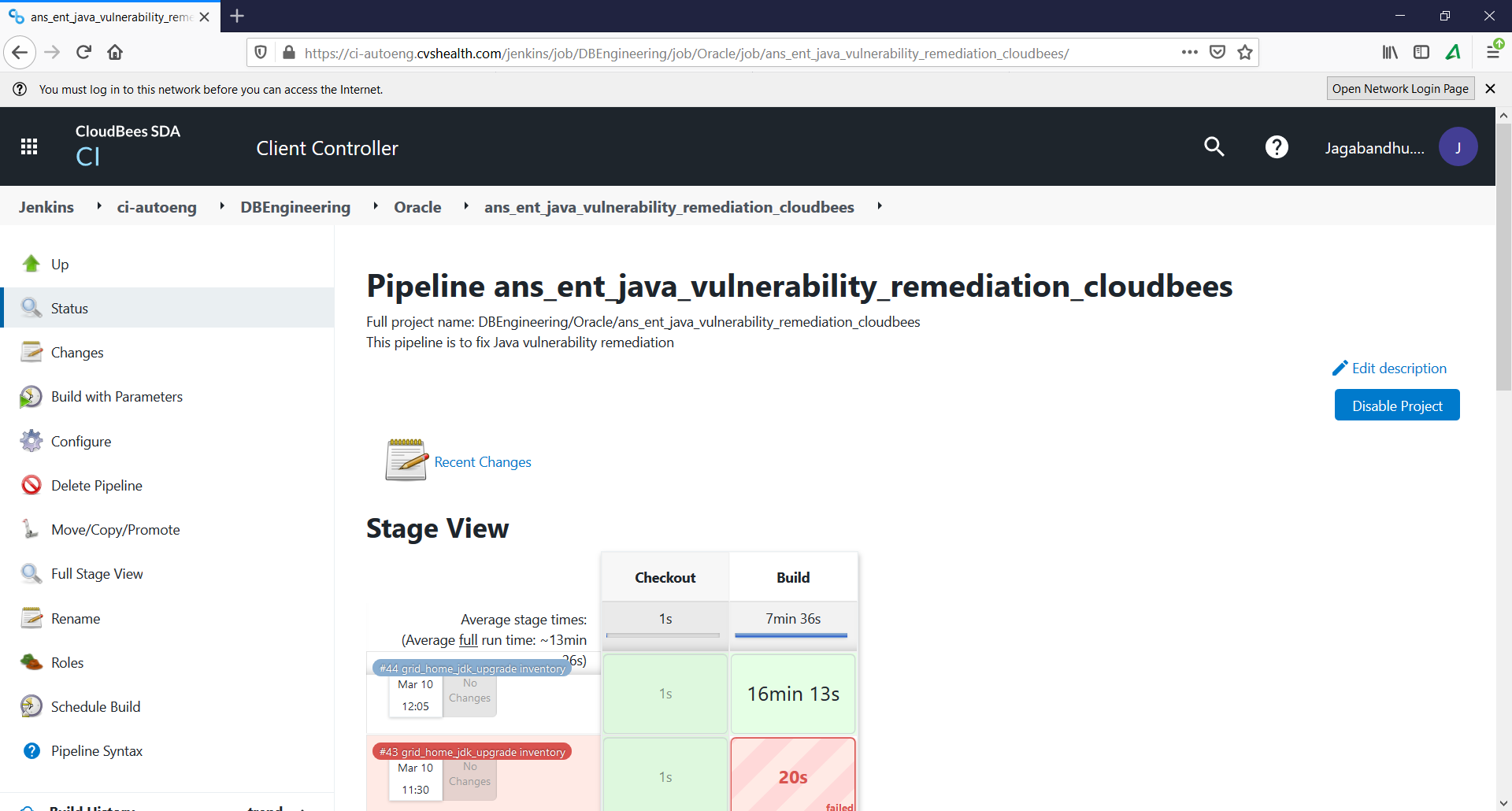
Go the below URL:

