😊 Mend Bolt Tool

Mend Bolt (formerly WhiteSource Bolt) is an automated security and compliance tool designed for Azure DevOps. It scans open-source dependencies in a project to identify security vulnerabilities, outdated libraries, and licensing issues. Mend Bolt helps development teams maintain secure and compliant software by providing actionable insights directly within Azure DevOps.

Key Features

- **Automated Security Scanning**: Continuously scans open-source components to detect known vulnerabilities.
- **License Compliance Monitoring**: Identifies open-source licenses in use and checks for compliance risks.
- **Vulnerability Remediation**: Suggests updated versions or alternative dependencies to mitigate security risks.
- **Integration with CI/CD Pipelines**: Works within Azure DevOps pipelines, allowing security checks to be automated as part of the software development lifecycle.
- **Reports and Alerts**: Provides detailed reports on vulnerabilities, along with severity levels and remediation steps.

Use Cases

1. **Secure Software Development**
    - Developers can integrate Mend Bolt into their Azure DevOps pipelines to ensure that third-party dependencies do not introduce security risks.
    - Automated security scans help catch vulnerabilities early in the development cycle.
2. **License Risk Management**
    - Organizations can prevent legal issues by ensuring that open-source licenses comply with company policies.
    - The tool alerts teams if a dependency has an incompatible or high-risk license.
3. **Continuous Monitoring in CI/CD Pipelines**
    - Teams implementing DevSecOps practices can use Mend Bolt to run security checks automatically during build and release stages.
    - Security issues can be flagged before deployment, reducing production risks.
4. **Compliance with Industry Standards**
    - Companies following security and compliance regulations (e.g., GDPR, ISO 27001) can use Mend Bolt to maintain adherence to best practices.
5. **Risk Prioritization and Management**

- Development teams can prioritize vulnerabilities based on severity and exploitability, focusing on the most critical issues first.

Benefits

- **Proactive Security**: Detects and fixes vulnerabilities before they reach production.
- **Time and Cost Savings**: Reduces manual security audits and lowers the cost of fixing security issues later in development.
- **Enhanced Compliance**: Helps organizations adhere to open-source licensing policies and industry regulations.
- **Seamless Integration**: Works natively with Azure DevOps, allowing teams to incorporate security without disrupting workflows.
- **Actionable Insights**: Provides clear guidance on how to fix security vulnerabilities and update dependencies.

Mend Bolt is a valuable tool for organizations looking to enhance the security of their software supply chain while ensuring compliance with open-source licensing requirements.

Summary

In this lab, we are integrating the **Mend Bolt** security tool into an Azure DevOps pipeline to scan for vulnerabilities in our application. Mend Bolt analyzes dependencies and checks for security risks.

Steps Overview

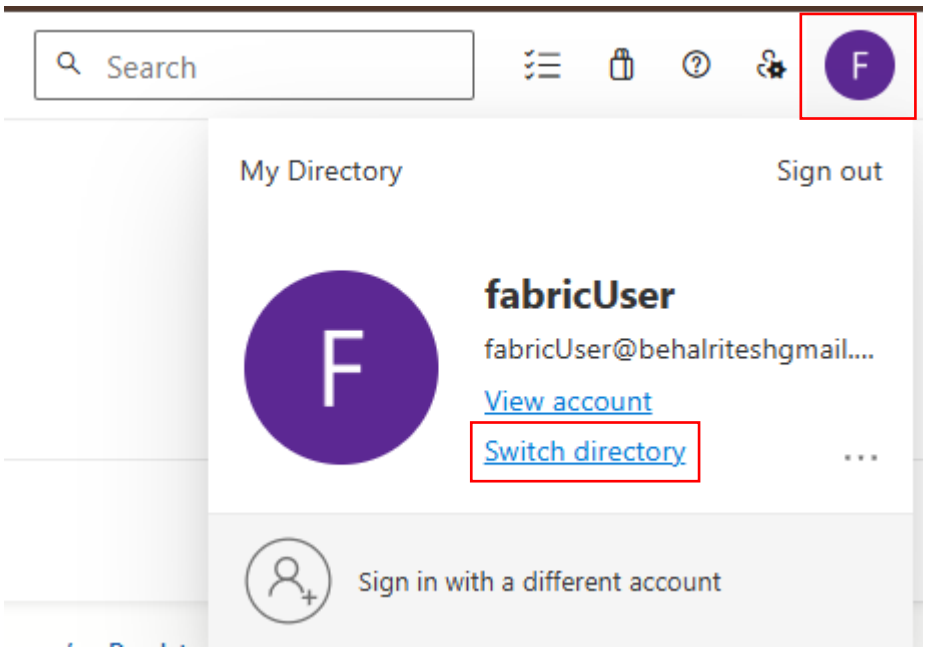
1. **Ensure Azure DevOps is linked to Microsoft Entra ID (Azure AD).**
2. **Install the Mend Bolt extension** from the Azure DevOps Marketplace.
3. **Create a new pipeline** for an application stored in Azure Repos.
4. **Modify the pipeline YAML** to include the Mend Bolt security scan task.
5. **Run the pipeline** and review the Mend Bolt security report.

