

# MCA DevOps Test Solution Outputs

## Deployment Strategy

### **Blue-Green Deployment**

#### **Description:**

Blue-Green strategy switches traffic between two identical environments.

#### **Steps**

1. Deploy initial (Blue) version. (v1)
2. Deploy new (Green) version. (v2)
3. Switch traffic to Green by Manual/Automatic Promote.
4. Verify rollout.

```
└ helm upgrade -i frontend -f values.yaml -n default .
Release "frontend" does not exist. Installing it now.
NAME: frontend
LAST DEPLOYED: Sat Feb 14 00:34:09 2026
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
1. Get the application URL by running these commands:
  http://mcadevopstest.bnpparibas.com/
└─ [k] ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ p main ?8
  kubectl get all -n default |grep frontend
  pod/foreground-79464f57b5-6zgrz      1/1    Running   0          2m39s
  service/frontend      ClusterIP  10.100.120.104  <none>    80/TCP   2m39s
  service/foreground-preview  ClusterIP  10.100.115.116  <none>    80/TCP   2m39s
  replicaset.apps/foreground-79464f57b5      1        1        1          2m39s
  horizontalpodautoscaler.autoscaling/foreground   Rollout/foreground  12m/80   1        10       1          2m39s
└─ [k] ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ p main ?8
  kubectl get pods -n default -o yaml|grep frontend |grep image:
  - image: netturi19/frontend:v1
    image: docker.io/netturi19/frontend:v1
└─ [k] ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ p main ?8
  helm upgrade -i frontend -f values.yaml -n default .
Release "frontend" has been upgraded. Happy Helming!
NAME: frontend
LAST DEPLOYED: Sat Feb 14 00:38:15 2026
NAMESPACE: default
STATUS: deployed
REVISION: 2
TEST SUITE: None
NOTES:
1. Get the application URL by running these commands:
  http://mcadevopstest.bnpparibas.com/
└─ [k] ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ p main ?8
  kubectl get all -n default |grep frontend
  pod/foreground-79464f57b5-6zgrz      1/1    Running   0          5m12s
  pod/foreground-84b4446b-tslgz      1/1    Running   0          67s
  service/frontend      ClusterIP  10.100.120.104  <none>    80/TCP   5m12s
  service/foreground-preview  ClusterIP  10.100.115.116  <none>    80/TCP   5m12s
  replicaset.apps/foreground-79464f57b5      1        1        1          5m12s
