# DevOps Shack

Sending Email Notifications To OutLook With Jenkins

Part 1: Outlook Configuration

You need to set up your **Outlook account** to send emails using Jenkins.

Step 1: Ensure Outlook Account is Enabled for SMTP

1. **Check if SMTP is enabled**: Outlook uses the smtp.office365.com server for sending emails. If you're using a personal account (like @outlook.com), **SMTP is already enabled by default**.

2. **Enable POP & IMAP**: Go to Settings icon on right-top & select Email→Forwarding & IMAP →SignIn
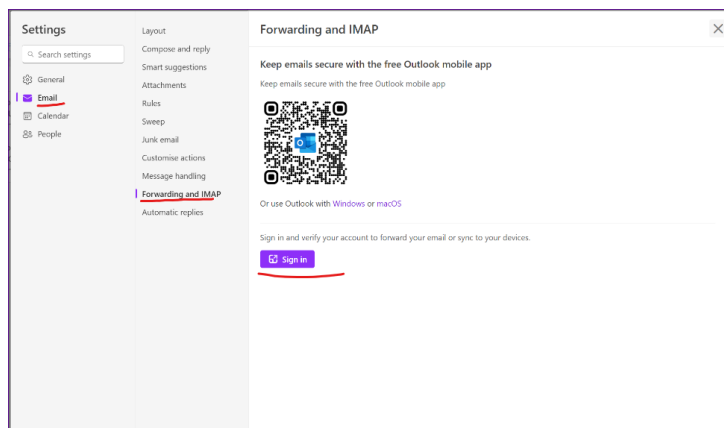
3. Enable POP & IMAP
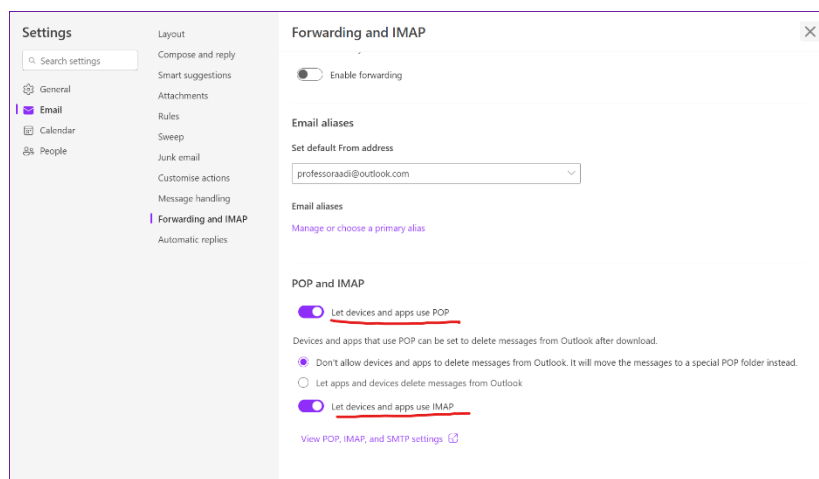
4. **Set up App Passwords** (if you have 2-factor authentication enabled):

   - Go to the [Microsoft Security](https://account.microsoft.com/) [https://account.microsoft.com/] page.

   - Under **Security basics**, select **Advanced security options**.

   - Go to **Additional Security** options.

   - Scroll down to **App passwords** and click **Create a new app password**.

   - Save the app password, as you'll use it in Jenkins.

Step 2: Verify SMTP Server Settings [Just Note down the details]

- **SMTP Server**: smtp.office365.com

- **Port**: 587

- **Encryption**: TLS

Part 2: Jenkins Configuration

Now that your Outlook account is set up, follow these steps to configure Jenkins.

Step 1: Configure Email Notifications in Jenkins

1. **Log into Jenkins**.

2. Navigate to **Manage Jenkins → Configure System**.

3. Scroll down to the **E-mail Notification** section and configure the SMTP settings:

   - **SMTP Server**: smtp.office365.com

   - **SMTP Port**: 587

   - **Check** the box for **Use SMTP Authentication**.

   - **Username**: professoraadi@outlook.com (your Outlook email address).

   - **Password**: Enter your Outlook password or App Password (if 2FA is enabled).

   - **Use TLS**: Check this option.

   - **Reply-To Address**: professoraadi@outlook.com.