End Goal

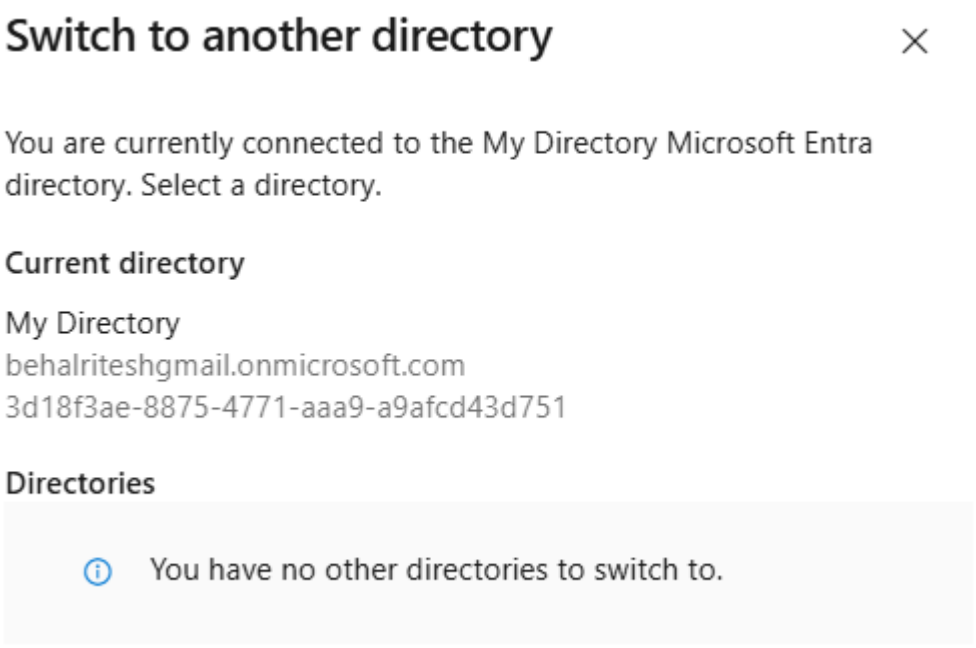
The objective is to **integrate security scanning** into the CI/CD pipeline, ensuring that the application dependencies are **free from vulnerabilities** before deployment.

😄 To begin with the Lab

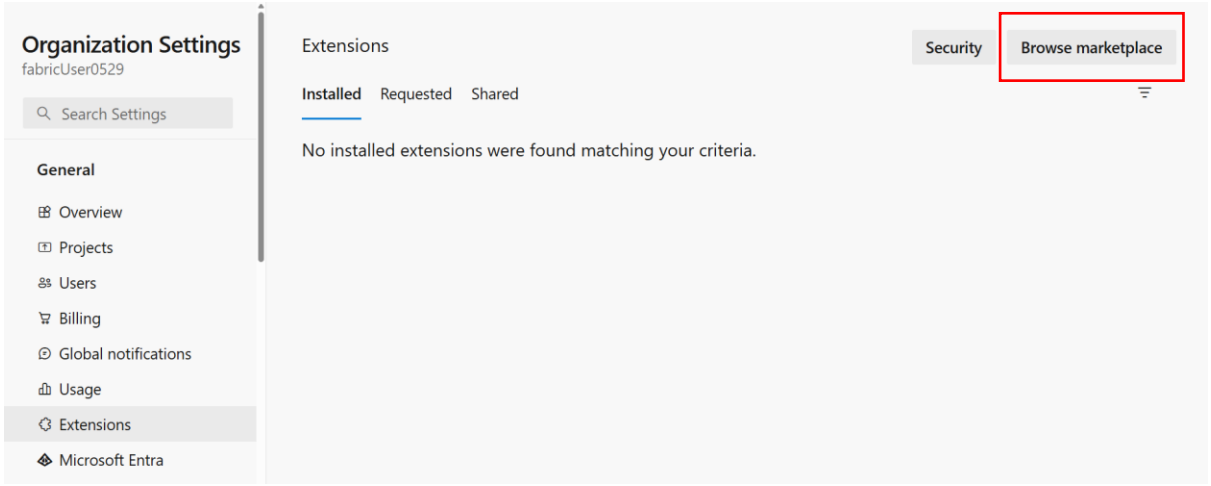
1. Now to use the mend bolt tool, your Azure DevOps account should be linked with your Microsoft Entra ID or Azure Active Directory.
2. Click on your user profile then choose to Switch directory.