<https://ci-autoeng.cvshealth.com/jenkins/job/DBEngineering/job/Oracle/>

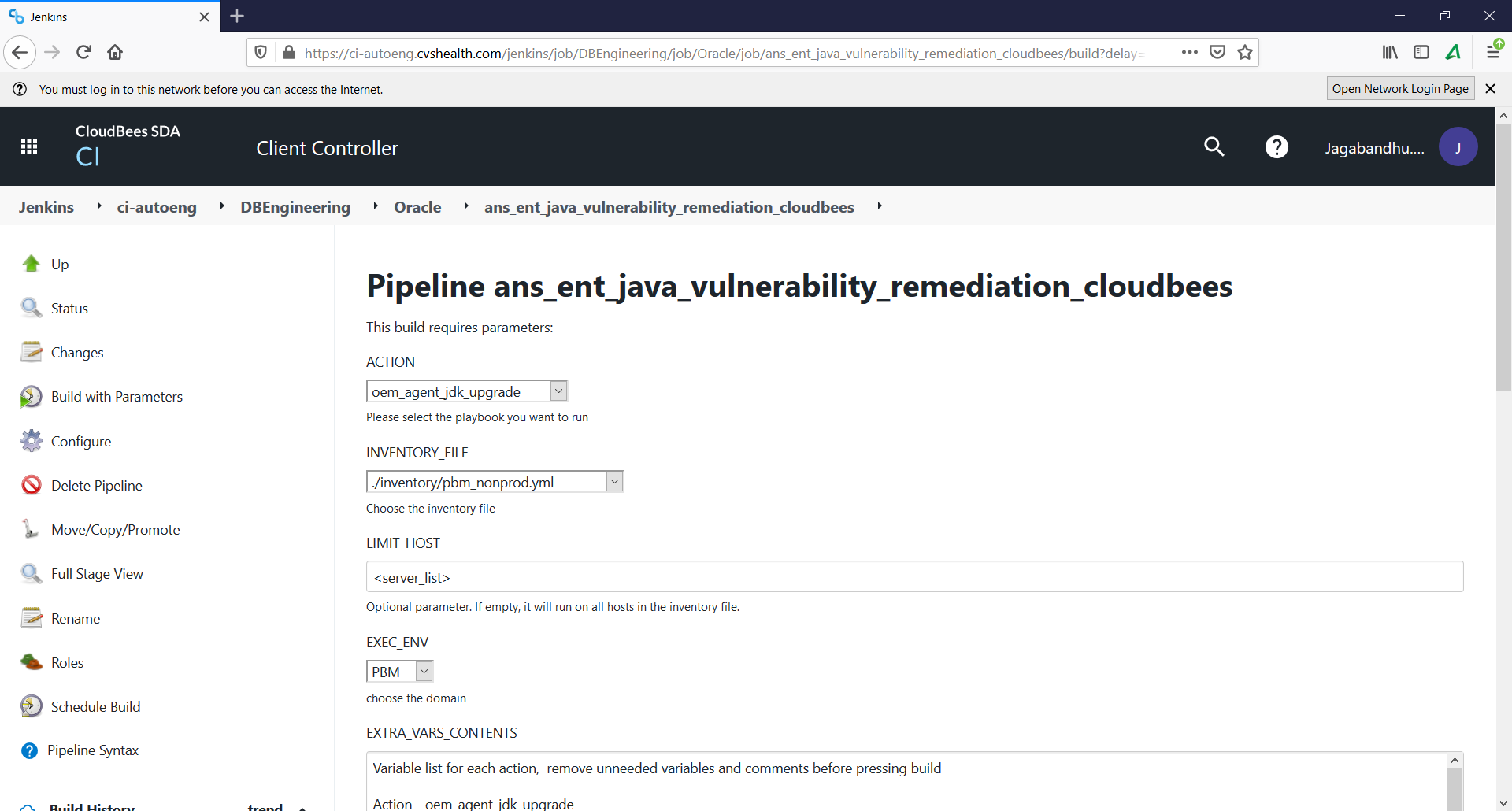
Look for [ans\_ent\_java\_vulnerability\_remediation\_cloudbee](https://ci-autoeng.cvshealth.com/jenkins/job/DBEngineering/job/Oracle/job/ans_ent_java_vulnerability_remediation_cloudbee/) pipeline and click on it …



You will see below screen, look for “**Build with Parameters**” in left side menu bars, if you don’t see that then you don’t have proper access, please request access, process explain at the beginning of this document.

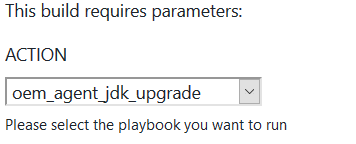


Click on “Build with Parameters” in left side menu bar

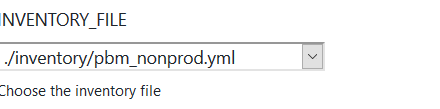


**Parameters of the JRAF pipeline**

1.