  replicaset.apps/foreground-84b4446b      1        1        1          67s
  horizontalpodautoscaler.autoscaling/foreground   Rollout/foreground  10m/80   1        10       1
└─ [k] ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ p main ?8
  kubectl get pods -n default -o yaml|grep frontend |grep image:
  - image: netturi19/frontend:v1
    image: docker.io/netturi19/frontend:v1
    image: netturi19/frontend:v2
    image: docker.io/netturi19/frontend:v2
    BLUE
    GREEN
```

```

└─ kubectl argo rollouts get rollout frontend
  Name:      frontend
  Namespace: default
  Status:    ⚡ Paused
  Message:   BlueGreenPause
  Strategy:  BlueGreen
  Images:    netturi19/frontend:v1 (stable, active)
              netturi19/frontend:v2 (preview)

  Replicas:
    Desired: 1
    Current: 2
    Updated: 1
    Ready:   1
    Available: 1

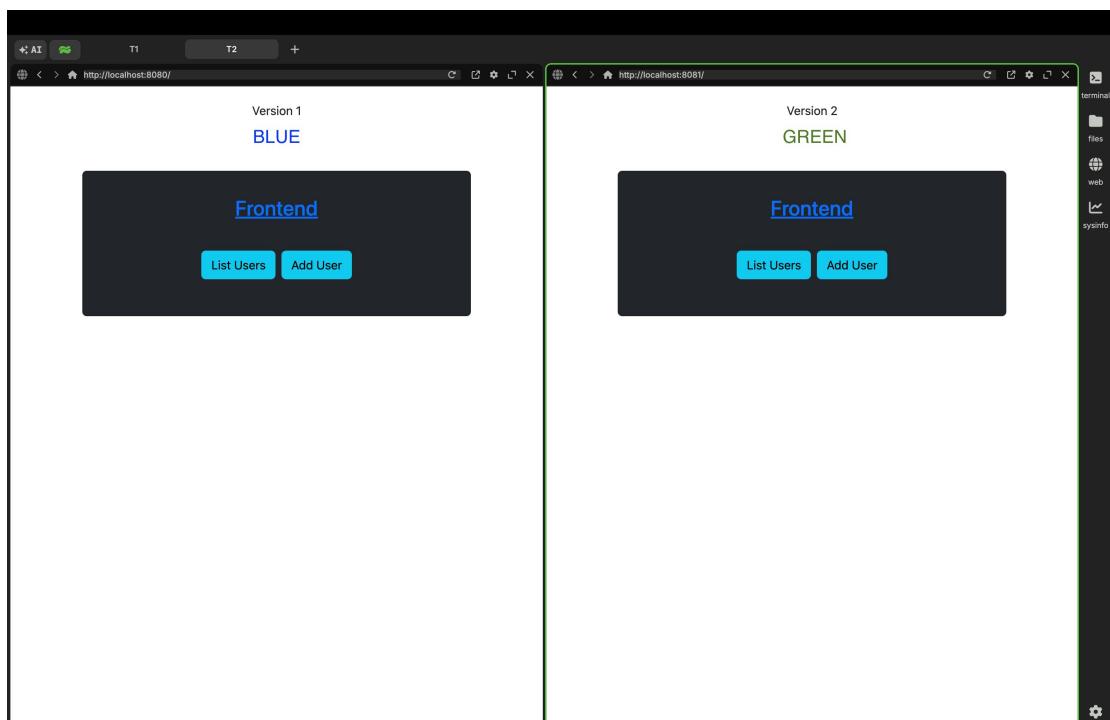
  NAME          KIND  STATUS  AGE  INFO
  ⬤ frontend   Rollout  ⚡ Paused  44m
    # revision:2
      └─ frontend-84b4446b
        └─ frontend-84b4446b-tslgz
          Pod     ✓ Healthy  40m  preview
    # revision:1
      └─ frontend-79464f57b5
        └─ frontend-79464f57b5-6zgrz
          Pod     ✓ Healthy  44m  stable,active
    ↵ [apple] ~/Documents/mca-devops-test/frontend/helm/main ?8
    kubectl argo rollouts promote frontend

  rollout 'frontend' promoted
  ↵ [apple] ~/Documents/mca-devops-test/frontend/helm/main ?8
  kubectl argo rollouts get rollout frontend
  Name:      frontend
  Namespace: default
  Status:    ✓ Healthy
  Strategy:  BlueGreen
  Images:    netturi19/frontend:v1
              netturi19/frontend:v2 (stable, active)

  Replicas:
    Desired: 1
    Current: 2
    Updated: 1
    Ready:   1
    Available: 1

  NAME          KIND  STATUS  AGE  INFO
  ⬤ frontend   Rollout  ✓ Healthy  45m
    # revision:2
      └─ frontend-84b4446b
        └─ frontend-84b4446b-tslgz
          Pod     ✓ Healthy  40m  stable,active
    # revision:1
      └─ frontend-79464f57b5
        └─ frontend-79464f57b5-6zgrz
          Pod     ✓ Healthy  45m  delay:17s
    ↵ [apple] ~/Documents/mca-devops-test/frontend/helm/main ?8
    kubectl get pods -n default -o yaml|grep frontend |grep image:
      - image: netturi19/frontend:v2
      image: docker.io/netturi19/frontend:v2

