**Lab 09**

**Laboratory Exercise**

**LAB EXERCISE: Patch Management**

**Reference:** [Configuring patch management (puppet.com)](https://puppet.com/docs/pe/2021.1/patch_management_setup.html)

**What You Learn**

* Use Puppet Enterprise to configure patching node groups, view available operating system patches for your nodes in the console, and apply patches using the pe\_patch::patch\_server task.

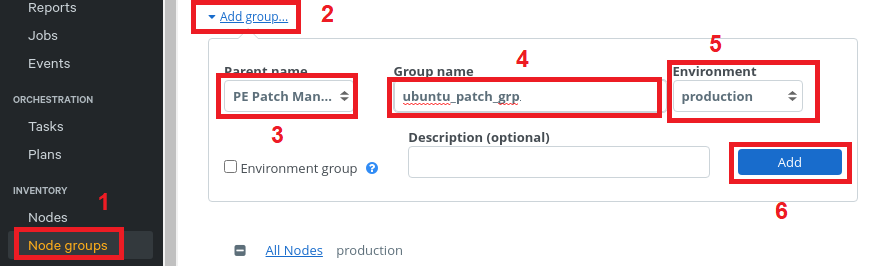
**Time to Complete**

Approximately 45 Minutes

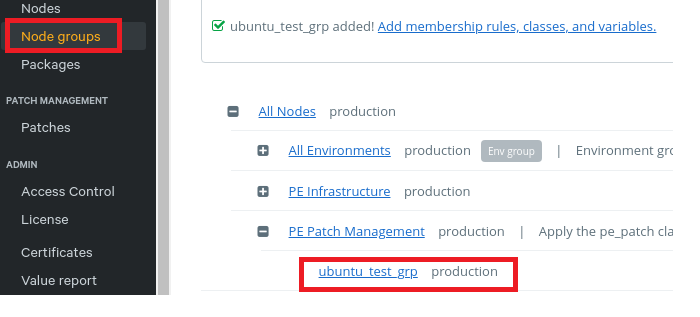
**Part 1: Patch a node group**

Before doing Part 1, need to install package **cron** on agent nodes i.e. puppetclient1.localdomain and puppetclient2.localdomain. You may use **apt-get** or **puppet resource package** command to achieve the above. You may refer to ealier labs if you are not sure of the command.

1. In the PE console, select **Node groups** -> **Add group**.
2. Specify options for the new node group, then select **Add**.
   * Parent name -> ***PE Patch Management****.*
   * Group name – Enter a meaningful name -> ***ubuntu\_patch\_grp***.
   * Environment -> ***production***.
   * Environment group – Do not select.

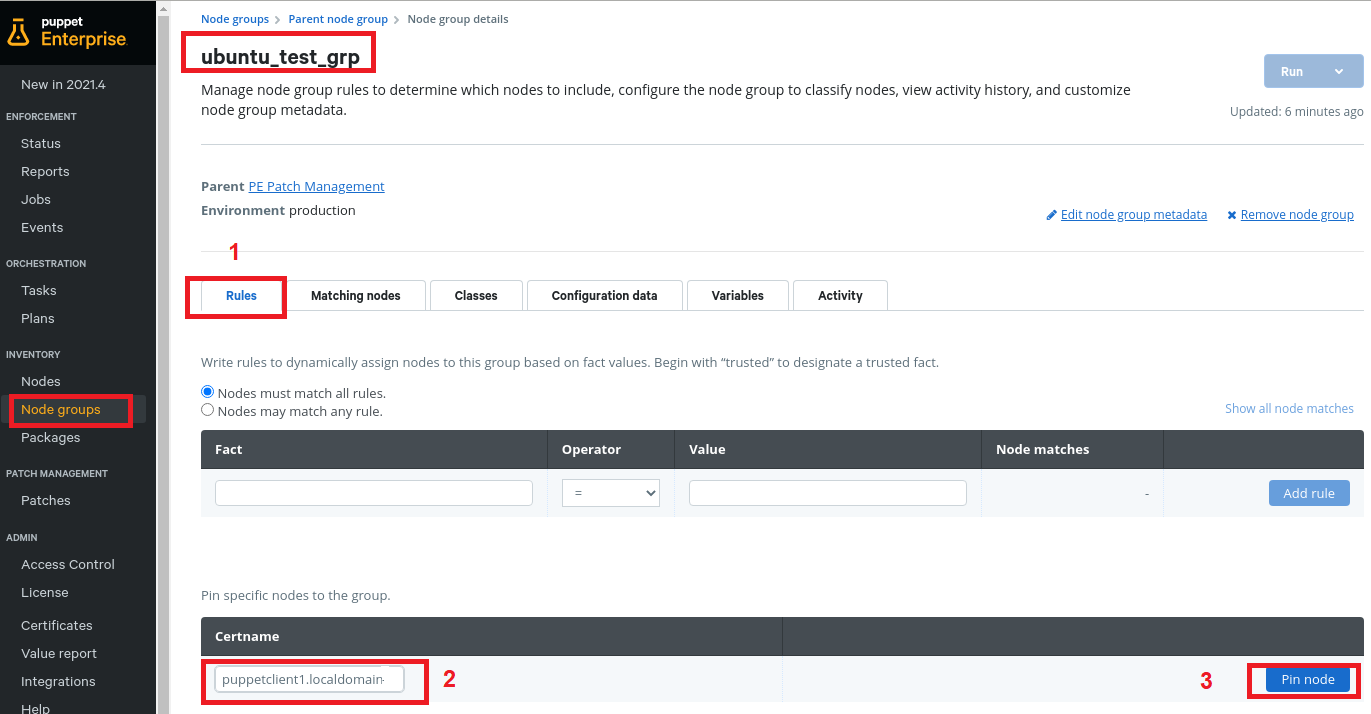


1. Select the patching node group that is just created -> ***ubuntu\_patch\_grp***



1. On the **Node group details** -> **Rules** tab -> add nodes to the group (pin them individually).

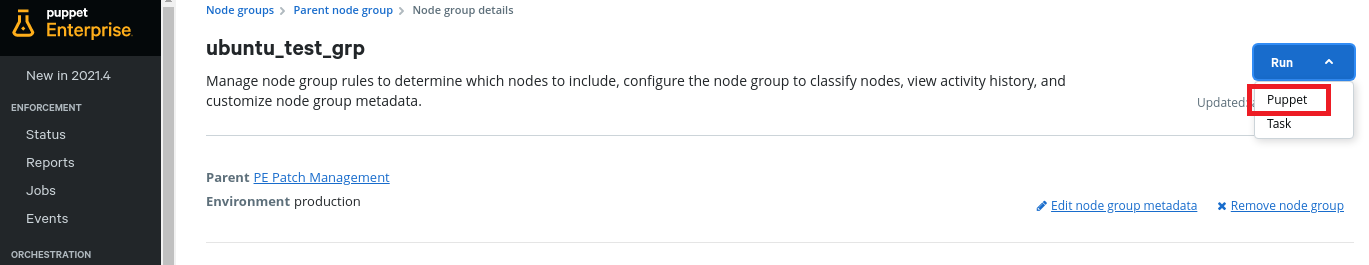
**Do not include the same node in multiple node groups under patch management.**



