

Lesson 09 Demo 04

Implementing OWASP ZAP DAST Scan in Jenkins Pipeline

Objective: To implement the ZAP DAST tool using Jenkins declarative pipeline to automate code scan using Jenkins build job

Tools required: Jenkins

Prerequisites: You need to have a Jenkins up and running.

Steps to be followed:

1. Create a Jenkins pipeline job to integrate the vulnerability scan tool

Step 1: Create a Jenkins pipeline job to integrate the vulnerability scan tool

- 1.1 Open the terminal and execute the following commands to install container runtime and Docker tool to proceed with pipeline execution:

```
sudo su
apt update
apt install containerd docker.io
chmod 777 /var/run/docker.sock
service docker restart
```

```
poojahksimplile@ip-172-31-79-37:~$ sudo su
root@ip-172-31-79-37:/home/poojahksimplile# apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:4 https://download.docker.com/linux/ubuntu jammy InRelease [48.8 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Ign:6 https://pkg.jenkins.io/debian-stable binary/ InRelease
Hit:7 https://pkg.jenkins.io/debian-stable binary/ Release
Hit:8 https://prod-cdn.packages.k8s.io/repositories/isv:/kubernetes:/core:/stable:/v1.28/deb InRelease
Ign:9 https://deb.nodesource.com/node_0.10 jammy InRelease
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1638 kB]
Get:11 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 Packages [1074 kB]
Get:12 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe Translation-en [245 kB]
Err:13 https://deb.nodesource.com/node_0.10 jammy Release
  404 Not Found [IP: 104.22.5.26 443]
Get:14 https://download.docker.com/linux/ubuntu jammy/stable amd64 Packages [32.0 kB]
Get:16 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [1427 kB]
Get:17 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [247 kB]
Get:18 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [852 kB]
Get:19 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [164 kB]
```

```

root@ip-172-31-79-37:/home/poojahksimplile# apt install containerd docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  docker-ce-rootless-extras libslirp0 slirp4netns
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  bridge-utils runc ubuntu-fan
Suggested packages:
  ifupdown aufs-tools cgroupfs-mount | cgroup-lite debootstrap docker-doc rinse zfs-fuse | zfsutils
The following packages will be REMOVED:
  containerd.io docker-ce docker-ce-cli
The following NEW packages will be installed:
  bridge-utils containerd docker.io runc ubuntu-fan
0 upgraded, 5 newly installed, 3 to remove and 75 not upgraded.
Need to get 69.3 MB of archives.
After this operation, 5173 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/main amd64 bridge-utils amd64 1.7-1ubuntu3 [34.4 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 runc amd64 1.1.7-0ubuntu1-22.04.2 [4267 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/main amd64 containerd amd64 1.7.2-0ubuntu1-22.04.1 [36.0 MB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 docker.io amd64 24.0.5-0ubuntu1-22.04.1 [28.9 MB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 ubuntu-fan all 0.12.16 [35.2 kB]

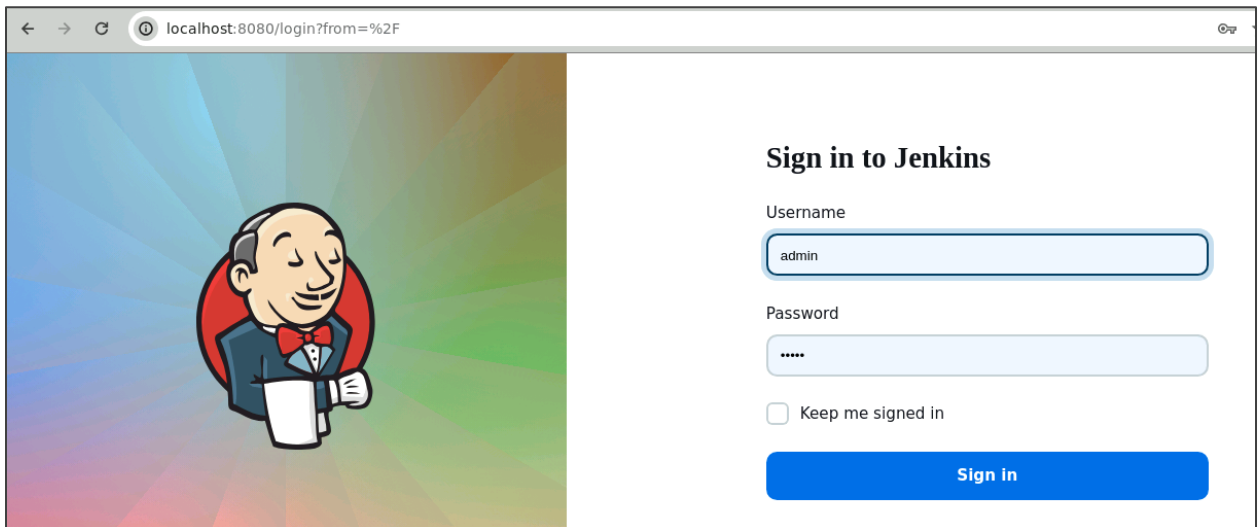
```

```

root@ip-172-31-79-37:/home/poojahksimplile# chmod 777 /var/run/docker.sock
root@ip-172-31-79-37:/home/poojahksimplile# service docker restart
root@ip-172-31-79-37:/home/poojahksimplile#

