**Jenkins Git Push Automation using GitHub Hook Trigger for GITScm Polling**

**Overview**

This document provides a step-by-step guide on how to configure Jenkins to automatically trigger a build whenever code is pushed to a GitHub repository using **GitHub Webhooks** and **GITScm Polling Trigger**.

**1. Prerequisites**

* A **Jenkins server** running and accessible.
* A **GitHub repository** with Jenkins integration.
* Installed **Jenkins Plugins**:
  + **Git Plugin**
  + **GitHub Integration Plugin**
* A **Jenkins job or pipeline** configured for your repository.
* **Jenkins Webhook URL** must be accessible if hosted externally.

**2. Configure GitHub Webhook**

1. Go to your **GitHub repository**.
2. Navigate to **Settings** → **Webhooks**.
3. Click **Add webhook**.
4. In **Payload URL**, enter:
5. http://your-jenkins-server/github-webhook/
   * If Jenkins is hosted on **AWS EC2**, replace your-jenkins-server with **your public EC2 IP** or domain:
   * http://<EC2-Public-IP>:8080/github-webhook/
   * If Jenkins is behind a firewall, use a **reverse proxy or ngrok**.
6. **Content Type:** Select application/json.
7. **Trigger:** Select **Just the push event**.
8. Click **Add webhook**.

**3. Configure Jenkins Job**

**Option 1: Freestyle Job**

1. Open **Jenkins Dashboard**.
2. Click **New Item** → **Freestyle Project**.
3. In **Source Code Management**, select **Git**.
4. Enter your **GitHub repository URL**.
5. Under **Credentials**, select your GitHub credentials.
6. **Branch to build** → Enter main (or your branch name).
7. In **Build Triggers**, select: ✅ **GitHub hook trigger for GITScm polling**
8. Click **Save**.

**Option 2: Pipeline Job**

1. Go to **Jenkins Dashboard** → **New Item** → **Pipeline**.
2. In **Pipeline definition**, select **Pipeline script from SCM**.
3. Choose **Git** and enter your **GitHub repository URL**.
4. **Branch to build:** main (or your preferred branch).
5. In **Build Triggers**, check: ✅ **GitHub hook trigger for GITScm polling**
6. Click **Save**.

**4. Test the Setup**

1. Push a change to your GitHub repository:
2. git add .
3. git commit -m "Trigger Jenkins build"
4. git push origin main
5. Go to **GitHub → Settings → Webhooks**.
6. Click on your webhook and check if the request was **delivered successfully**.
7. Go to **Jenkins Dashboard** → Open your job → Check if the build was triggered.

**5. Troubleshooting**

| **Issue** | **Solution** |
| --- | --- |
| Webhook request fails | Check Jenkins URL, make sure it's publicly accessible |
| Build not triggering | Ensure "GitHub hook trigger for GITScm polling" is checked |
| Jenkins not receiving webhook | Restart Jenkins and test manually using curl |
| Permissions issue | Ensure Jenkins has access to the GitHub repository |

**6. Conclusion**

By following these steps, you have successfully **configured Jenkins to trigger a build automatically** whenever you **push code to GitHub** using the **GitHub Webhook and GITScm Polling Trigger**.