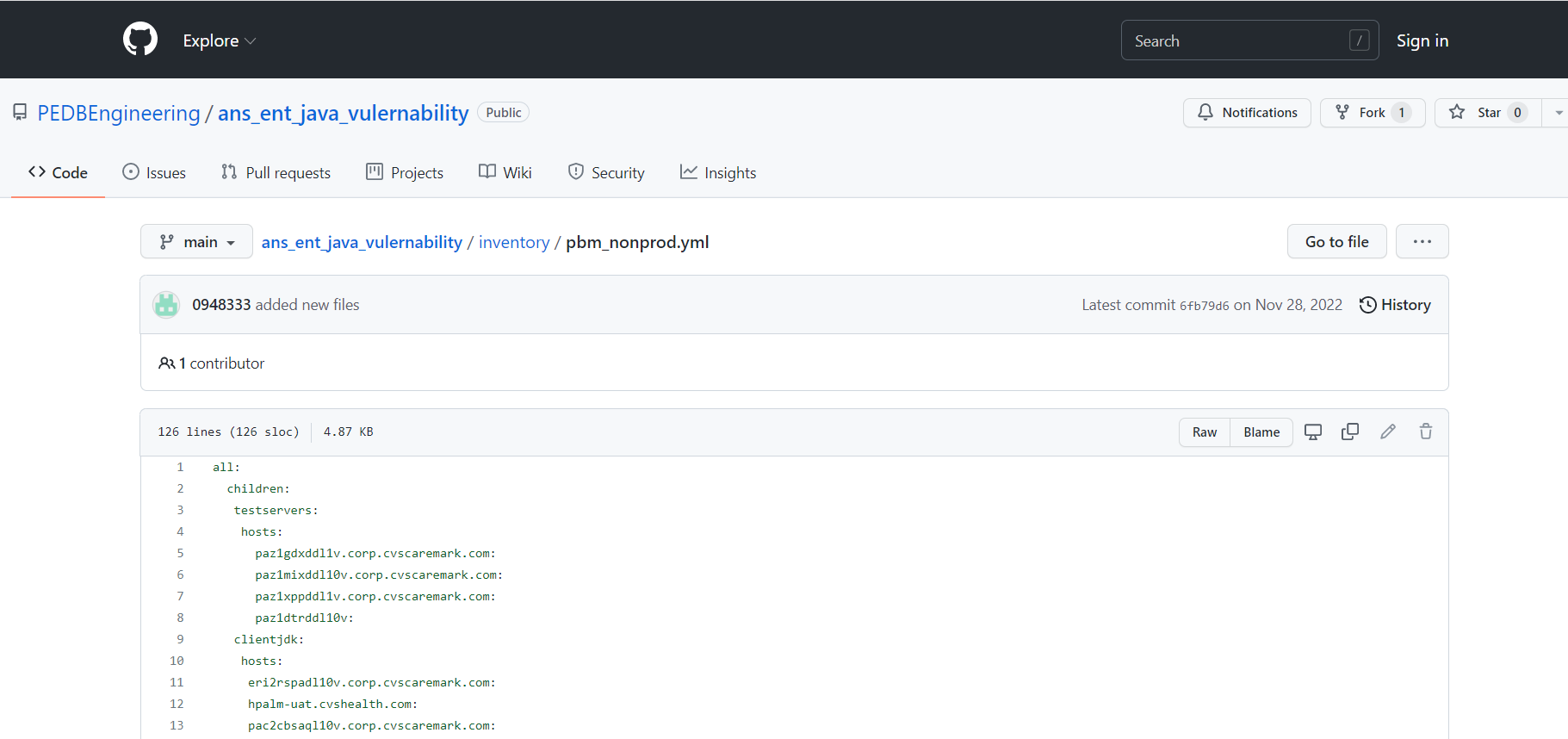


2.

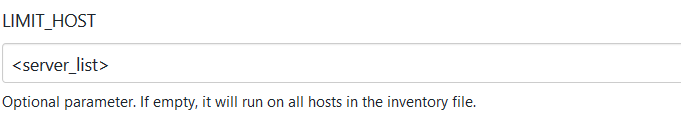


Make sure to choose inventory file based on your LOB and env (DEV/PROD/NON-PROD etc) and also the host that you going to run playbook against is present in the inventory file exactly the way in the inventory file (FQDN)

Example of Inventory file in github:



3.



**Example :**

If you want to run the playbook on a **single host** then:

paz1gdxddl1v.corp.cvscaremark.com

If you want to run the playbook in more the one host then separate host with “:” no spaces should be there before or after

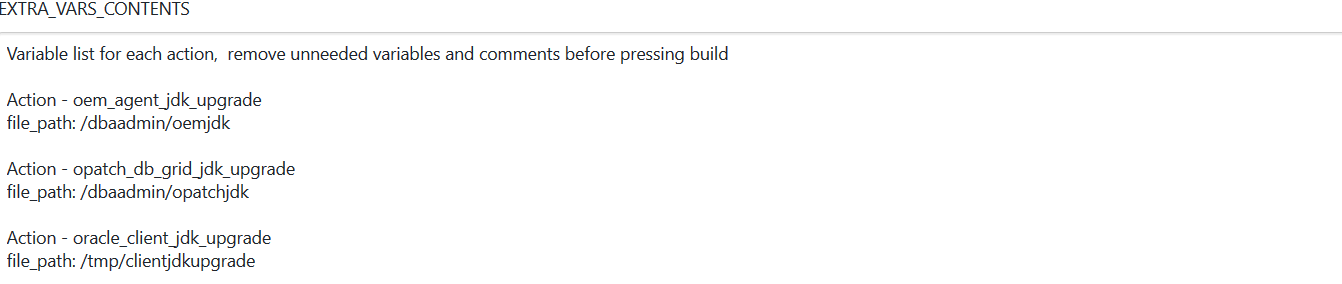
Example of 2 hosts :

paz1gdxddl1v.corp.cvscaremark.com**:**paz1xppddl1v.corp.cvscaremark.com

4.

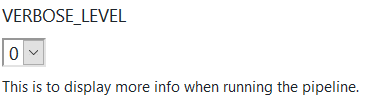


5.





6.