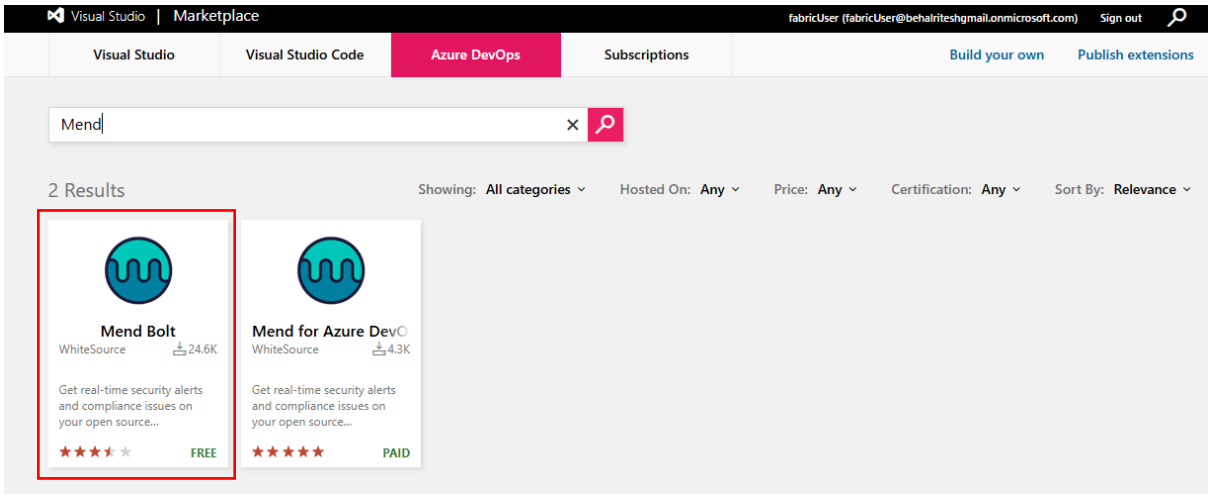
3. Check that your current directory is the same as the Azure active directory. If not then you will see one more directory here. OR you will see a message to switch to another directory.

