**Setting Up a Trigger for a Pipeline Using a Generic WebHook Trigger**

**Step 1: Install Generic Webhook Trigger Plugin**

1. Go to Jenkins dashboard.
2. Navigate to **Manage Jenkins** -> **Manage Plugins**.
3. In the **Available** tab, search for **Generic Webhook Trigger**.
4. Install the plugin and restart Jenkins if necessary.

**Step 2: Configure Jenkins Job**

1. Create or open the Jenkins job you want to trigger.
2. In the job configuration, scroll down to the **Build Triggers** section.
3. Check the checkbox for **Generic Webhook Trigger**.

**Step 3: Configure Post Parameters**

1. In the **Post Parameters** section, add the following:
   * **Variable**: ref
   * **Expression (JSON)**: $.ref

**Step 4: Write a String as Token**

1. Write any keyword string as token that will be used in the webhook URL (e.g., github\_token).

**Step 5: Configure Optional Filter**

1. In the **Optional Filter** section, configure the following:
   * **Expression**: refs/heads/branch\_name (Replace branch\_name with the actual branch name)
   * **Text**: $ref

**Step 6: Configure GitHub Webhook**

1. Go to your GitHub repository **Settings**.
2. Navigate to **Webhooks**.
3. Click **Add webhook**.
4. In the **Payload URL** field, enter:
5. http://<Jenkins\_URL>:8080/generic-webhook-trigger/invoke?token=<github\_token>

Replace <Jenkins\_URL> with your Jenkins server URL and <github\_token> with the token specified earlier.  
Example:

http://65.0.31.109:8080/generic-webhook-trigger/invoke?token=github\_token

1. Set the **Content type** to application/json.
2. Select the events you want to trigger the webhook for (e.g., **Push events**).

**Step 7: Save Changes**

1. Save your Jenkins job configuration.

Now, your Jenkins job is configured to trigger via a generic webhook whenever a push event occurs on the specified branch in your GitHub repository.