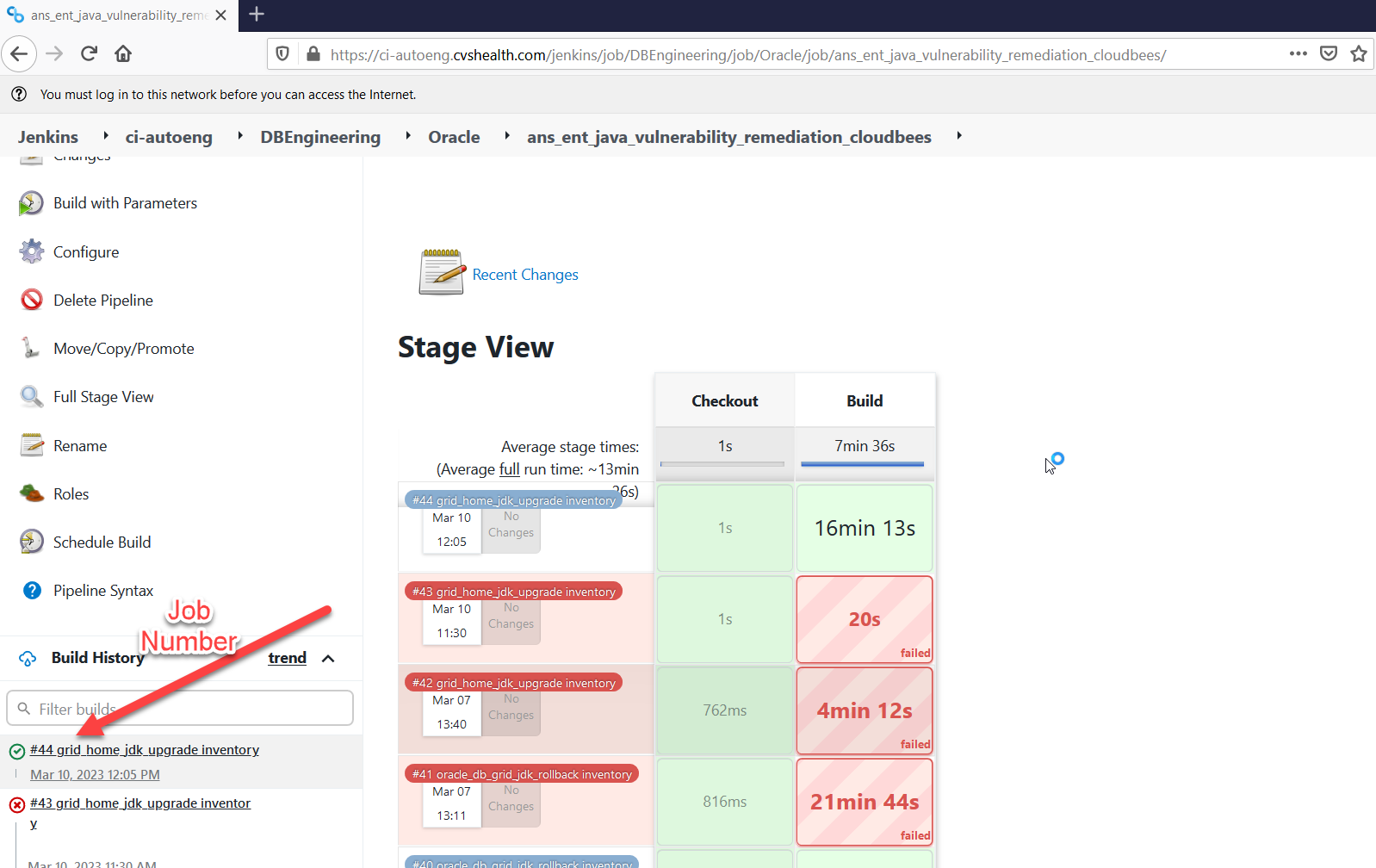


7.

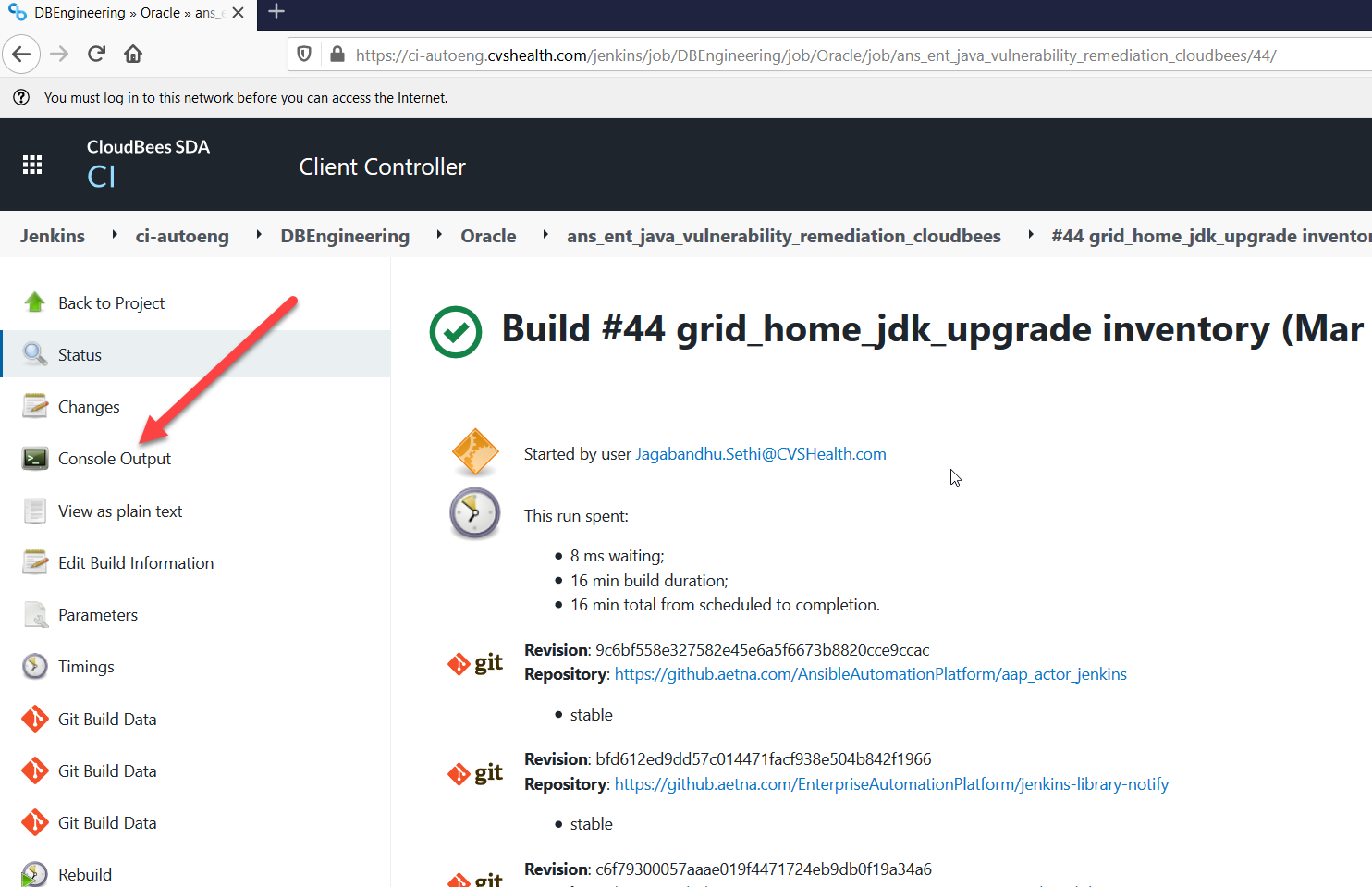


Click on build button once you put all relevant parameters to execute the playbook.