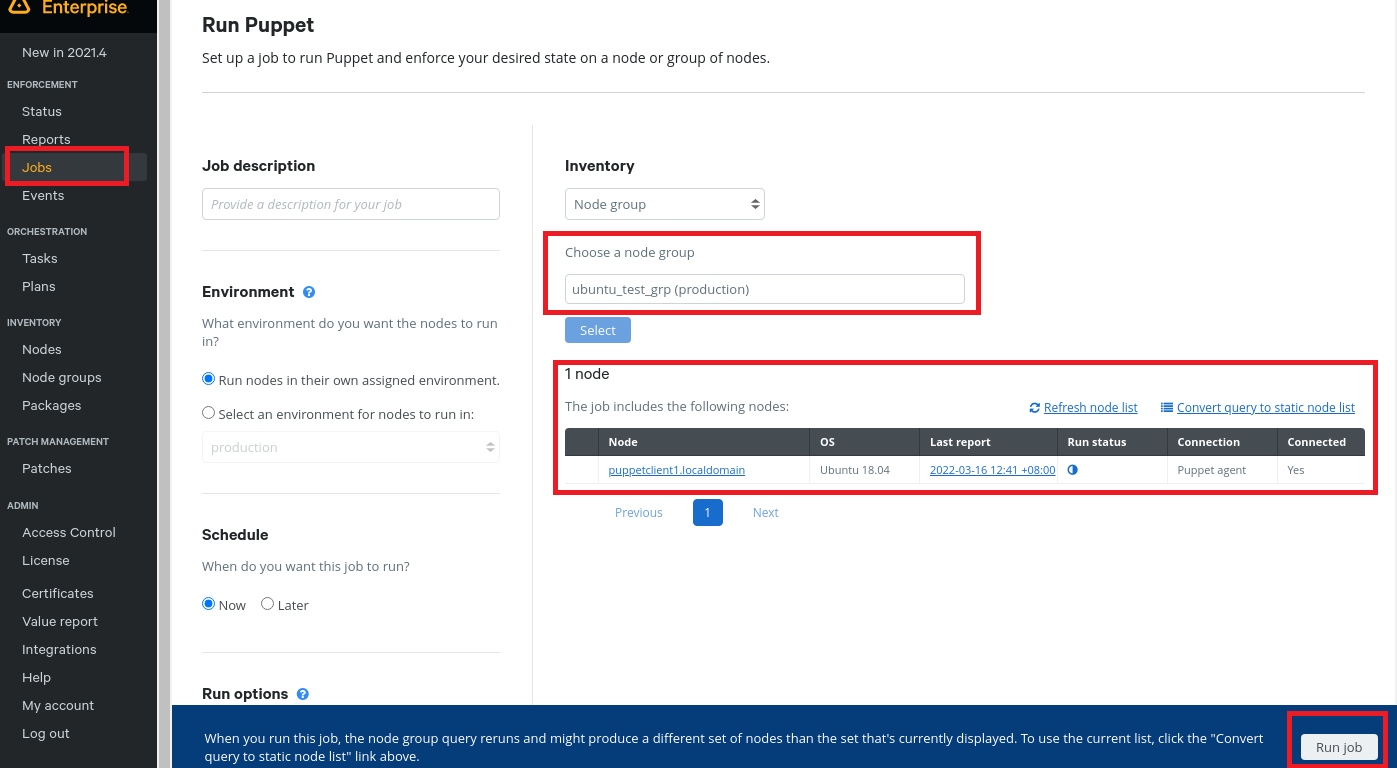
1. Click on Pin node after the node is selected. Then click on **Commit 1 change**