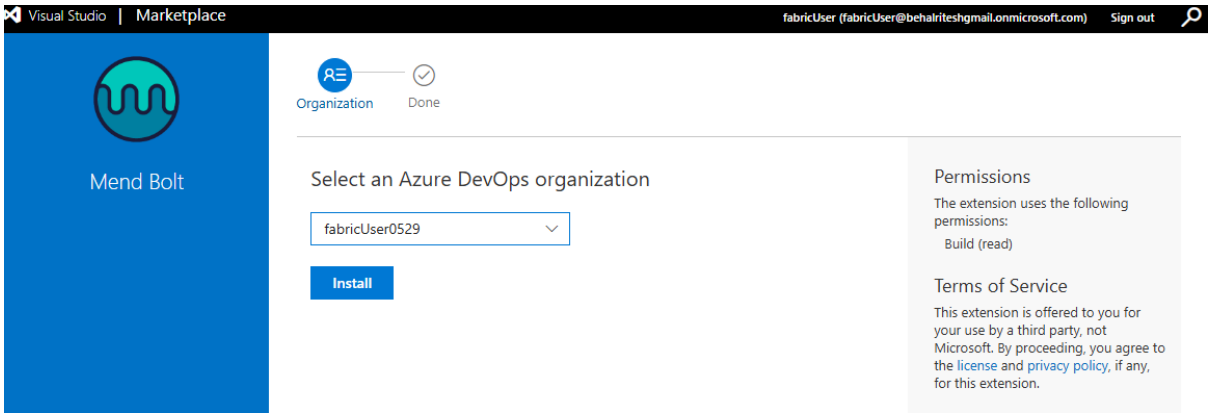


4. Now go to Organization settings choose the extension tab then click on Browse Marketplace.

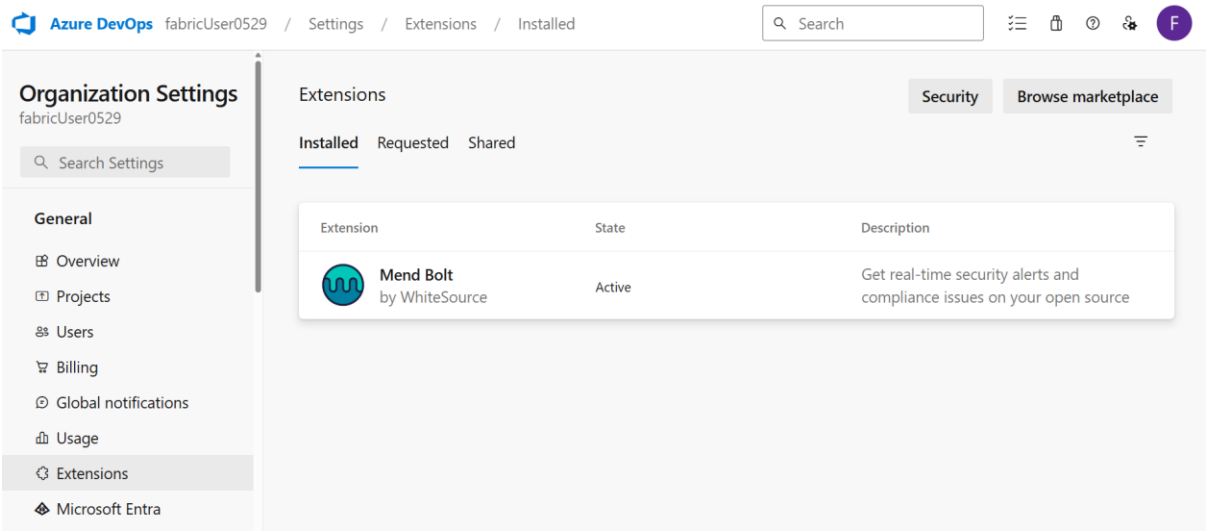


5. In the Marketplace search for Mend and