Once you click on “Build”, look for job number like in the screen shot below.



Click on the latest job number that might be running like in the screenshot above.



Click on “Console output” to view the running job log. Wait until the job is done.

**Example of a successful job:**



**Example of a failed job:**

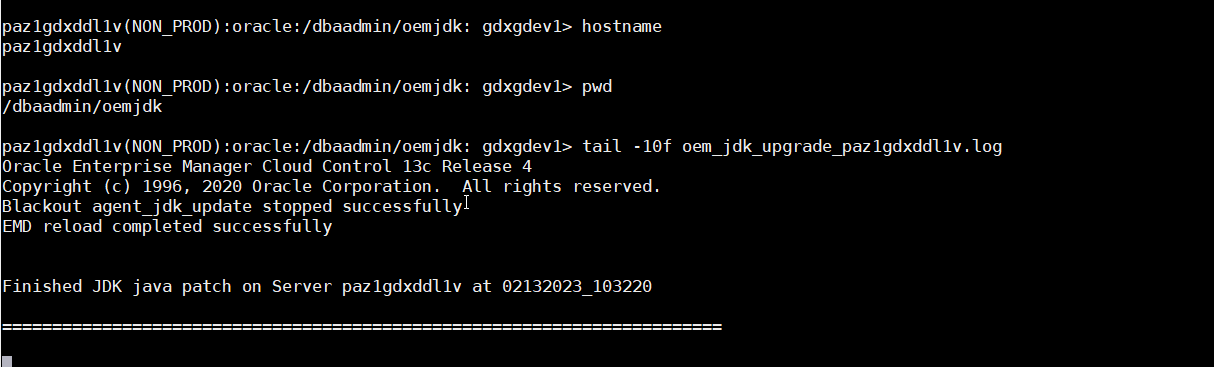


**View logs in target hosts for your playbook:**

Example:

Host name: paz1gdxddl1v.corp.cvscaremark.com

Log Dir path for “Action - oem\_agent\_jdk\_upgrade”: /dbaadmin/oemjdk



**Please verify log in all the target hosts to see if there is any failure and why it failed.**

**Individual Playbook run instructions**

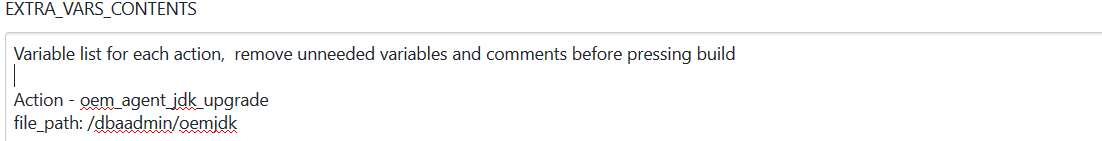
Please cleanup all contents inside EXTRA\_VARS\_CONTENTS box except for the playbook you want to execute. The parameter only should be in JSON format. Any other info which is not in JSON format should be cleaned up. No space or empty lines should be there inside the box.

1. **Action: oem\_agent\_jdk\_upgrade**

**Note**: Parameters like “Action, INVENTORY\_FILE, LIMIT\_HOST, EXEC\_ENV” are same for all playbooks, only the EXTRA\_VARS\_CONTENTS change for each playbook.

***For this playbook NO DOWN TIME needed.***

EXTRA\_VARS\_CONTENTS



Keep those above lines initially then edit the lines as below…final look as below.



Make sure the **file\_path** is exists in your host.

Example:

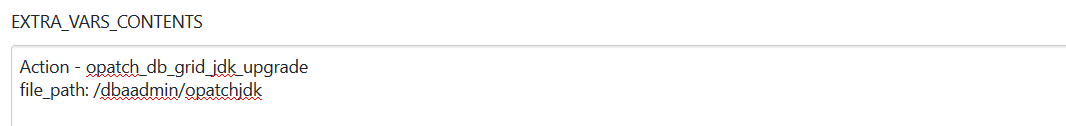
**/dbaadmin** should be there in host. Don’t remove “oemjdk” , if you rerun the same playbook in the same host and you want to preserve the old log from last run you can put something like “oemjdk1 or oemjdk2 etc…after first run

**Note**: you can have any dir name which should be owned by oracle, this a staging dir that is used by pipeline that need to be present in the server otherwise the playbook run will fail in that host.

1. **Action: opatch\_db\_grid\_jdk\_upgrade**

Note: Parameters “Action, INVENTORY\_FILE, LIMIT\_HOST, EXEC\_ENV” are same for all playbooks, only the EXTRA\_VARS\_CONTENTS change for each playbook.

***For this playbook NO DOWN TIME needed.***



**Final look after editing the above screenshot.**



Make sure the **file\_path** is exists in your host.

Example:

**/dbaadmin** should be there in host. Don’t remove “opatchjdk” , if you rerun the same playbook in the same host and you want to preserve the old log from last run you can put something like “opatchjdk 1 or opatchjdk2 etc…after first run

**Note**: you can have any dir name which should be owned by oracle, this a staging dir that is used by pipeline that need to be present in the server otherwise the playbook run will fail in that host.