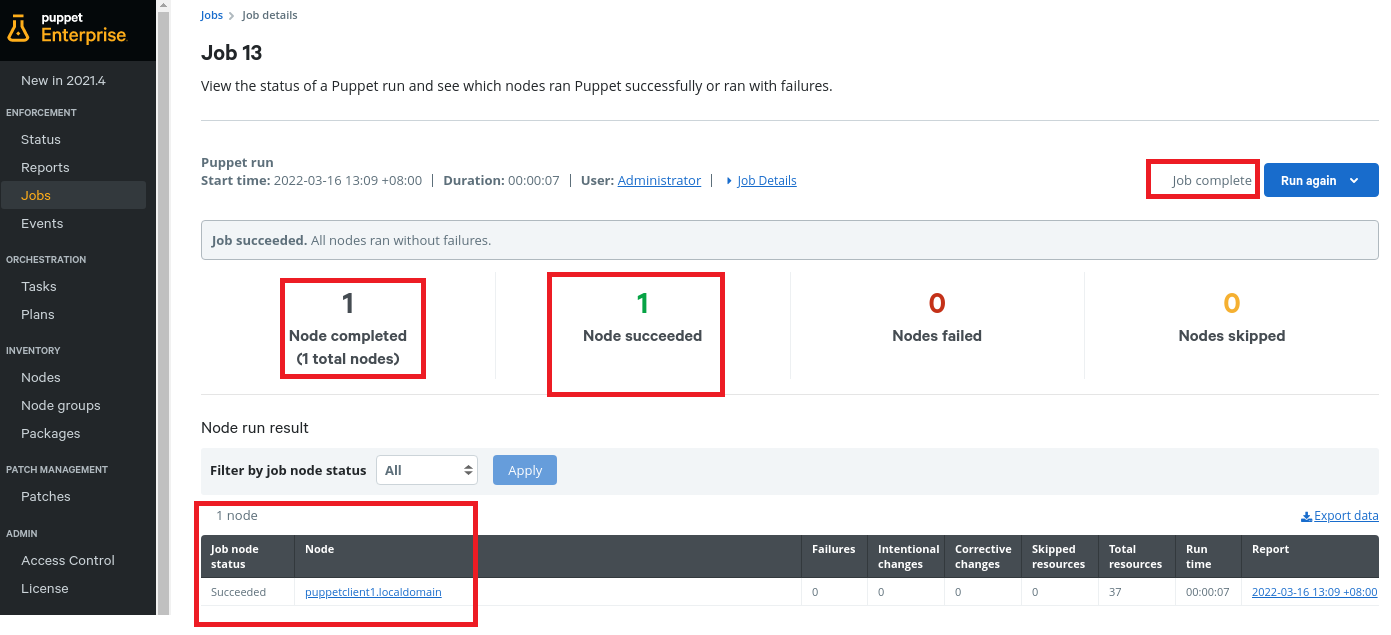


1. At the top right corner of the page, select **Run** -> **Puppet**



1. Click on **Run job**





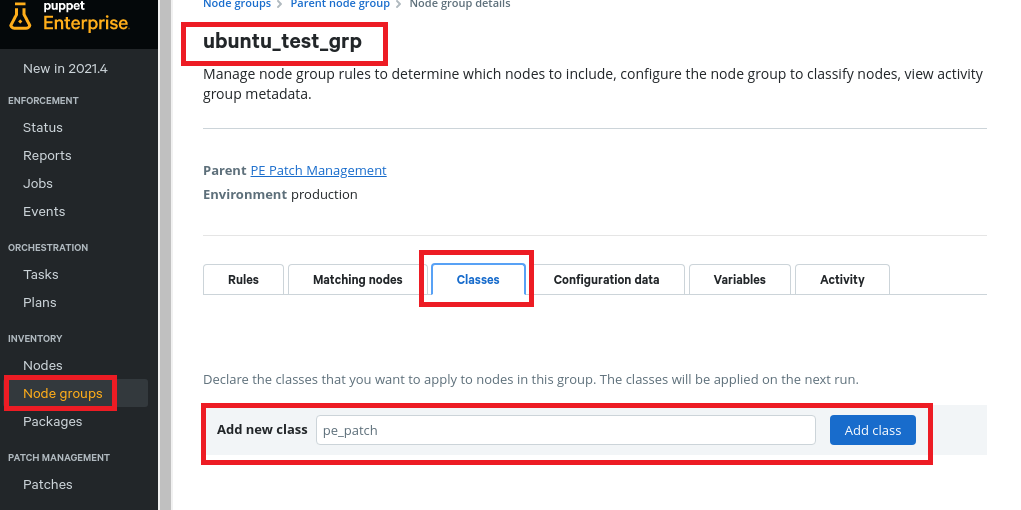
PE can now manage patches for the nodes in your new node group ***ubuntu\_patch\_grp***.

Repeat these steps to add any additional node groups you want under patch management.

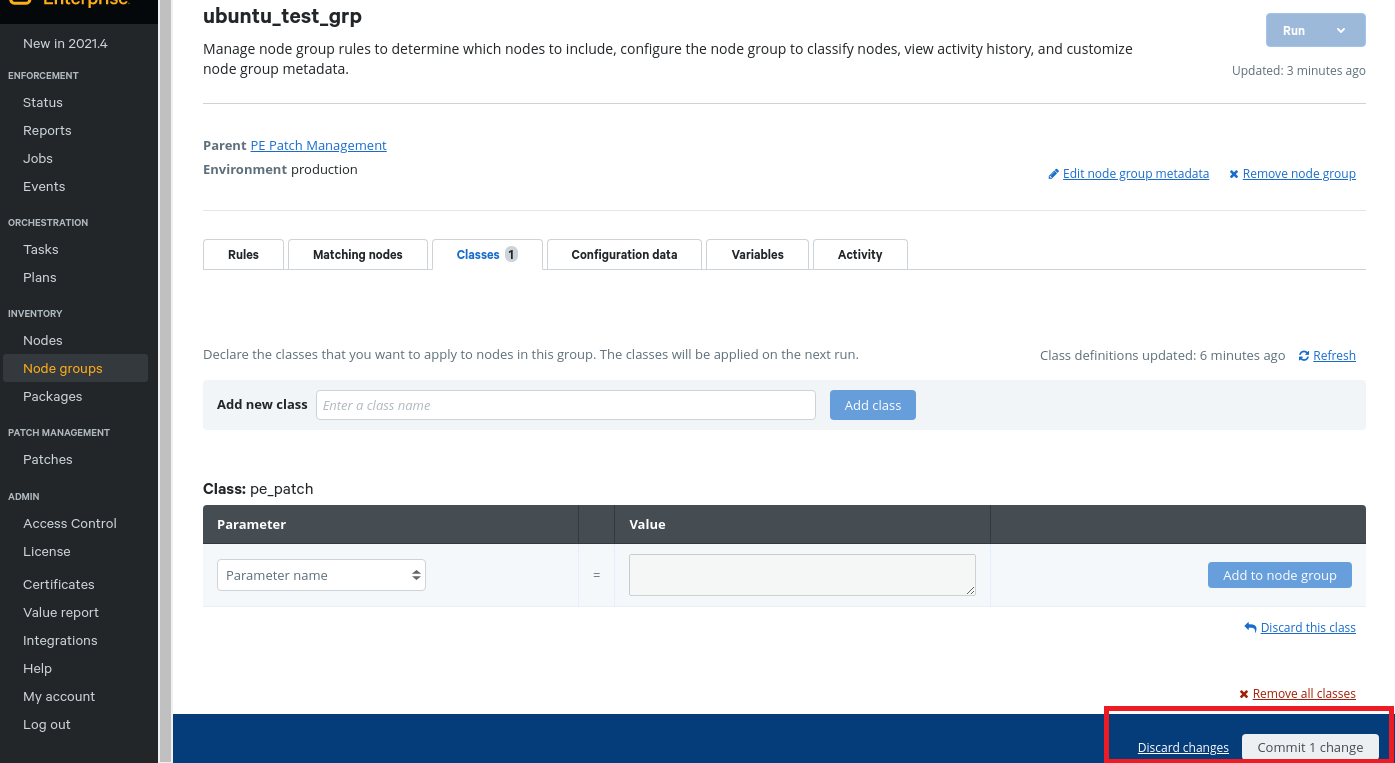
**Specify patching parameters**

Set parameters for node group ***ubuntu\_patch\_grp*** under patch management by first applying the **pe\_patch** class to them, then specifying the desired parameters.

1. On the **Node groups** page, select the ***ubuntu\_patch\_grp*** -> the parameters to be applied to.
2. If it does not already exist, add the **pe\_patch** class to the node group.
   1. On the **Classes** tab, enter **pe\_patch** and select **Add class**.



