OR-001a — Developer Operations

Go to OR-001 — Operation by Identified User Categories

Statement

Crucible SHALL perform each Crucible operation invoked by a Developer in accordance with the applicable operational scenario.

Derived From

This requirement derives from:

- OR-001 — Operation by Identified User Categories

The Original Requirement states:

> *The system SHALL support operation by:*
>
> > *Developers*
> > *DevSecOps Engineers*
> > *Platform Engineers*
> > *System Administrators*
> > *Security Engineers*
> > *Compliance Officers*[C1]

OR-001a preserves the intent that Developers participate in Crucible operation by requiring Crucible to perform each operation invoked by a Developer in accordance with the applicable operational scenario.

The separate requirements derived from OR-001 address:

- OR-001b — DevSecOps Engineer Operations
- OR-001c — Platform Engineer Operations
- OR-001d — System Administrator Operations
- OR-001e — Security Engineer Operations
- OR-001f — Compliance Officer Operations

Rationale

The Original Requirement combines six independently testable user categories within one normative statement.

OR-001a separates Developer operations from operations associated with the other identified user categories. The separate requirement provides a stable identifier for verification and Traceability.

The applicable operational scenario identifies:

- The Crucible operations available to a Developer
- The conditions under which a Developer invokes each operation
- The Controlled Inputs provided by a Developer
- The status and results presented to a Developer
- The completion conditions applicable to each operation
- The failure and exception conditions applicable to each operation
- The Artifacts and Evidence produced by each operation

The Crucible Concept of Operations [C2] identifies Build, Capture, and Deploy as high-level Crucible operations.

OR-001a does not establish that:

- Every Crucible operation applies to a Developer
- Every Developer invokes the same Crucible operations
- Every operational scenario identifies the same Developer operations
- Developer constitutes an Operational Role
- A particular role-assignment mechanism applies
- A particular authorization or access-control model applies
- A particular interface supports Developer operations

Separate operational and functional requirements define the detailed behavior, inputs, outputs, completion conditions, failures, exceptions, Artifacts, and Evidence associated with each Crucible operation.

Applies To

This requirement applies to:

- Crucible
- Developers
- Operational scenarios applicable to Developers
- Crucible operations identified for Developers
- Developer invocation of Crucible operations
- Controlled Inputs
- Operation status
- Operation results
- Operation failures
- Operation exceptions
- Artifacts
- Evidence
- Provenance
- Traceability
- Connected Environments
- Disconnected Environments
- Air-Gapped Environments

Verification

Verification confirms that:

1. The applicable operational scenario identifies each Crucible operation that a Developer can invoke
2. The applicable operational scenario identifies the conditions under which the Developer can invoke each operation
3. The Developer can invoke each identified Crucible operation under the specified conditions
4. Crucible performs each operation invoked by the Developer
5. The performed operation corresponds to the operation invoked by the Developer
6. Crucible produces each status, result, failure indication, exception indication, Artifact, Evidence item, or other output specified for the operation
7. The verification record preserves Traceability among the applicable operational scenario, the Developer invocation, and the performed Crucible operation

Verification includes:

- Inspection of the applicable operational scenario
- Inspection of the Crucible operations identified for the Developer
- Inspection of the invocation conditions
- Invocation of each identified Crucible operation
- Observation of Crucible performance of each invoked operation
- Comparison of each invoked operation with the corresponding performed operation
- Inspection of operation status and results
- Failure-condition testing
- Exception-condition testing
- Artifact inspection
- Evidence inspection
- Provenance inspection
- Traceability inspection

The verification record identifies:

1. The applicable operational scenario
2. The Developer
3. The invoked Crucible operation
4. The invocation conditions
5. The Controlled Inputs provided by the Developer
6. The performed Crucible operation
7. The operation completion condition
8. Each generated status or result
9. Each observed failure or exception
10. Each generated Artifact
11. The generated Evidence
12. The associated Provenance
13. The associated Traceability

Related Architecture Sections

No related architecture sections have been identified.

Referenced By

The following pages reference this requirement:

- OR-001b — DevSecOps Engineer Operations
- OR-001c — Platform Engineer Operations
- OR-001d — System Administrator Operations
- OR-001e — Security Engineer Operations
- OR-001f — Compliance Officer Operations
- Annex D: Requirements Traceability

Delivery Phase

To Be Determined

Implementation Status

Not Assessed

Implementation status requires verification that the current Crucible implementation performs each Crucible operation invoked by a Developer in accordance with the applicable operational scenario.

Requirement Status

Draft

This requirement derives from OR-001 in the Crucible System Requirements Specification, Version 1.1 Draft.

Issues

The following unresolved issues affect this requirement:

- ☐ Define **Developer** or reference an authoritative definition. Clarify whether Developer identifies a person, user category, organizational function, Operational Role, or another kind of actor.

- ☐ Distinguish **Developer** from **DevSecOps Engineer**, **Platform Engineer**, **System Administrator**, **Security Engineer**, and **Compliance Officer**.

Notes for Editors

This requirement page should retain the stable requirement identifier `OR-001a`.

This page is a leaf requirement page and omits a trailing `:start` from its namespace.

The parent OR-001 page is a non-leaf page and retains a trailing `:start` in its namespace.

Changes to the Statement should preserve:

- Crucible as the responsible actor
- Developer as the applicable user category
- The applicable operational scenario as the source of the operational context
- Invocation of the operation by the Developer
- Performance of the invoked operation by Crucible

The unresolved meaning of Developer should remain recorded in the Issues section until an authoritative definition or classification resolves the issue.

Material changes should receive review and should update the verification criteria, related architecture sections, source records, and Issues section.

To reference this requirement Statement from another wiki page, insert:

```
{{section>dido:02-crusible:99-annexes:annex-c-requirements:02-operational-requirements:or-001:or-001a#Statement&noheader&nofooter&noeditbtn}}
```

Do not rename this page after an external citation unless a redirect or move plan is in place.