```



# Canary Deployment

### Description:

Canary strategy gradually shifts traffic to the new version.

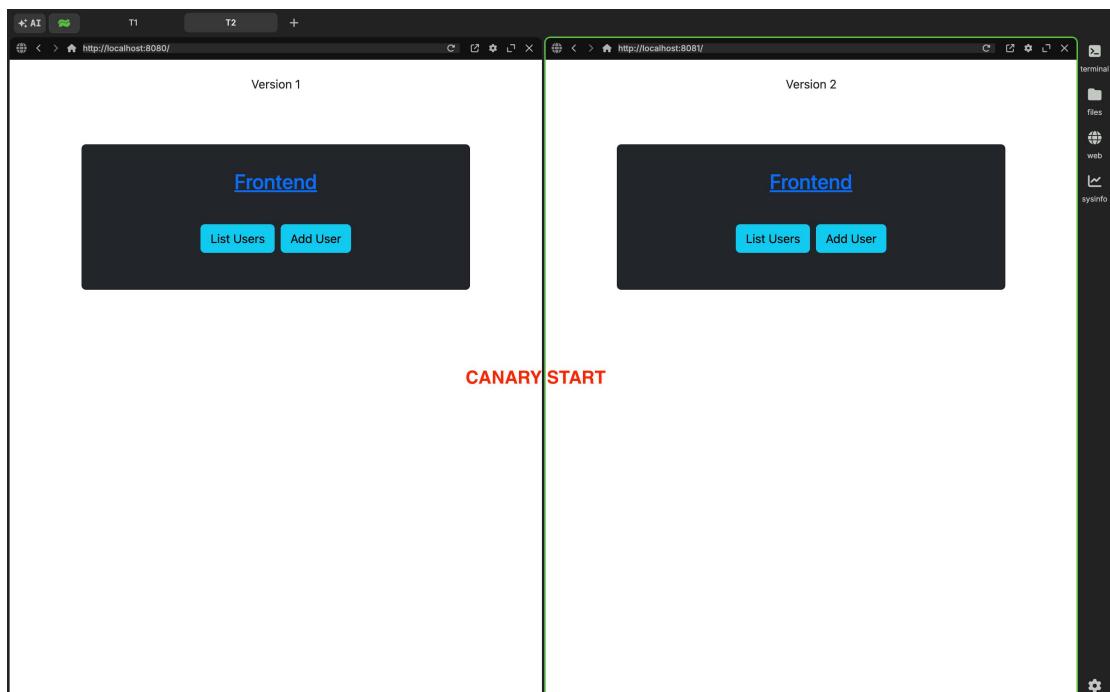
## Steps

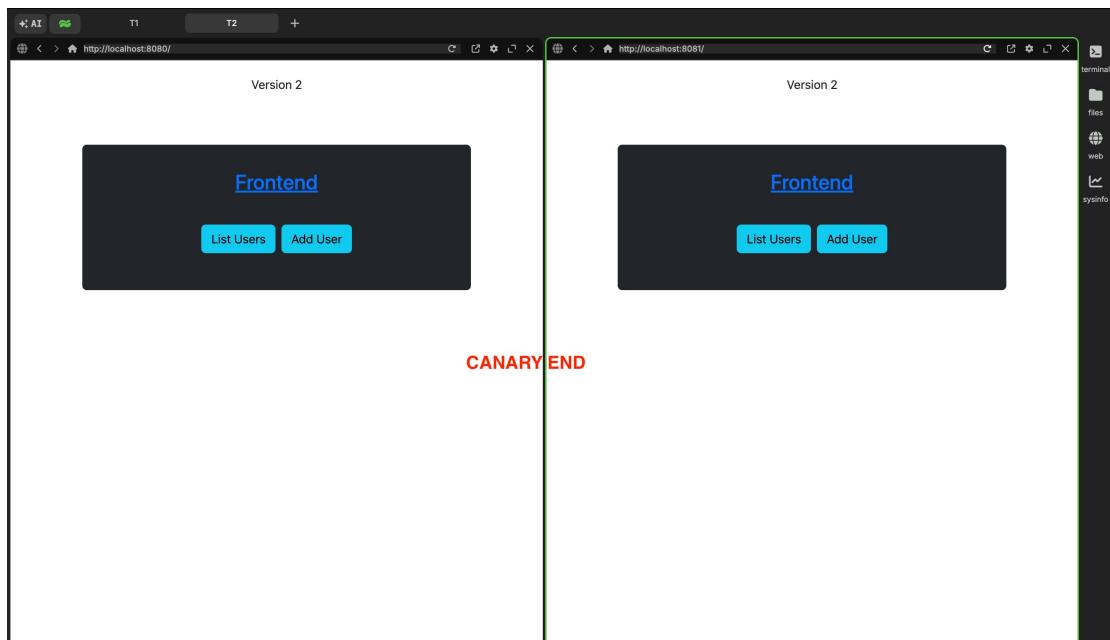
- Deploy base version.
  - Release canary version.
  - Increase traffic weight.
  - Complete rollout.

```
apple ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ P main ?8
helm upgrade -i frontend -f values.yaml -n default .
Release "frontend" does not exist. Installing it now.
NAME: frontend
LAST DEPLOYED: Sat Feb 14 01:26:55 2026
NAMESPACE: default
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
1. Get the application URL by running these commands:
  http://mcadevopstest.bnpparibas.com/
apple ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ P main ?8
kubectl get all -n default |grep frontend
pod/foreground-79464f57b5-5wxmz      1/1    Running   0        101s
pod/foreground-79464f57b5-k2t1c      1/1    Running   0        101s
pod/foreground-79464f57b5-lp9c6      1/1    Running   0        101s
pod/foreground-79464f57b5-zhcmv     1/1    Running   0        116s
pod/foreground-79464f57b5-zrmvx     1/1    Running   0        101s
service/frontend                   ClusterIP 10.100.42.244 <none>       80/TCP  117s
service/frontend-canary           ClusterIP 10.100.233.85 <none>       80/TCP  117s
replicaset.apps/foreground-79464f57b5 5          5          5        116s
horizontalpodautoscaler.autoscaling/foreground   Rollout/foreground 80m/80  5          10          5          116s

apple ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ P main ?8
kubectl get pods -n default -o yaml|grep frontend |grep image:
- image: netturi19/frontend:v1
  image: docker.io/netturi19/frontend:v1

apple ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ P main ?8
helm upgrade -i frontend -f values.yaml -n default .
Release "frontend" has been upgraded. Happy Helming!
NAME: frontend
LAST DEPLOYED: Sat Feb 14 01:29:31 2026
NAMESPACE: default
STATUS: deployed
REVISION: 2
TEST SUITE: None
NOTES:
1. Get the application URL by running these commands:
  http://mcadevopstest.bnpparibas.com/
apple ~ /Documents/mca-devops-test/frontend/helm/frontend ➜ ⌂ P main ?8
```