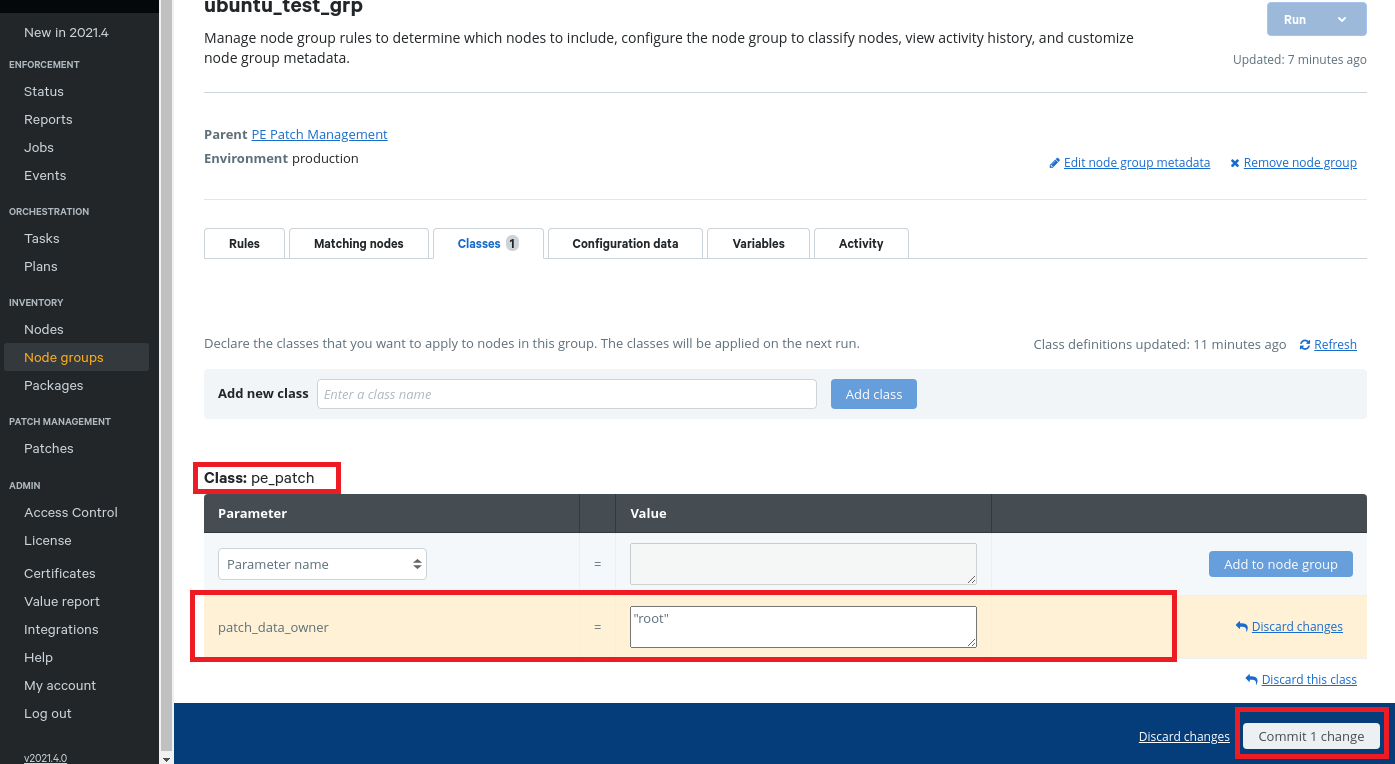
* 1. Commit changes.



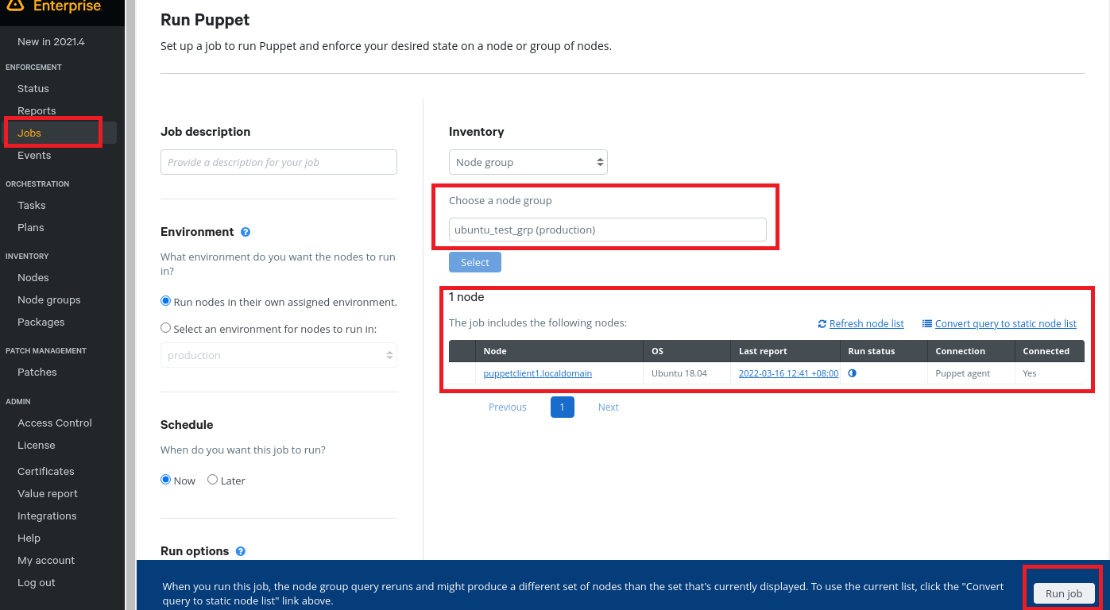
1. In the **pe\_patch** class, add the **patch\_group** parameter and specify a value that describes the nodes in this node group.

**Note:** The **patch\_group** parameter is used to identify which nodes to run patching plans against. You might specify **patch\_group** names that match your node groups, or apply the same **patch\_group** parameter across several patching node groups that have similar characteristics.

1. Specify patching parameters to be applied in the **pe\_patch** class.
2. Commit changes.



Run Puppet on nodes in the node group before running patching tasks or plans.