choose the free version and install it.

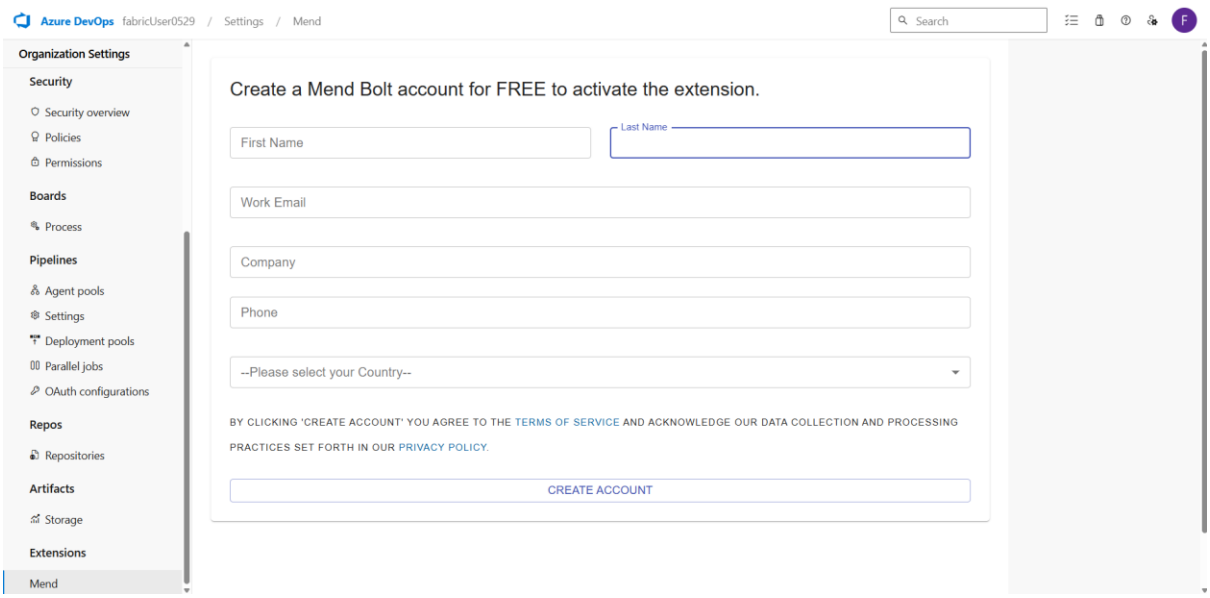




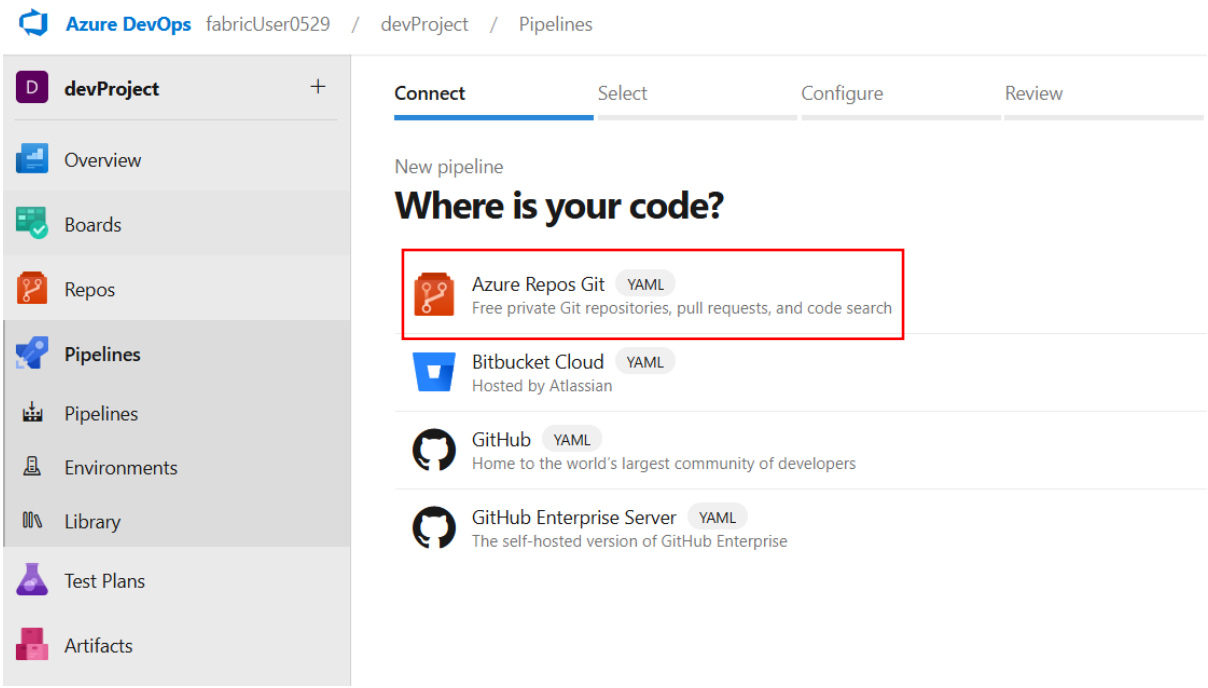
6. After installing it you can see it in your extensions tab in the organization settings.