## Application Functionality

### Description:

Basic application functionality verified after deployment.

The image contains two screenshots of a web application. The top screenshot shows a 'Frontend' interface with a 'List Users' and 'Add User' button. Below it is a form with 'Name' and 'Email' fields, both containing placeholder text ('Premjith Retnakumar' and 'premjith.rk@gmail.com'). There is also a 'Submit' button. The bottom screenshot shows a 'Frontend' interface with a 'List Users' and 'Add User' button. Below it is a table with two rows of user data:

#	Name	Email
1	Premjith Retnakumar	premjith.rk@gmail.com
2	Virat Kohli	abc@gmail.com

Query    Query History

```
1 ▾ SELECT * FROM public.users
2 ORDER BY id ASC
```

Data Output    Messages    Notifications

	<b>id</b> [PK] bigint	<b>email</b> character varying (255)	<b>name</b> character varying (255)
1	1	premjith.rk@gmail.com	Premjith Retnakumar
2	2	abc@gmail.com	Virat Kohli

## SonarQube Code Analysis

### Description:

Static code analysis performed before deployment.

### Checks Performed

- Code Quality Gate
- Bugs & Vulnerabilities
- Code Smells
- Coverage

### Screenshots

#### Frontend Sonarqube Scan Report

The screenshot shows the SonarQube Community Frontend Scan Report for the 'main' branch. The report is green and labeled 'Passed'. It displays the following metrics:

- 545 Lines of Code
- Version 1.0.0
- Last analysis 1 minute ago
- Quality Gate: Passed
- Security: 0 Open issues (A)
- Reliability: 2 Open issues (C)
- Maintainability: 4 Open issues (A)
- Coverage: 0.0% (On 10 lines to cover)
- Accepted issues: 0
- Duplications: 54.7% (On 625 lines)
- Security Hotspots: 0

## Backend Sonarqube Scan Report

SonarQube community Projects Issues Rules Quality Profiles Quality Gates Administration More ▾

backend Bind project main ?

Overview Issues Security Hotspots Code Measures Activity Project Settings ▾ Project Information

To benefit from more of SonarQube Community Build's features, [set up analysis](#) in your favorite CI and set up DevOps platform integration in your [project settings](#).

main 88 Lines of Code Version not provided Set as homepage

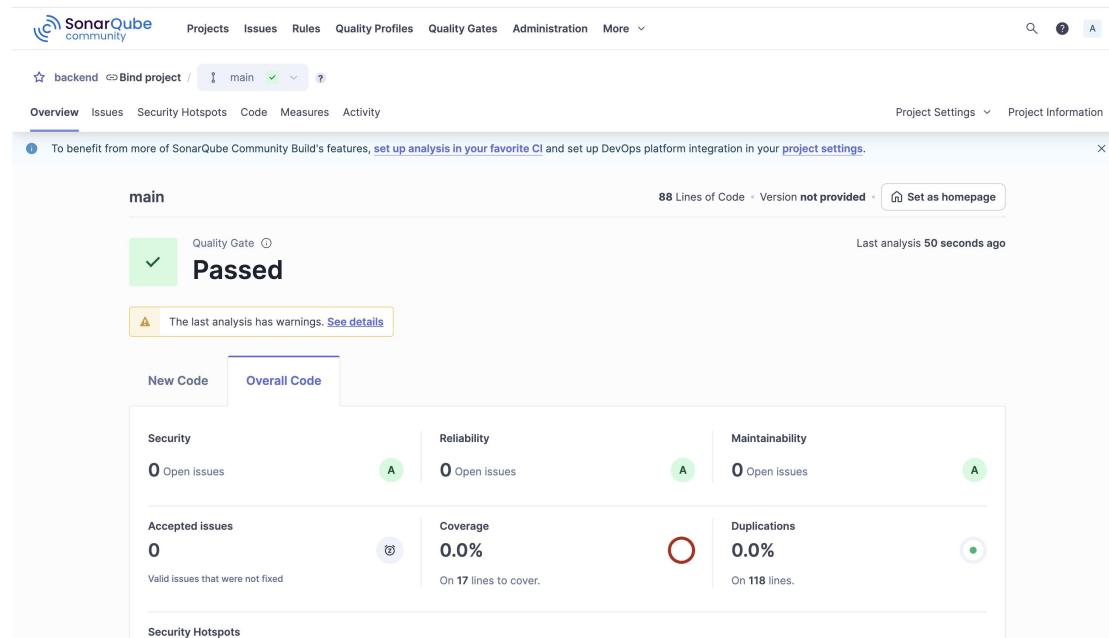
Quality Gate Passed Last analysis 50 seconds ago