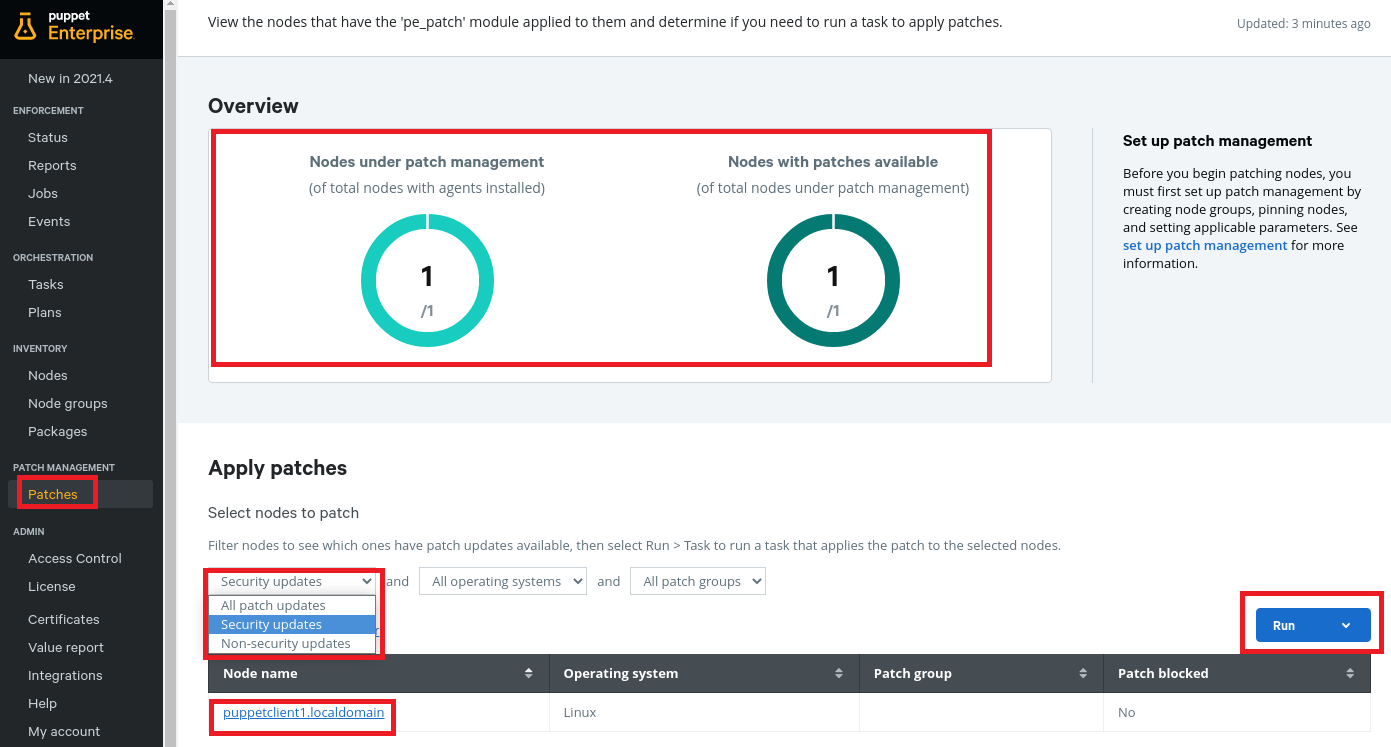
For patch parameters, refer to <https://puppet.com/docs/pe/2019.8/patch_management_setup.html#config_patch_management>

## **Now, we are ready to patch the puppet node**

Use the patch\_server task to apply patches to nodes. You can limit patches to security or non-security updates, Windows or \*nix nodes, or a specific patch group.

**Before you begin**

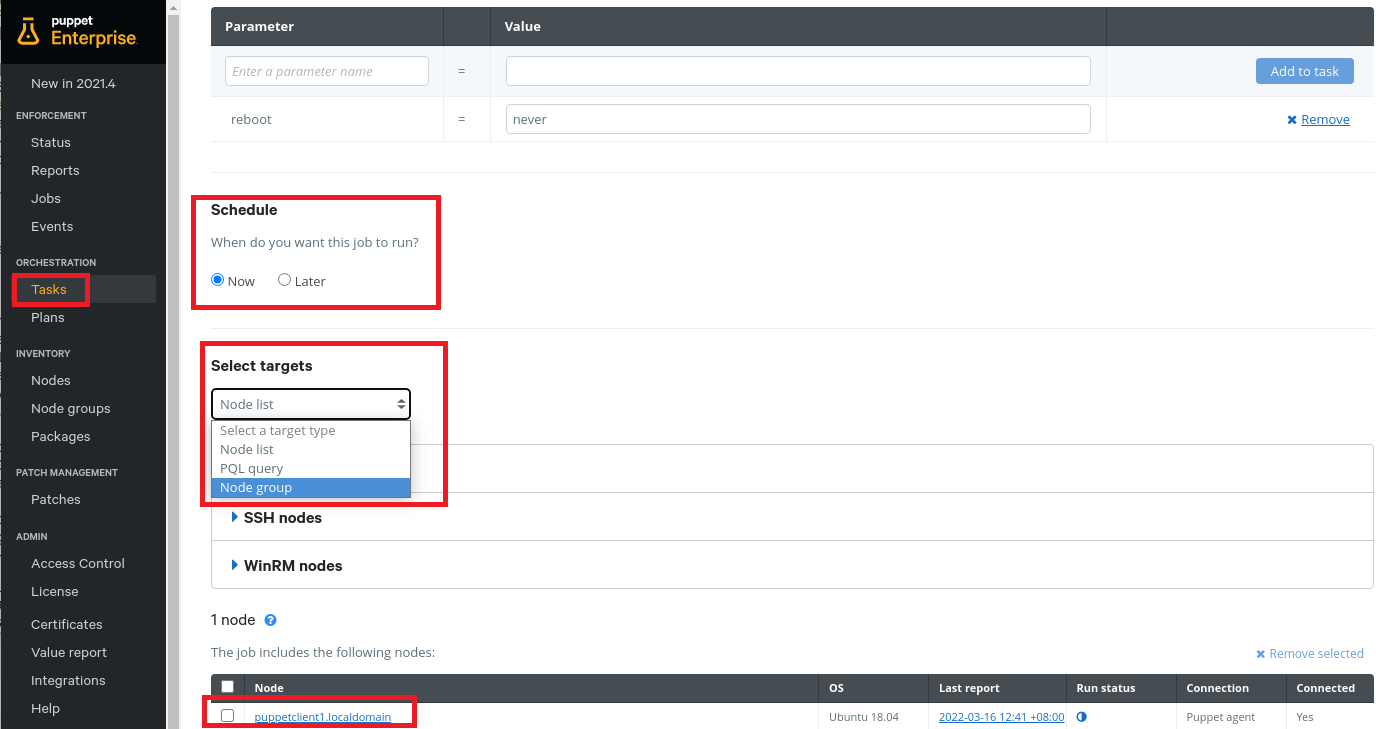
1. On the **Patches** page, in the **Apply patches** section, use the filters to specify which patches to apply to which nodes. For example, if **Security updates** and **Linux** are selected, the results include security patches for Linux nodes, not all security patches and all Linux patches.