```

1.2 Log in to Jenkins using your credentials



localhost:8080/login?from=%2F

Sign in to Jenkins

Username

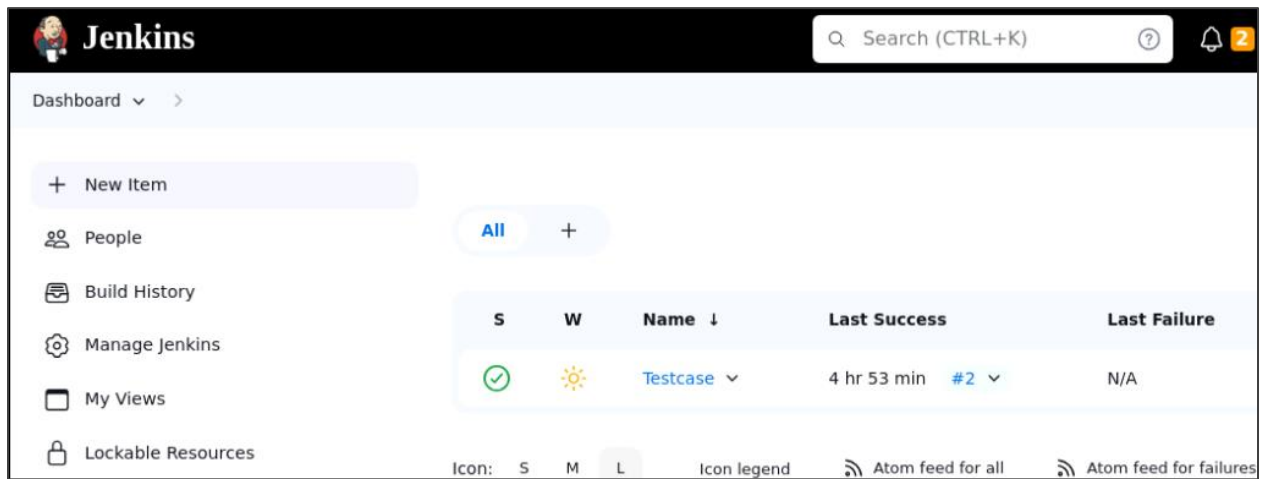
Password

☐ Keep me signed in

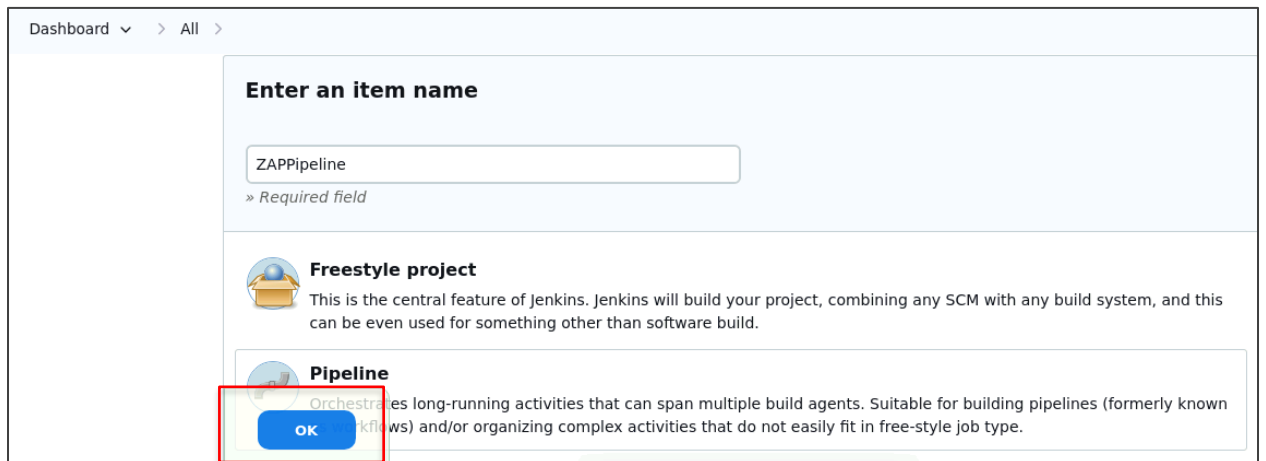
Sign In

Note: The credentials for accessing Jenkins in the lab are Username: **admin** and Password: **Root123\$**.

1.3 In the Jenkins dashboard, click on **New Item**



1.4 Create a Jenkins pipeline job, click on **Pipeline**, put **ZAPPipeline** under **Enter an item name**, and then click on **OK**



1.5 In the **Configure** page, click on **Pipeline** and paste the following code under Script and then click on **Save**:

```
def scan_type

def target

pipeline {
    agent any
    environment {
        zapDockerName = "ghcr.io/zaproxy/zaproxy:stable"
    }
    parameters {
        choice choices: ['Baseline', 'APIS', 'Full'],
            description: 'Type of scan that is going to perform inside the container',
            name: 'SCAN_TYPE'

        string defaultValue: 'https://medium.com/',
            description: 'Target URL to scan',
            name: 'TARGET'

        booleanParam defaultValue: true,
            description: 'Parameter to know if wanna generate report.',
            name: 'GENERATE_REPORT'
    }
    stages {
        stage('Parameter Initialization') {
            steps {
                script {
                    echo ""
```