The last analysis has warnings. See details

New Code Overall Code

Security	Reliability	Maintainability
0 Open issues	A	0 Open issues
0 Accepted issues	Critical	0 Open issues
0 Valid issues that were not fixed	Critical	A
Coverage 0.0% On 17 lines to cover.		
Duplications 0.0% On 118 lines.		

Security Hotspots



SonarQube community Projects Issues Rules Quality Profiles Quality Gates Administration More ▾

My Favorites All

Create Project ▾

Filters Clear All Filters

Quality Gate

Passed	Failed
2	0

Security

A ≥ 0 info issues	B ≥ 1 low issue	C ≥ 1 medium issue	D ≥ 1 high issue	E ≥ 1 blocker issue
2	0	0	0	0

Reliability

A ≥ 0 info issues	B ≥ 1 low issue	C ≥ 1 medium issue	D ≥ 1 high issue
1	0	1	0

backend Public Last analysis: 1 minute ago - 88 Lines of Code - Java, Unknown Passed

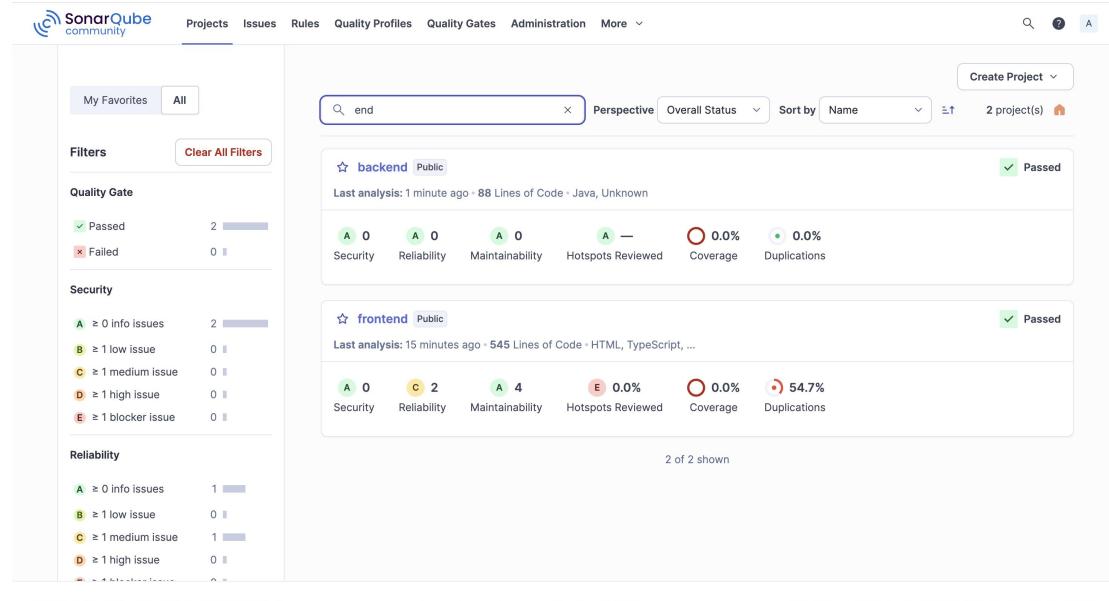
Security	Reliability	Maintainability	Hotspots Reviewed	Coverage	Duplications
A 0	A 0	A 0	A —	0.0%	0.0%

frontend Public Last analysis: 15 minutes ago - 545 Lines of Code - HTML, TypeScript, ... Passed

Security	Reliability	Maintainability	Hotspots Reviewed	Coverage	Duplications
A 0	C 2	A 4	E 0.0%	0.0%	54.7%

2 of 2 shown

SonarQube™ technology is powered by [SonarSource SA](#) Community Build - v25.8.0.112029 - MQR MODE GPL v3 ▾ Community ▾ Documentation ▾ Plugins ▾ Web API



# Jenkins CI/CD Pipeline

## Description:

The CI/CD pipeline automates building, testing, scanning, and deploying the application.