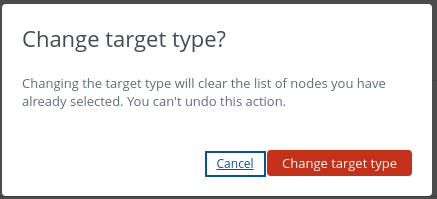
1. Select **Run** -> **Task**.

The **Run a task** page appears with patching information pre-filled for the pe\_patch::patch\_server task.

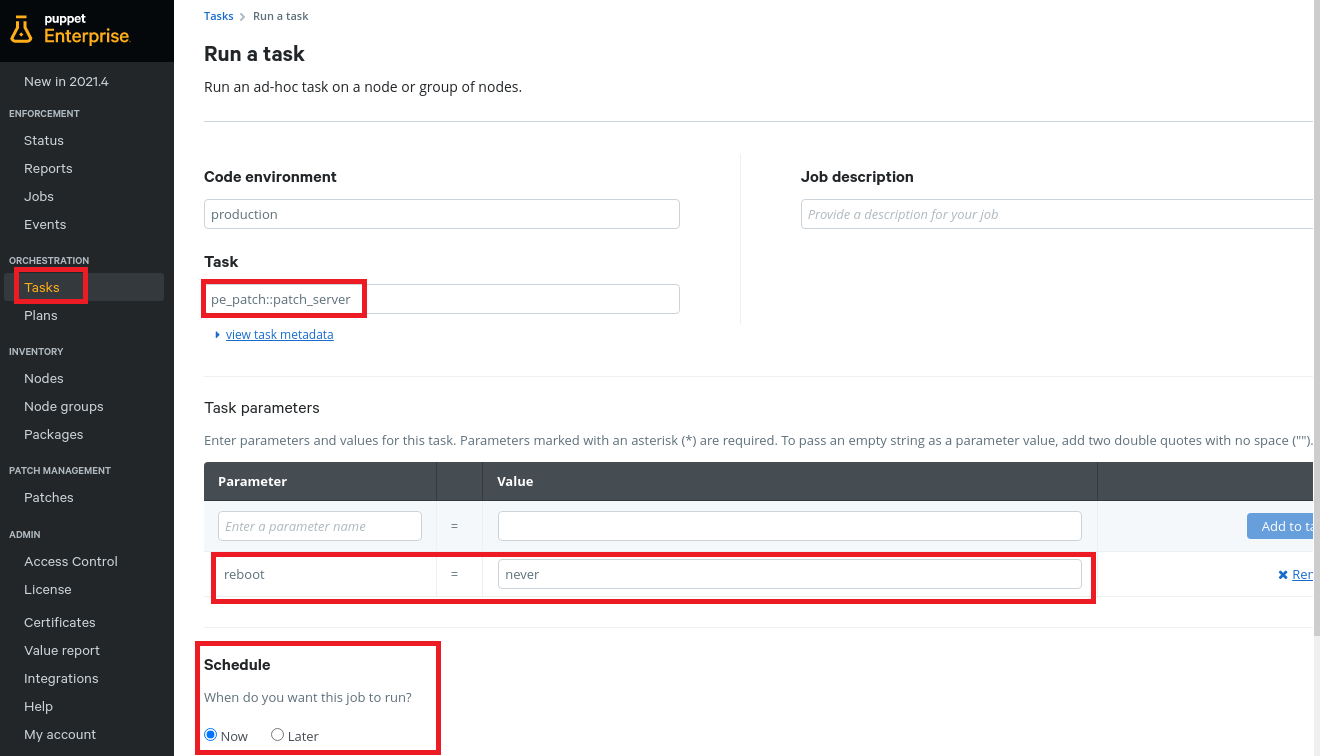
1. **Optional:**In the **Job details** field, provide a description of the task run. This text appears on the **Tasks** page.
2. **Optional:**Under **Task parameters**, add optional parameters to the task. For a full list of available parameters, refer to <https://puppet.com/docs/pe/2021.2/patch_management_patching_nodes.html#config_patch_task>
3. **Optional:**If you want to schedule the task to run later, under **Schedule**, select **Later** and choose a time.
4. Select **Run task** to apply patches.