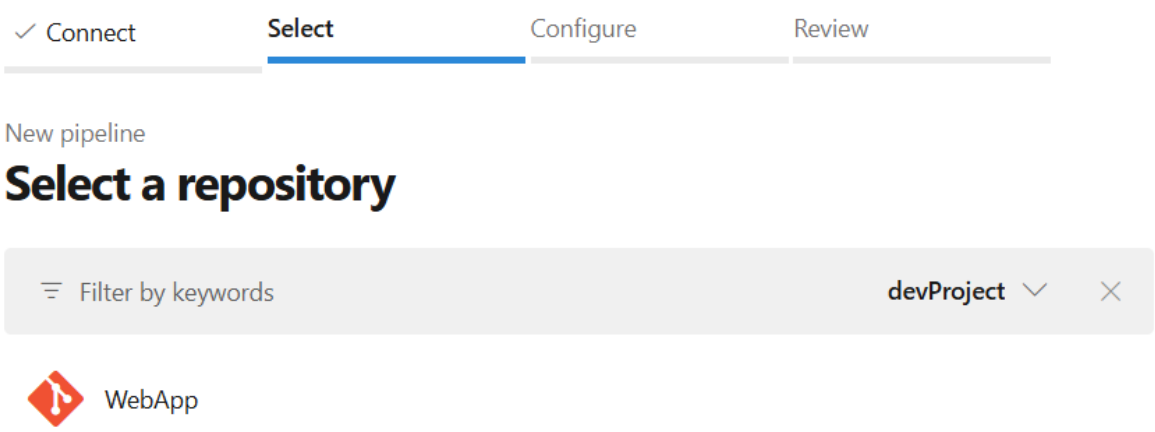
7. If you scroll to the bottom on the left pane, you will see the extension tab here and you have to open mend from here. Then you will see that it will ask you to create an account. Complete this process.



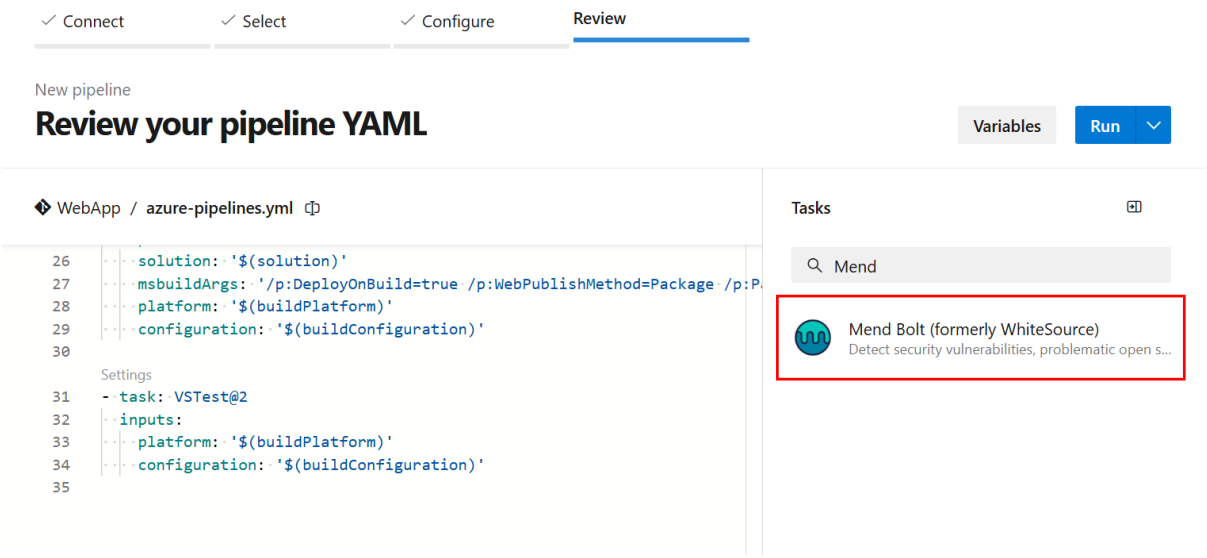
8. Now we are going to create a pipeline for our application stored in Azure Repos. Click on Create Pipeline.
9. Choose Azure Repos Git.