The screenshot shows the Jenkins interface for the 'Backend-Pipeline'. On the left, there's a sidebar with links like Status, Changes, Build Now, Configure, Delete Pipeline, Full Stage View, Trivy FS Scan Report, Trivy Image Scan Report, Trivy Image Secret Scan Report, Favorite, Job Config History, Open Blue Ocean, Rename, and Pipeline Syntax. The main area is titled 'Backend-Pipeline' and shows a 'Stage View' for four builds. Each build has a row of colored bars representing different stages: Build, SonarQube, SCA trivy FS scan, trivy FS SBOM Generation and Scan, Docker Image build, trivy image SBOM Generate, trivy image scan, trivy image secrets scan, Docker Image Push, K8s UAT-Deploy, and Clean Workspace After build. Below each bar is a numerical value representing the average stage time. The first build (Feb 14, 15:22) has values: 26s, 25s, 42s, 39s, 3s, 682ms, 1s, 768ms, 18s, 331ms, 944ms. The second build (Feb 14, 15:17) has values: 26s, 25s, 33s, 39s, 3s, 689ms, 1s, 806ms, 18s, 331ms, 941ms. The third build (Feb 14, 15:09) has values: 27s, 25s, 47s, 39s, 3s, 680ms, 1s, 765ms, 19s, 946ms. The fourth build (Feb 14, 15:03) has values: 25s, 26s, 39s, 39s, 3s, 685ms, 1s, 759ms, 18s, 941ms. The fifth build (Feb 14, 15:03) has values: 26s, 26s, 47s, 39s, 3s, 676ms, 1s, 744ms, 18s, 951ms. A note at the top says 'Average stage times (full run time: ~2min 40s)'.

# Trivy Security Scanning

## Description:

Security scans performed on both the filesystem and container images.

## Filesystem Scan

Scans project source code and dependencies for vulnerabilities.