1. **Action: oracle\_client\_jdk\_upgrade**

Note: Parameters “Action, INVENTORY\_FILE, LIMIT\_HOST, EXEC\_ENV” are same for all playbooks, only the EXTRA\_VARS\_CONTENTS change for each playbook.

***For this playbook NO DOWN TIME needed.***



**Final look after editing the above screenshot.**



Make sure the **file\_path** is exists in your host.

Example:

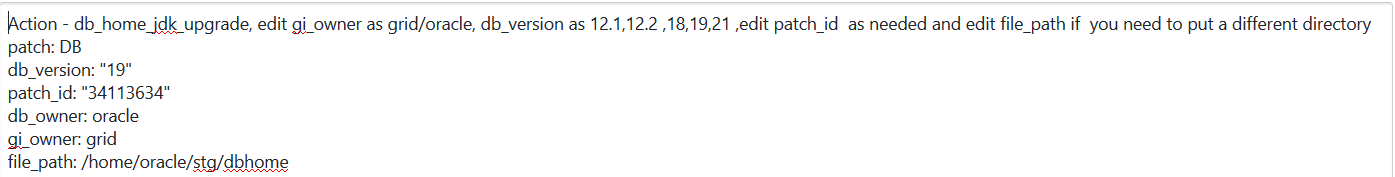
Please **don’t change** anything for this playbook. Don’t remove “clientjdkupgrade” , if you rerun the same playbook in the same host and you want to preserve the old logs from last run you can put something like “clientjdkupgrade1 or clientjdkupgrade2 etc…after first run

1. **Action: db\_home\_jdk\_upgrade**

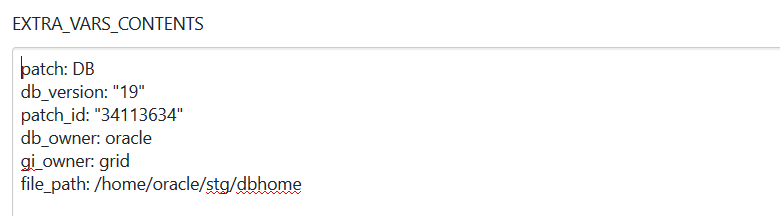
Note: Parameters “Action, INVENTORY\_FILE, LIMIT\_HOST, EXEC\_ENV” are same for all playbooks, only the EXTRA\_VARS\_CONTENTS change for each playbook.

***For this playbook NO DOWN TIME needed.***

EXTRA\_VARS\_CONTENTS



**Final look after editing the above screenshot.**



Make sure the **file\_path** is exists in your host.

Example:

**/home/oracle/stg** should be there in the host server. Don’t remove “dbome”, if you rerun the same playbook in the same host and you want to preserve the old logs from last run you can put something like “dbhome1 or dbhome2 etc…after first run

**Note**: you can have any dir name as staging dir not necessarily “**/home/oracle/stg”** which should be owned by oracle, this a staging dir that is used by pipeline that need to be present in the server otherwise the playbook run will fail in that host.

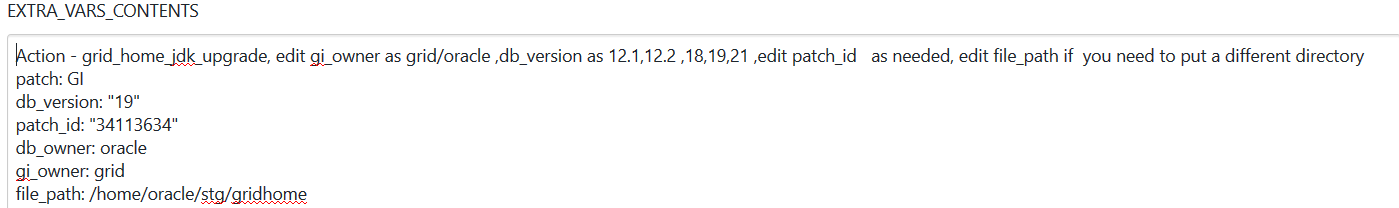
Please change “gi\_owner” based on either oracle or grid who owns the grid software

1. **Action: grid\_home\_jdk\_upgrade**

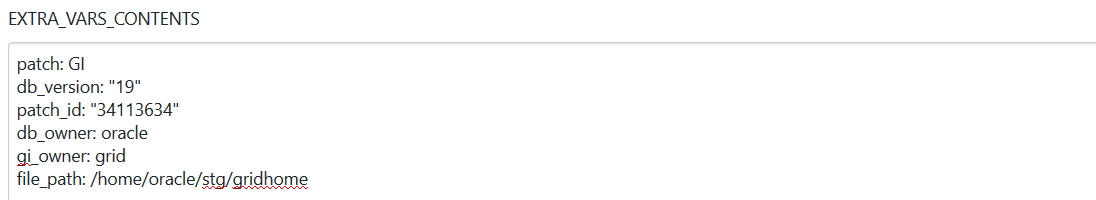
Note: Parameters “Action, INVENTORY\_FILE, LIMIT\_HOST, EXEC\_ENV” are same for all playbooks, only the EXTRA\_VARS\_CONTENTS change for each playbook.

***For this playbook DOWN TIME needed****.*

EXTRA\_VARS\_CONTENTS



**Final look after editing the above screenshot.**



Example:

**/home/oracle/stg** should be there in the host server. Don’t remove “gridhome”, if you rerun the same playbook in the same host and you want to preserve the old logs from last run you can put something like “gridhome1 or gridhome2 etc…after first run

**Note**: you can have any dir name as staging dir not necessarily “**/home/oracle/stg”** which should be owned by oracle, this a staging dir that is used by pipeline that need to be present in the server otherwise the playbook run will fail in that host.