4. **Save the configuration**.

Step 2: Configure Extended Email Notification (Email-Ext Plugin)

1. Ensure the **Email Extension Plugin** is installed. If not:

   - Go to **Manage Jenkins → Manage Plugins**.

   - Under **Available Plugins**, search for Email Extension Plugin, install it, and restart Jenkins if needed.

2. After installing the plugin, go to **Manage Jenkins → Configure System**.

3. Scroll to the **Extended E-mail Notification** section and configure:

    o **SMTP Server**: smtp.office365.com.

    o **Default User E-mail Suffix**: @outlook.com.

    o **Use SMTP Authentication**: Enable it.

    o **Username**: professoraadi@outlook.com.

    o **Password**: Your Outlook password or App Password.

    o **Use TLS**: Enable it.

    o **From Address**: Set to professoraadi@outlook.com.

    o **Reply-to Address**: professoraadi@outlook.com.

4. Save the configuration.

Step 3: Test Email Configuration

1. Scroll down in the **E-mail Notification** section.

2. **Test configuration by sending a test e-mail**:

    o Enter a valid recipient email (e.g., professoraadi@outlook.com).

    o Send the test email. If successful, the configuration is correct.

Part 3: Jenkins Pipeline Example

Here's the Jenkins pipeline script you provided, complete with the email notification functionality:

```
pipeline {
   agent any

   stages {
      stage('Build') {
         steps {
            echo 'Building...'
            // Your build steps go here
         }
      }

      stage('Test') {
         steps {
            echo 'Testing...'
            // Your test steps go here
         }
      }

      stage('Deploy') {
         steps {
            echo 'Deploying...'
```

```groovy
                // Your deployment steps go here
            }
        }
    }

    post {
        always {
            script {
                def jobName = env.JOB_NAME
                def buildNumber = env.BUILD_NUMBER
                def pipelineStatus = currentBuild.result ?: 'UNKNOWN'
                def bannerColor = pipelineStatus.toUpperCase() == 'SUCCESS' ? 'green' : 'red'

                def body = """
                    <html>
                    <body>
                    <div style="border: 4px solid ${bannerColor}; padding: 10px;">
                    <h2>${jobName} - Build ${buildNumber}</h2>
                    <div style="background-color: ${bannerColor}; padding: 10px;">
                    <h3 style="color: white;">Pipeline Status: ${pipelineStatus.toUpperCase()}</h3>
                    </div>
                    <p>Check the <a href="${env.BUILD_URL}">console output</a>.</p>
                    </div>
                    </body>
                    </html>
                """

                emailext (
                    subject: "${jobName} - Build ${buildNumber} - ${pipelineStatus.toUpperCase()}",
                    body: body,
                    to: 'professoraadi@outlook.com',
                    from: 'professoraadi@outlook.com',
                    replyTo: 'professoraadi@outlook.com',
                    mimeType: 'text/html'
                )
            }
        }
    }
}
```

Explanation of the Pipeline

- **Pipeline Stages**: The pipeline has three stages: Build, Test, and Deploy. You can replace the echo commands with actual build, test, and deployment scripts.

- **Post Section**: After every pipeline run, regardless of the result, the post section will send an email notification. This uses the emailext step from the **Email-Ext** plugin.

- **Dynamic Email Body**: The HTML email body includes dynamic values such as:

  - **Pipeline status** (Success or Failure) that changes the color of the banner.

  - A link to the **Jenkins console output**.

- **Email Settings**: The email is sent from professoraadi@outlook.com to the same address for testing purposes. You can replace the to field with other recipients.

Part 4: Run the Pipeline

1. Go to Jenkins Dashboard and create a new pipeline job.

2. Add the above pipeline script.

3. Save and run the pipeline.

4. After the pipeline runs, check your Outlook inbox for the email notification with the pipeline results.

Troubleshooting:

1. **SMTP Authentication Error**:

   o Ensure that your Outlook password or App password is correct.

   o Verify that the **SMTP Server** and **Port** are correct.

2. **Email Not Sent**:

   o Check Jenkins logs for errors.

   o Ensure that Outlook SMTP settings are properly configured.

   o If using App passwords, make sure it is correctly entered in Jenkins.

By following these steps, you should have Jenkins successfully configured to send email notifications via your Outlook account. Let me know if you encounter any issues during the setup!