The current parameters are:

Scan Type: \${params.SCAN_TYPE}

Target: \${params.TARGET}

Generate report: \${params.GENERATE_REPORT}

""""

}

}

}

stage('Setting up OWASP ZAP docker container') {

steps {

echo 'Pulling up last OWASP ZAP container --> Start'

sh "docker pull \${zapDockerName}"

echo 'Pulling up last VMS container --> End'

echo 'Starting container --> Start'

sh "docker run -dt --name owasp \${zapDockerName} /bin/bash "

}

}

stage('Prepare wrk directory') {

when {

environment name : 'GENERATE_REPORT', value: 'true'

}

steps {

script {

sh ""

docker exec owasp \

mkdir /zap/wrk

""

```

    }
}
}

stage('Scanning target on owasp container') {
    steps {
        script {
            scan_type = "${params.SCAN_TYPE}"
            echo "----> scan_type: $scan_type"
            target = "${params.TARGET}"
            if (scan_type == 'Baseline') {
                sh """
                    docker exec owasp \
                    zap-baseline.py \
                    -t $target \
                    -r report.html \
                    -l
                """
            }
            else if (scan_type == 'APIS') {
                sh """
                    docker exec owasp \
                    zap-api-scan.py \
                    -t $target \
                    -r report.html \
                    -l
                """
            }
        }
    }
}