Please change “gi\_owner” based on either oracle or grid who owns the grid software

1. **Action: oracle\_db\_grid\_jdk\_rollback**

Note: Parameters “Action, INVENTORY\_FILE, LIMIT\_HOST, EXEC\_ENV” are same for all playbooks, only the EXTRA\_VARS\_CONTENTS change for each playbook.

***For this playbook DOWN TIME needed****.*

Please put parameter in EXTRA\_VARS\_CONTENTS as below for DB and GRID JDK rollback as needed …remaining other parameter should not be changed.

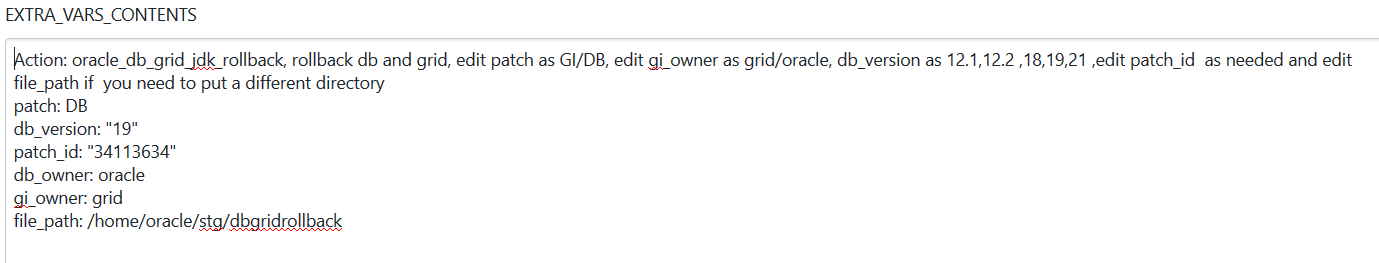
For DB home JDK rollback:

**Patch:DB**

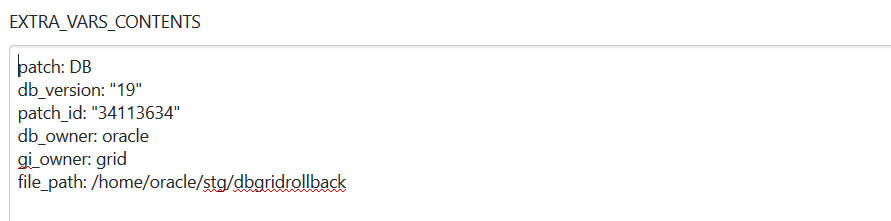
For GRID home JDK rollback:

**Patch:GI**

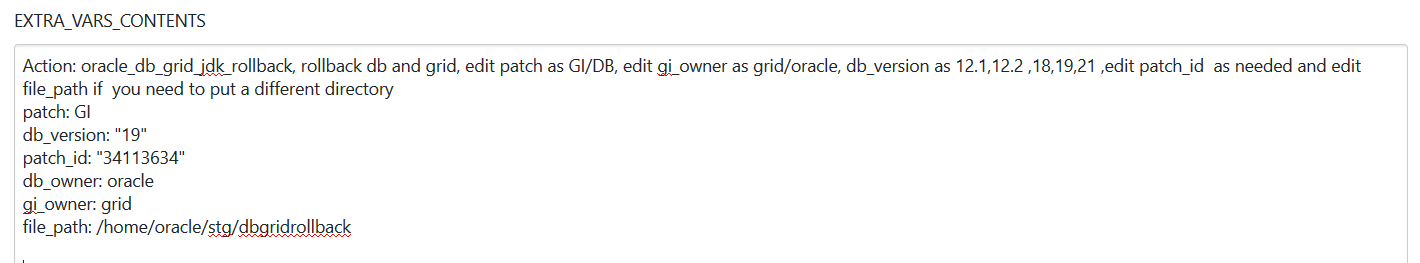
**Example of DB home rollback for JDK patch:**



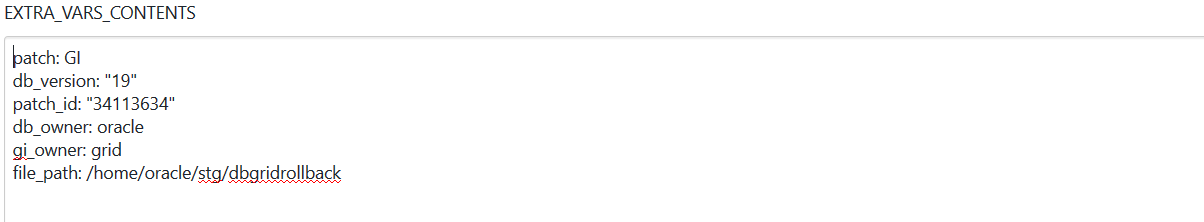
**Final look after editing the above screenshot.**



**Example of GRID home rollback for JDK patch:**



**Final look after editing the above screenshot.**



**Example:**

**/home/oracle/stg** should be there in the host server. Don’t remove “clientjdkupgrade”, if you rerun the same playbook in the same host and you want to preserve the old logs from last run you can put something like “clientjdkupgrade1 or clientjdkupgrade2 etc…after first run