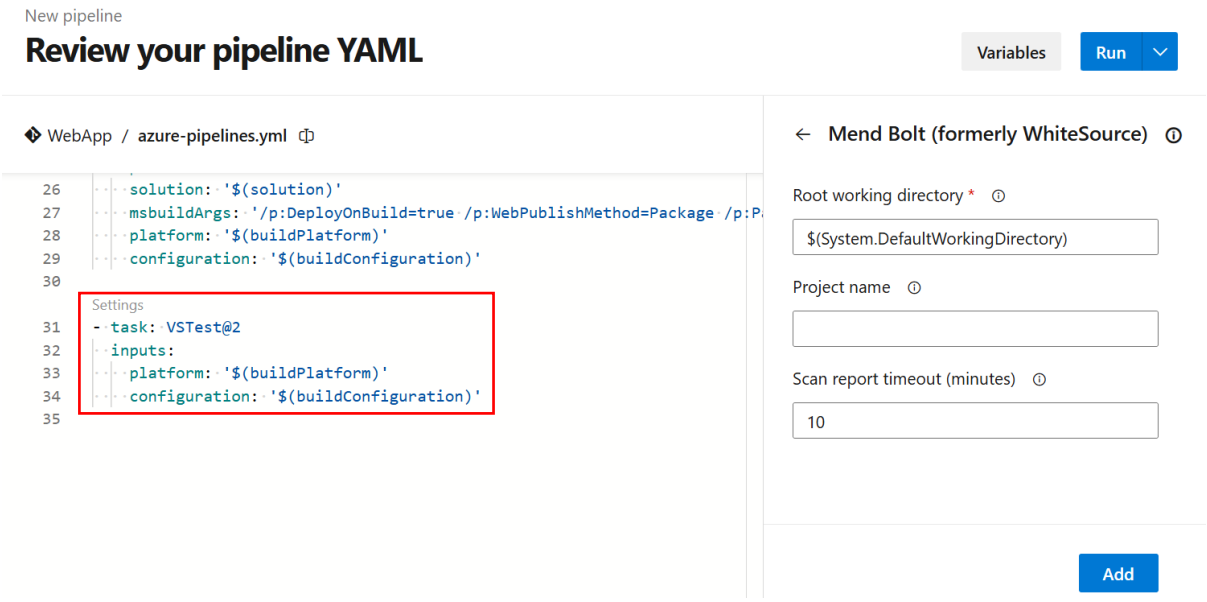
10. Then choose your Web application and move to review page.



11. Now while reviewing your YAML code for the pipeline in the Tasks section search for Mend and click on it.



12. Then keep everything as it is and click on add to work with mend bolt too. Also, delete the task starting from line 31 then add the Mend bolt.