### pom.xml – Trivy Report – 2026-02-14 02:45:37.320206441 +0530 IST m=+47.243644310

pom						Links
Package	Vuln ID	Severity	Installed	Fixed	Description	
ch.qos.logback:logback-core	CVE-2024-12798	MEDIUM	1.5.11	1.5.13, 1.3.15	ACE vulnerability in JaninoEventEvaluator by QOS.CH logback-core upto including version 0.1 to 1.3.14 and 1.4.0 to 1.5.12 in Java applications allows attacker to execute	<a href="https://access.redhat.com/security/cve/CVE-2024-12798">https://access.redhat.com/security/cve/CVE-2024-12798</a> <a href="https://github.com/qos-ch/logback">https://github.com/qos-ch/logback</a> <a href="https://github.com/qos-ch/logback/commit/2cb6d520df7592ef1c3a1">https://github.com/qos-ch/logback/commit/2cb6d520df7592ef1c3a1</a> <a href="https://logback.qos.ch/news.html#1.3.15">https://logback.qos.ch/news.html#1.3.15</a>
ch.qos.logback:logback-core	CVE-2025-11226	MEDIUM	1.5.11	1.5.19, 1.3.16	ACE vulnerability in conditional configuration file processing by QOS.CH logback-core up to and including version 1.5.18 in Java applications, allows an attacker to execute	<a href="https://access.redhat.com/security/cve/CVE-2025-11226">https://access.redhat.com/security/cve/CVE-2025-11226</a> <a href="https://github.com/qos-ch/logback">https://github.com/qos-ch/logback</a> <a href="https://github.com/qos-ch/logback/commit/616a2544f36b3016e0ef">https://github.com/qos-ch/logback/commit/616a2544f36b3016e0ef</a> <a href="https://logback.qos.ch/issues/974">https://logback.qos.ch/issues/974</a>
ch.qos.logback:logback-core	CVE-2024-12801	LOW	1.5.11	1.5.13, 1.3.15	Server-Side Request Forgery (SSRF) in SaxEventRecorder by QOS.CH logback-core up to 1.3.14 and 1.4.0 to 1.5.12 on the Java platform, allows an attacker to forge	<a href="https://access.redhat.com/security/cve/CVE-2024-12801">https://access.redhat.com/security/cve/CVE-2024-12801</a> <a href="https://github.com/qos-ch/logback">https://github.com/qos-ch/logback</a> <a href="https://github.com/qos-ch/logback/commit/f05041cba4c4ac0a627">https://github.com/qos-ch/logback/commit/f05041cba4c4ac0a627</a> <a href="https://logback.qos.ch/news.html#1.3.15">https://logback.qos.ch/news.html#1.3.15</a>
ch.qos.logback:logback-core	CVE-2026-1225	LOW	1.5.11	1.5.25	ACE vulnerability in configuration file processing by QOS.CH logback-core up to and including version 1.5.24 in Java applications, allows an attacker to instantiate classes	<a href="https://access.redhat.com/security/cve/CVE-2026-1225">https://access.redhat.com/security/cve/CVE-2026-1225</a> <a href="https://github.com/qos-ch/logback">https://github.com/qos-ch/logback</a> <a href="https://github.com/qos-ch/logback/commit/197ae1844b1be8486ed">https://github.com/qos-ch/logback/commit/197ae1844b1be8486ed</a> <a href="https://logback.qos.ch/issues/997">https://logback.qos.ch/issues/997</a>
org.apache.commons:commons-lang3	CVE-2025-48924	MEDIUM	3.14.0	3.18.0	Uncontrolled Recursion vulnerability in Apache Commons Lang. This issue affects Apache Commons Lang. Starting with commons-lang:commons-lang 2.0 to 2.6, and, from	<a href="http://www.openwall.com/lists/oss-security/2025/07/11/1">http://www.openwall.com/lists/oss-security/2025/07/11/1</a> <a href="https://access.redhat.com/security/cve/CVE-2025-48924">https://access.redhat.com/security/cve/CVE-2025-48924</a> <a href="https://github.com/apache/commons-lang/commit/h424803a0d2be818e4fcb251ce031c22aca53">https://github.com/apache/commons-lang/commit/h424803a0d2be818e4fcb251ce031c22aca53</a>
org.apache.tomcat.embed:tomcat-embed-core	CVE-2025-24813	CRITICAL	10.1.31	10.1.31, 10.1.35, 9.0.99	Path Equivalence: 'file.getName()' (Internal Dot) leading to Remote Code Execution and/or Information disclosure and/or malicious content added to uploaded files	<a href="http://www.openwall.com/lists/oss-security/2025/03/10/5">http://www.openwall.com/lists/oss-security/2025/03/10/5</a> <a href="https://access.redhat.com/security/cve/CVE-2025-24813">https://access.redhat.com/security/cve/CVE-2025-24813</a> <a href="https://logback.qos.ch/commit/2332817">https://logback.qos.ch/commit/2332817</a>
org.apache.tomcat.embed:tomcat-	CVE-	HIGH	10.1.31	11.0.2,	Time-of-check Time-of-use (TOCTOU) Race Condition	<a href="http://www.openwall.com/lists/oss-security/2024/12/17/4">http://www.openwall.com/lists/oss-security/2024/12/17/4</a>

## Trivy Report – 2026-02-14 02:39:34.441954114 +0530 IST m=+0.386795348

CRITICAL: 0 HIGH: 0 MEDIUM: 0 LOW: 0 UNKNOWN: 0

Filter by package/type All Severities ▾

No scan results found.

## Docker Image Scan

Scans the built container image for OS and library vulnerabilities.