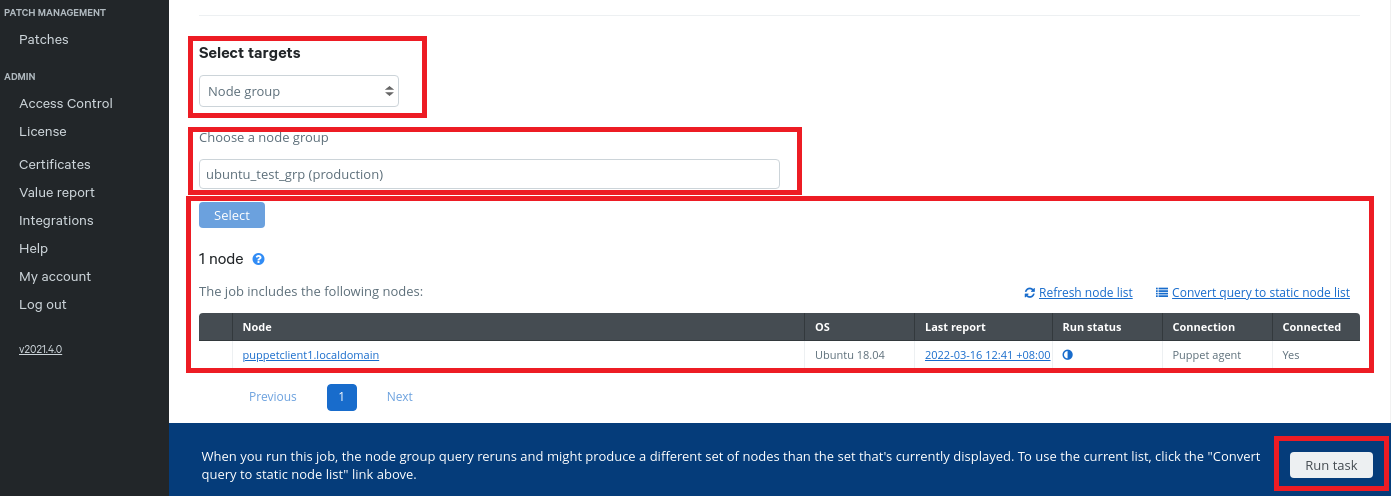


1. When prompt to change target type, click on **Change target type**.

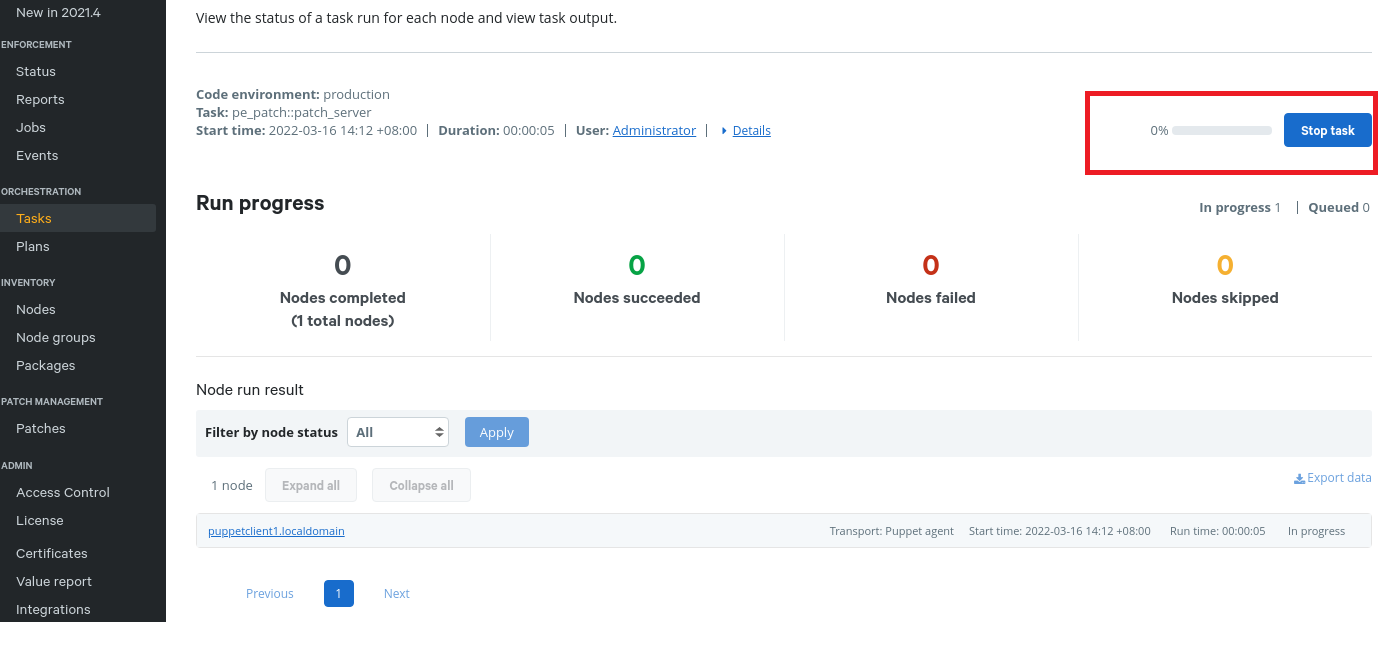


1. Check the settings

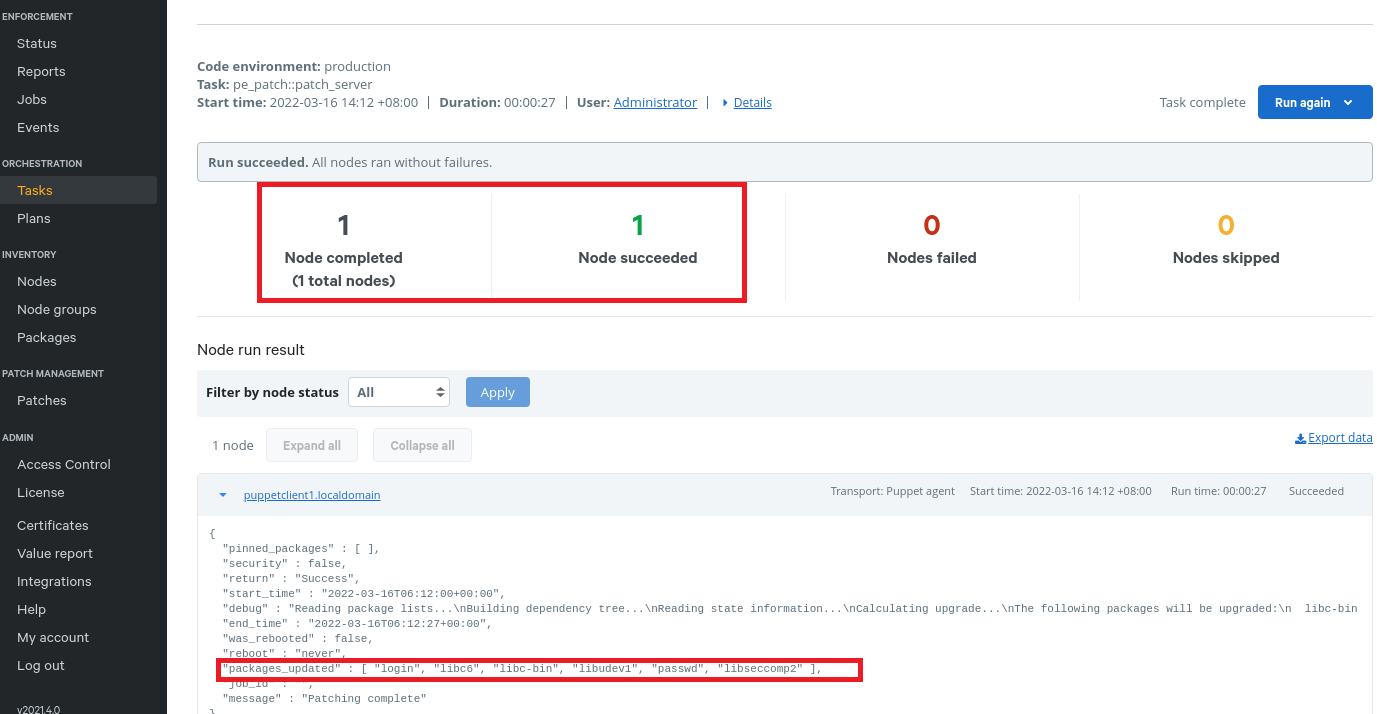
****

****

1. Click on **Run task**

****

10 The patching tasks is successfully completed.

****

**Do this:**

You have added **puppetclient1.localdomain** to patch management. Now, follow **step 3 to step 10**, add the agent node **puppetclient2.localdomain** to the patch management. After which, patch security patch for puppetclient2.localdomain.