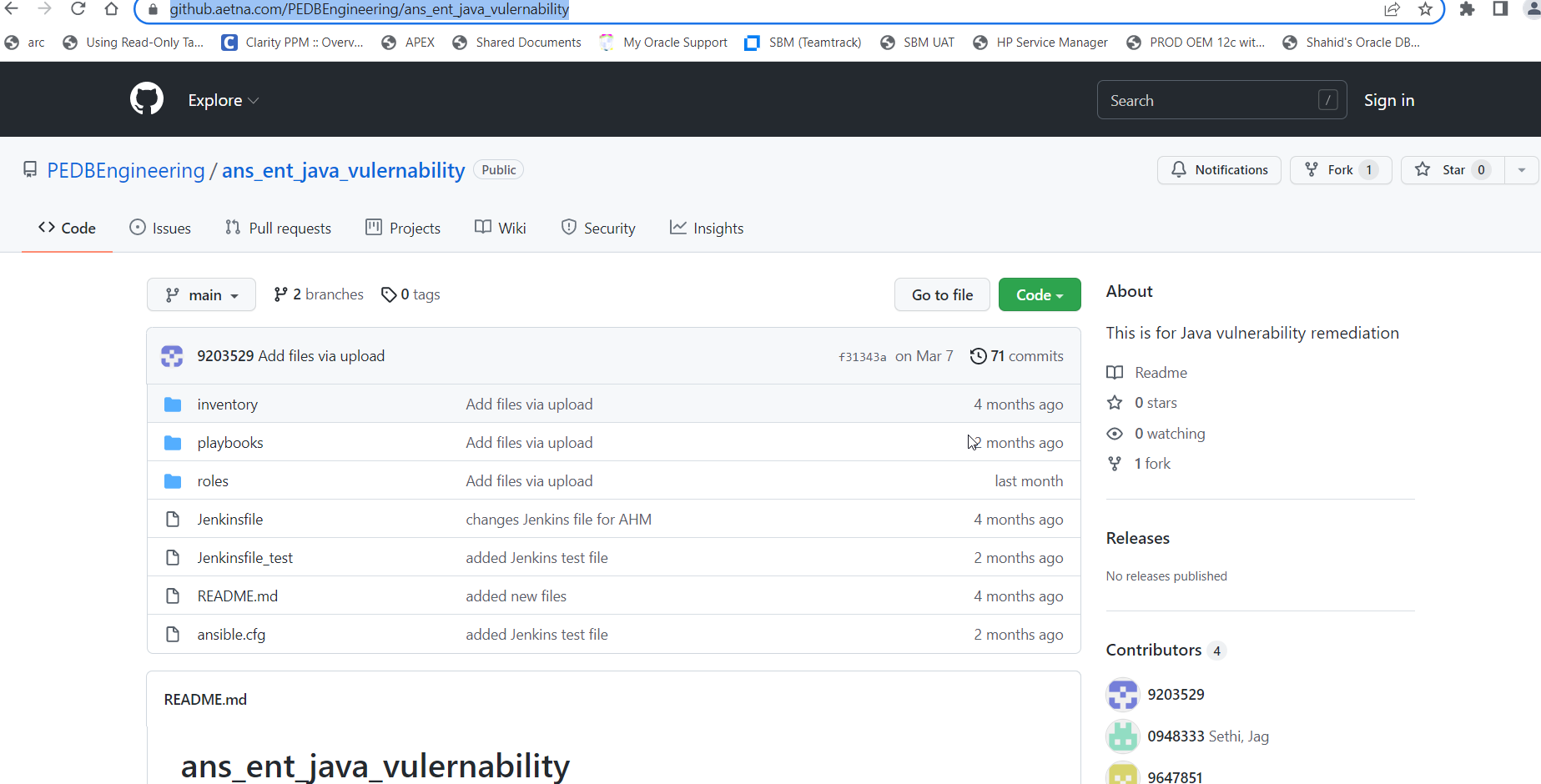
**Note**: you can have any dir name which should be owned by oracle, this a staging dir that is used by pipeline that need to be present in the server otherwise the playbook run will fail in that host.

Please change “gi\_owner” based on either oracle or grid who owns the grid software

**Github location for playbooks for this Pipeline:**

**url:**

[**https://github.aetna.com/PEDBEngineering/ans\_ent\_java\_vulernability**](https://github.aetna.com/PEDBEngineering/ans_ent_java_vulernability)



**Contacts if help needed:**

* *If host missing in your inventory file and you need that to be put in the inventory file*
* *If job failed and you are not able figure out why*

Sethi, Jag [Jagabandhu.Sethi@CVSHealth.com](mailto:Jagabandhu.Sethi@CVSHealth.com)

Suram, Sudhakar [SuramS@aetna.com](mailto:SuramS@aetna.com)

Allen, Mary C. [Mary.Allen@CVSHealth.com](mailto:Mary.Allen@CVSHealth.com)

Sunkara, Sudhakar [Sudhakar.Sunkara@CVSHealth.com](mailto:Sudhakar.Sunkara@CVSHealth.com)

Parekh, Hiten [Hiten.Parekh@CVSHealth.com](mailto:Hiten.Parekh@CVSHealth.com)

Pagala, Sudeep Reddy [SudeepReddy.Pagala@CVSHealth.com](mailto:SudeepReddy.Pagala@CVSHealth.com)

**NOTE:**

* *All the* *relevant logs can be found under “file\_path” dir in the target server after the playbook run is finished or while running.*
* *Make sure to have at least 4 to 5 GB of space available in the “file\_path” directory.*
* *Please don’t run playbook more than 10 hosts at a time as it may be difficult for you to track and make sure the playbook is run for every host that you put in “*LIMIT\_HOST” *section.*
* *You can rerun the playbook for that failed host later after fixing the root cause.*