13. Here you can see that the mend bolt task has been added starting from line 31 to 33.

New pipeline

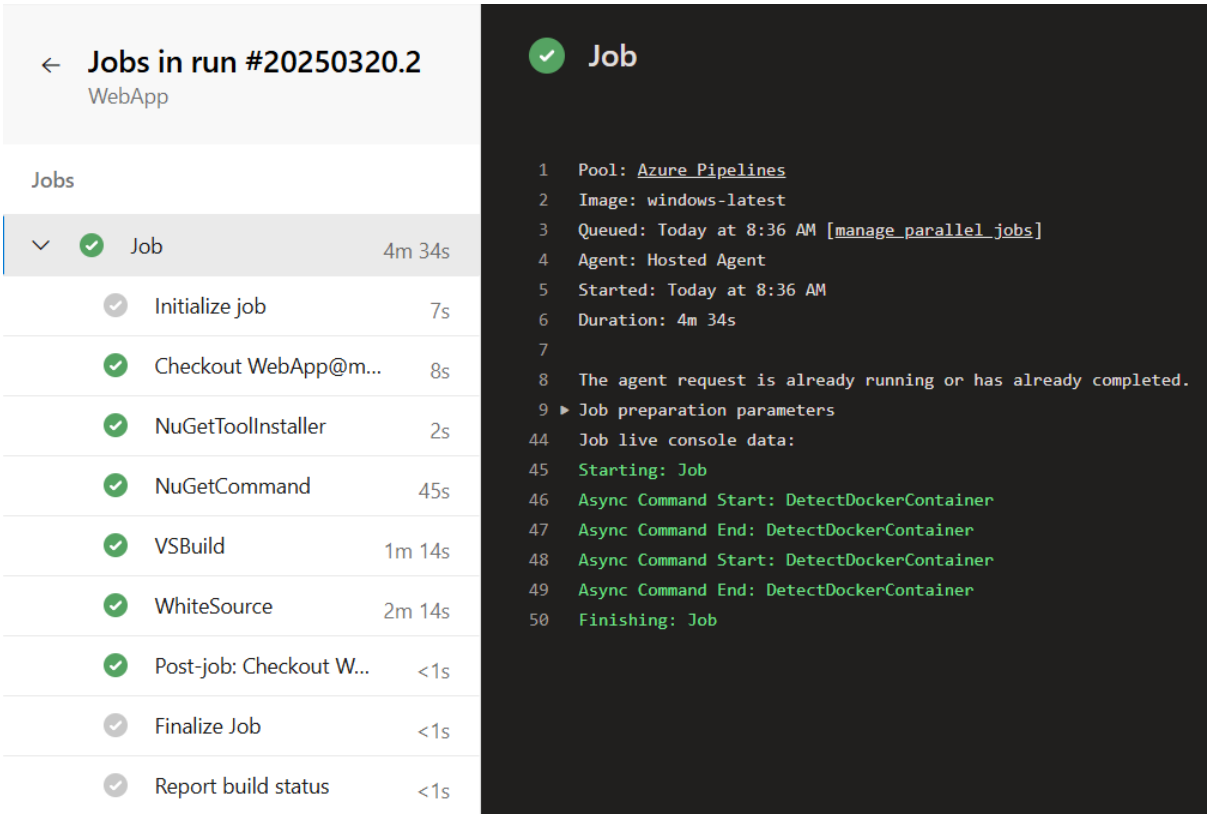
Review your pipeline YAML

WebApp / azure-pipelines.yml *

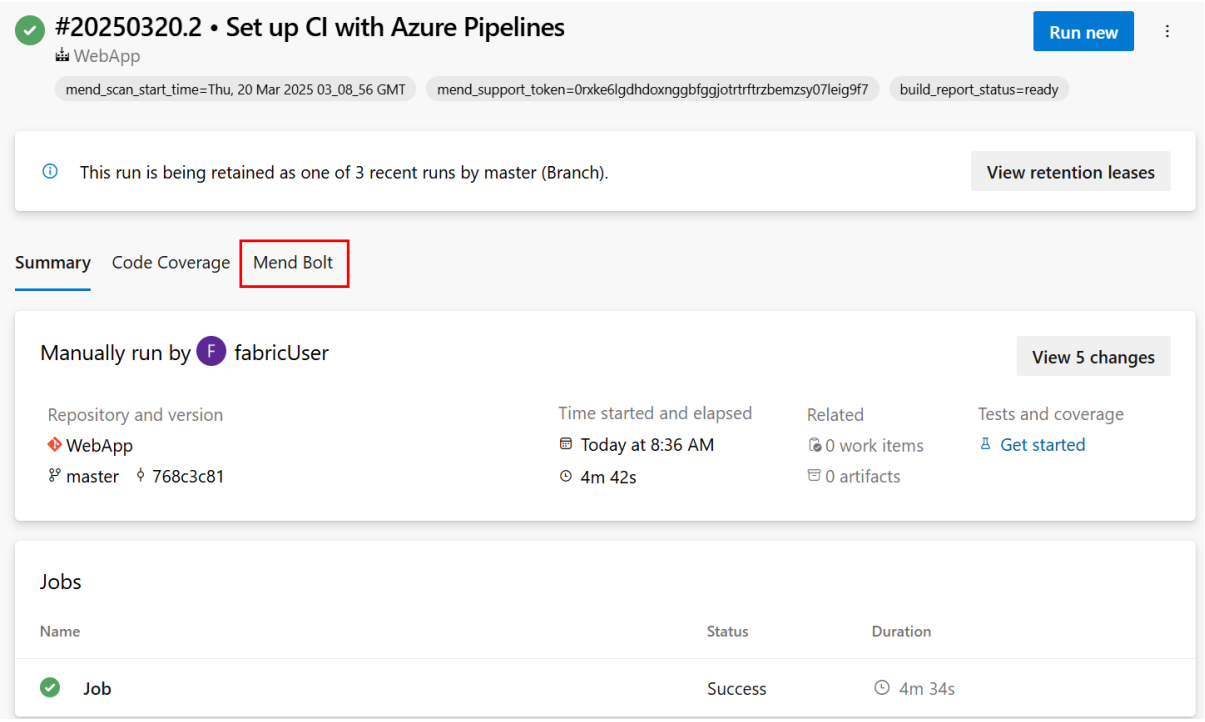
```
Settings
- task: VSBuild@1
  inputs:
    solution: '$(solution)'
    msbuildArgs: '/p:DeployOnBuild=true /p:WebPublishMethod=Package /p:P
    platform: '$(buildPlatform)'
    configuration: '$(buildConfiguration)'

Settings
- task: WhiteSource@21
  inputs:
    cwd: '$(System.DefaultWorkingDirectory)'
```

14. In the end just save and run your pipeline and wait for the job to get completed.
15. Below you can see that your pipeline run has been completed.



16. You can see a new tab for Mend bold after the completion of your Job. Open it.



17. And according to mend there is no risk while running your pipeline.