### **Part 2 Patch management blackout window**

Blackout window can be configured to node groups. During the blackout window, patch activities will not be able to run.

1. On the **Node groups** page -> ***ubuntu\_patch\_grp***
2. On the **Class: pe\_patch** tab, under **Parameter**, add the **blackout\_windows** parameter
3. In the **Value** field, enter the blackout window as a JSON hash of keys and an ISO compliant timestamp.

For example, set a blackout window from 16 March 2022 SGT 00:00 to 17 March 2022 SGT 00:00 looks like this:

{

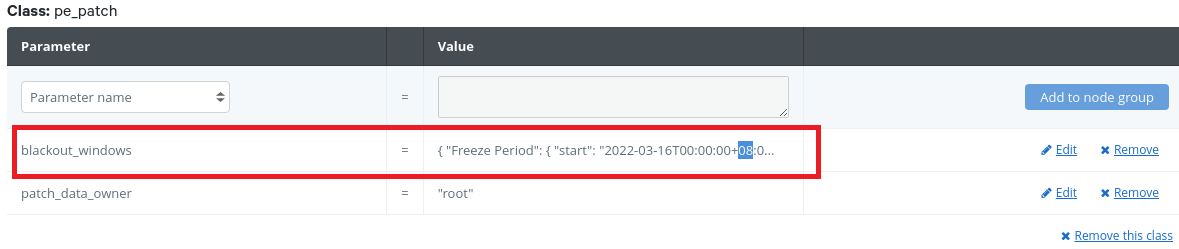
“Freeze Period”: {

“start”: “2022-03-16T00:00:00+08:00”,

“end”: “2022-03-17T00:00:00+00:00”

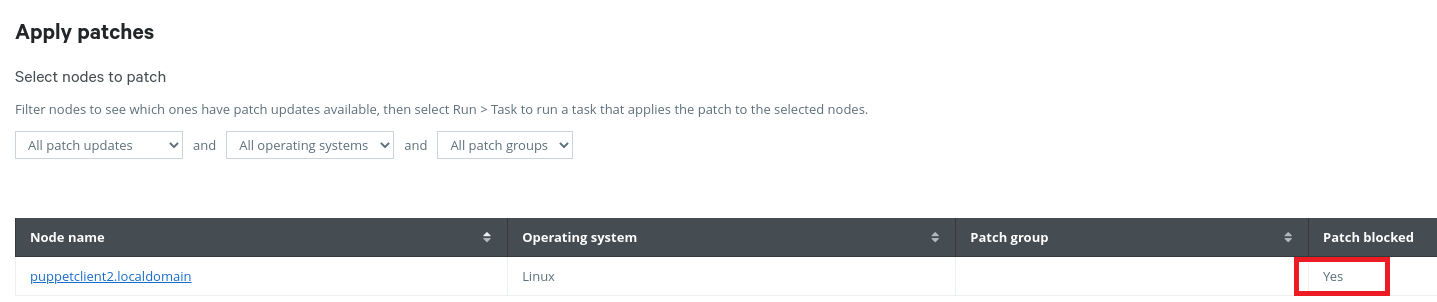
}

}



Note: the blackout windows apply against the time of client and not that of puppet master.

1. Commit changes.
2. Notice **Patch Blocked -> Yes**

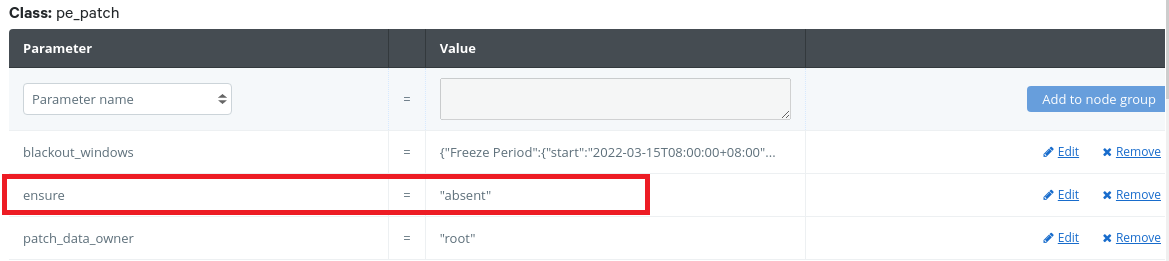


## **Part 3 Disable patch management**

There are 2 ways to disable patch management:

* Use the console to disable patch management by editing the ensure parameter in the **PE Patch Management** node group.
* Remove patch management by deleting patching node groups.

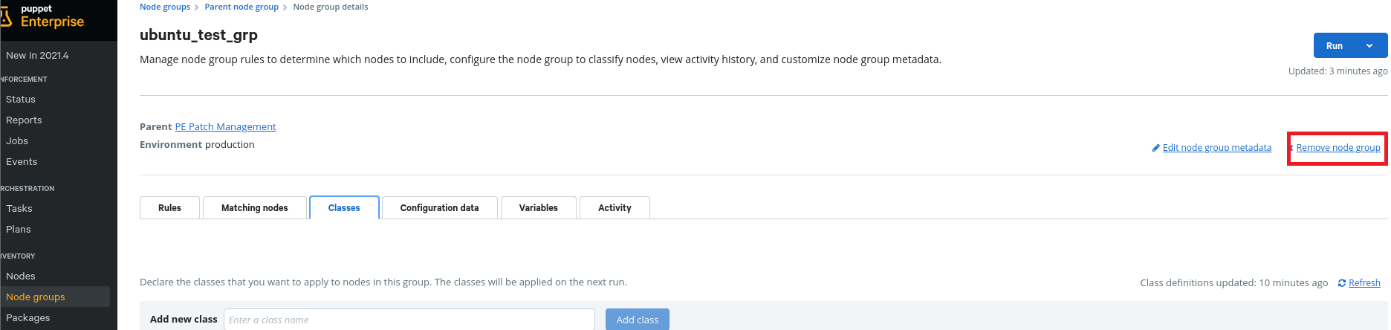
1. In the console, click **Node groups** and select the **PE Patch Management** node group i.e. ***ubuntu\_patch\_grp***.
2. On the **Class** tab, under the **pe\_patch** class, select the ***ensure*** parameter, and change the value to ***absent***.



1. Click **Add to node group** and commit the change.
2. Run Puppet.

The client components of the pe\_patch class, like cron and scripts, are removed from PE.

1. **Alternatively:**To remove patch management from your infrastructure, click **Remove node group** on the **Node details** page for the **PE Patch Management** node group.



The **Patch Management** in the console sidebar remains active after disabling patch management. However, the **Patches** page no longer reports patch information.