### Screenshot

#### backend:v1 (debian 12.13) – Trivy Report – 2026-02-14 00:05:13.59030905 +0530 IST m=+45.666660901

CRITICAL: 2 HIGH: 12 MEDIUM: 14 LOW: 25 UNKNOWN: 0

Filter by package/type All Severities ▾

Package	Vuln ID	Severity	Installed	Fixed	Description	Links
gcc-12-base	CVE-2022-27943	LOW	12.2.0-14+deb12u1		liberty/rust-demangle.c in GNU GCC 11.2 allows stack consumption in demangle_const, as demonstrated by nm-new. Show more	<a href="https://access.redhat.com/security/cve/CVE-2022-27943">https://access.redhat.com/security/cve/CVE-2022-27943</a> <a href="https://gcc.gnu.org/bugzilla/show_bug.cgi?id=105020">https://gcc.gnu.org/bugzilla/show_bug.cgi?id=105020</a> <a href="https://gcc.gnu.org/gitoliteweb.cgi?i=cc-ccc.git;h=1a770b01e14">https://gcc.gnu.org/gitoliteweb.cgi?i=cc-ccc.git;h=1a770b01e14</a> <a href="https://gcc.gnu.org/gitoliteweb.cgi?i=cc-ccc.git;h=9234cdca6e6">https://gcc.gnu.org/gitoliteweb.cgi?i=cc-ccc.git;h=9234cdca6e6</a>
libc6	CVE-2026-0861	HIGH	2.36-9+deb12u13		Passing too large an alignment to the memalign suite of functions (memalign, posix_memalign, aligned_alloc) in the GNU C Library version 2.30 to 2.42 may result in an integer	<a href="http://www.openwall.com/lists/oss-security/2026/01/16/5">http://www.openwall.com/lists/oss-security/2026/01/16/5</a> <a href="https://access.redhat.com/errata/RHSA-2026-1334">https://access.redhat.com/errata/RHSA-2026-1334</a> <a href="https://access.redhat.com/security/cve/CVE-2026-0861">https://access.redhat.com/security/cve/CVE-2026-0861</a> <a href="https://bugzilla.redhat.com/2429771">https://bugzilla.redhat.com/2429771</a>
libc6	CVE-2025-15281	MEDIUM	2.36-9+deb12u13		Calling wordexp with WRDE_REUSE in conjunction with WRDE_APPEND in the GNU C Library version 2.0 to version 2.42 may cause the interface to return uninitialized	<a href="http://www.openwall.com/lists/oss-security/2026/01/20/3">http://www.openwall.com/lists/oss-security/2026/01/20/3</a> <a href="https://access.redhat.com/security/cve/CVE-2025-15281">https://access.redhat.com/security/cve/CVE-2025-15281</a> <a href="https://nvd.nist.gov/vuln/detail/CVE-2025-15281">https://nvd.nist.gov/vuln/detail/CVE-2025-15281</a> <a href="https://sourceware.org/bugzilla/show_bug.cgi?id=33814">https://sourceware.org/bugzilla/show_bug.cgi?id=33814</a>
libc6	CVE-2026-0915	MEDIUM	2.36-9+deb12u13		Calling getnetbyaddr or getnetbyaddr_r with a configured nsswitch.conf that specifies the library's DNS backend for networks and queries for a zero-valued network in the GNU	<a href="http://www.openwall.com/lists/oss-security/2026/01/16/6">http://www.openwall.com/lists/oss-security/2026/01/16/6</a> <a href="https://access.redhat.com/errata/RHSA-2026-1334">https://access.redhat.com/errata/RHSA-2026-1334</a> <a href="https://access.redhat.com/security/cve/CVE-2026-0915">https://access.redhat.com/security/cve/CVE-2026-0915</a> <a href="https://bugzilla.redhat.com/2429771">https://bugzilla.redhat.com/2429771</a>
libc6	CVE-2010-4756	LOW	2.36-9+deb12u13		The glob implementation in the GNU C Library (aka glibc or libc6) allows remote authenticated users to cause a denial of service (CPU and memory consumption) via crafted glob	<a href="http://cxb.net/stuff/glob-0day.c">http://cxb.net/stuff/glob-0day.c</a> <a href="http://securityreason.com/achievement_securityalert/89">http://securityreason.com/achievement_securityalert/89</a> <a href="http://securityreason.com/exploitalert/9223">http://securityreason.com/exploitalert/9223</a> <a href="https://access.redhat.com/security/cve/CVE-2010-4756">https://access.redhat.com/security/cve/CVE-2010-4756</a>
libc6	CVE-2018-20796	LOW	2.36-9+deb12u13		In the GNU C Library (aka glibc or libc6) through 2.29, check_dst_limits_calc_pos_1 in posix/regexec.c has Uncontrolled Recursion, as demonstrated by '(227)'	<a href="http://www.securityfocus.com/bid/107180">http://www.securityfocus.com/bid/107180</a> <a href="https://access.redhat.com/security/cve/CVE-2018-20796">https://access.redhat.com/security/cve/CVE-2018-20796</a> <a href="https://debsbugs.gnu.org/cgi/bugreport.cgi?bug=34141">https://debsbugs.gnu.org/cgi/bugreport.cgi?bug=34141</a> <a href="https://lists.gnu.org/archive/html/bug-gnulib/2019-01/msg00000.html">https://lists.gnu.org/archive/html/bug-gnulib/2019-01/msg00000.html</a>
libc6	CVE-	LOW	2.36-		GNU Libc current is affected by: Mitigation bypass. The	<a href="https://access.redhat.com/security/cve/CVE-2019-1010022">https://access.redhat.com/security/cve/CVE-2019-1010022</a>

#### frontend:v1 (alpine 3.23.3) – Trivy Report – 2026-02-13 23:59:16.76959231 +0530 IST m=+12.362145552

CRITICAL: 0 HIGH: 0 MEDIUM: 0 LOW: 0 UNKNOWN: 0

Filter by package/type All Severities ▾

alpine

No Vulnerabilities found

No Misconfigurations found