```

```
    else if (scan_type == 'Full') {
        sh """
            docker exec owasp \
            zap-full-scan.py \
            -t $target \
            -r report.html \
            -l
        """
    }
    else {
        echo 'Something went wrong...'
    }
}
}
}

stage('Copy Report to Workspace') {
    steps {
        script {
            sh '''
                docker cp owasp:/zap/wrk/report.html ${WORKSPACE}/report.html
            '''
        }
    }
}

post {
    always {
```

```

    echo 'Removing container'

    sh '''

        docker stop owasp

        docker rm owasp

    '''

    archiveArtifacts 'target/*.jar'

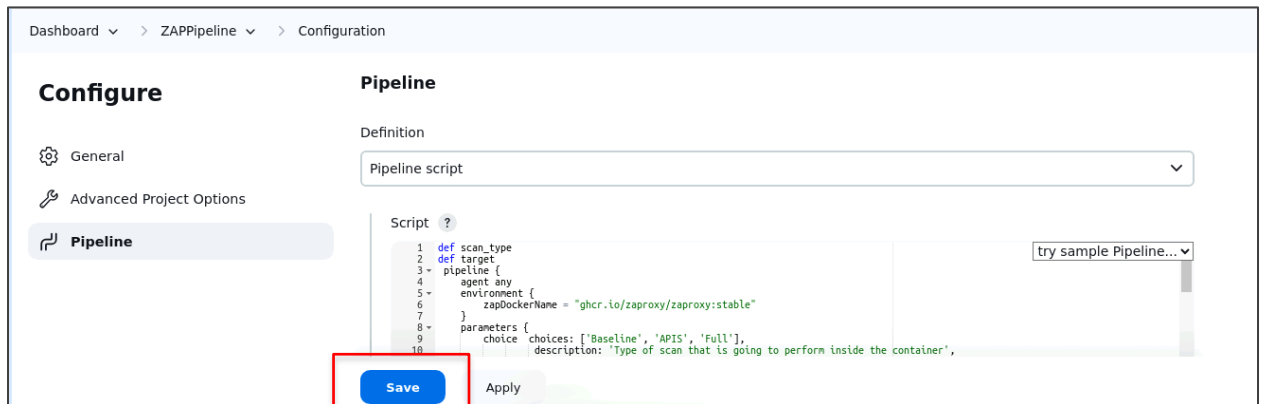
    cleanWs()

}

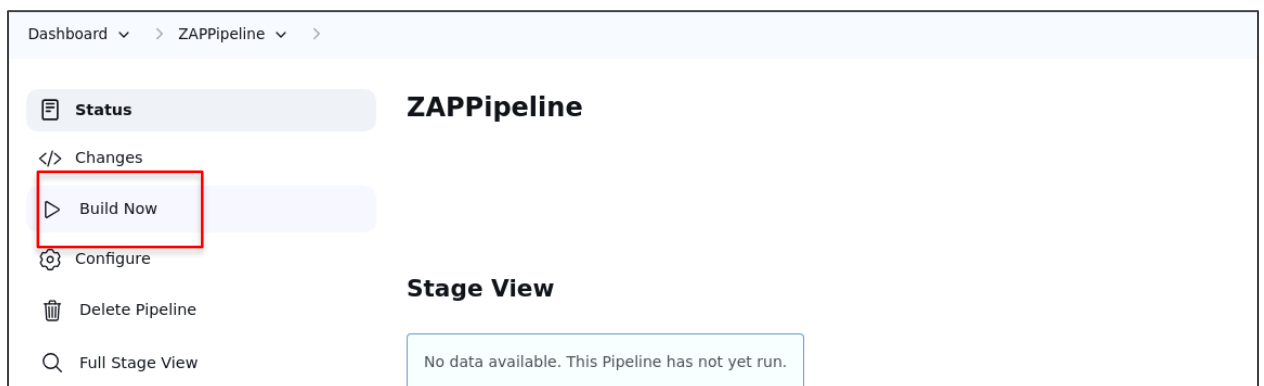
}

}

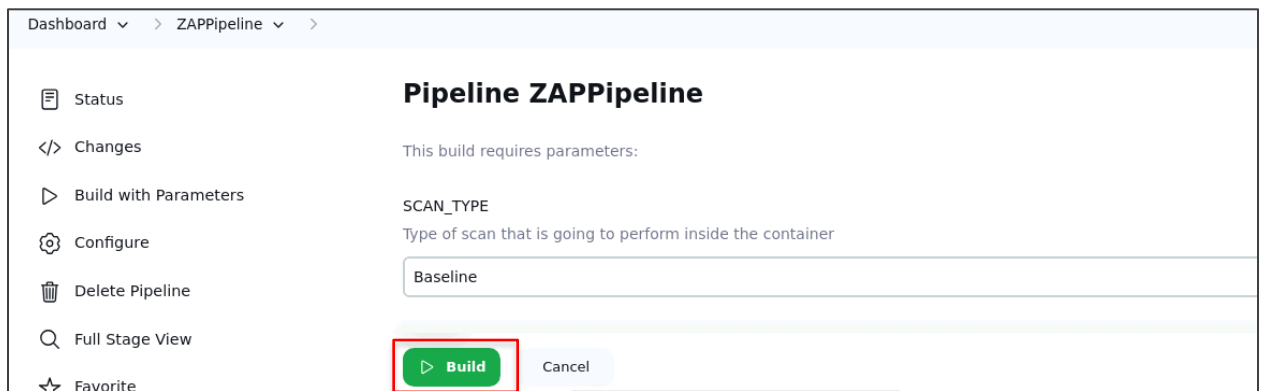
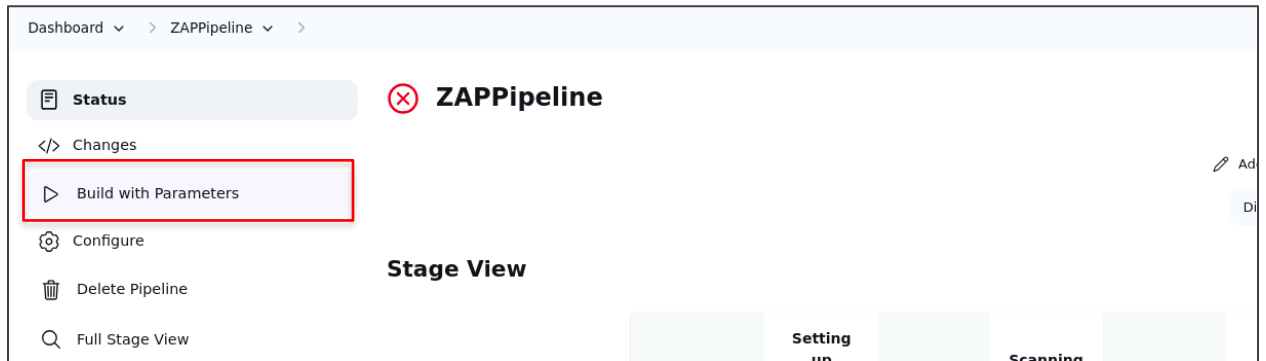
```



1.6 Now, click on **Build Now**



1.7 Initially, the pipeline will fail because it is a parameterized build. Then, **Build with Parameters** option appears. Click on it to initiate the pipeline with the provided parameters and click on **Build**.



You can see the build is successful.

1.8 Navigate back to the **Status** of the build and click on **report.html** to see the HTML report archived in Jenkins

Dashboard > ZAPipeline > #4

Status

Build #4 (May 9, 2024, 5:29:45 AM)

Build Artifacts

report.html 223.75 KB view

Started by user admin

Changes

Console Output

Edit Build Information

Delete build '#4'

Parameters

ZAP Scanning Report

Site: <https://medium.com>

Generated on Thu, 9 May 2024 05:32:24

ZAP Version: 2.15.0

ZAP is supported by the [Crash Override Open Source Fellowship](#)

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	5
Low	6
Informational	9
False Positives:	0

You can see the HTML report.

By following these steps, you have successfully implemented the ZAP DAST tool using Jenkins declarative pipeline to automate code scan using Jenkins build job.