Archive Manifest

Revision

Version 2
1/27/22 8:19 AM

SME

Charles Wilson

Abstract

This document describes the process to create a manifest of archived elements needed to reproduce the product.

Group / Owner

DevOps / Information Systems Security Developer

Motivation

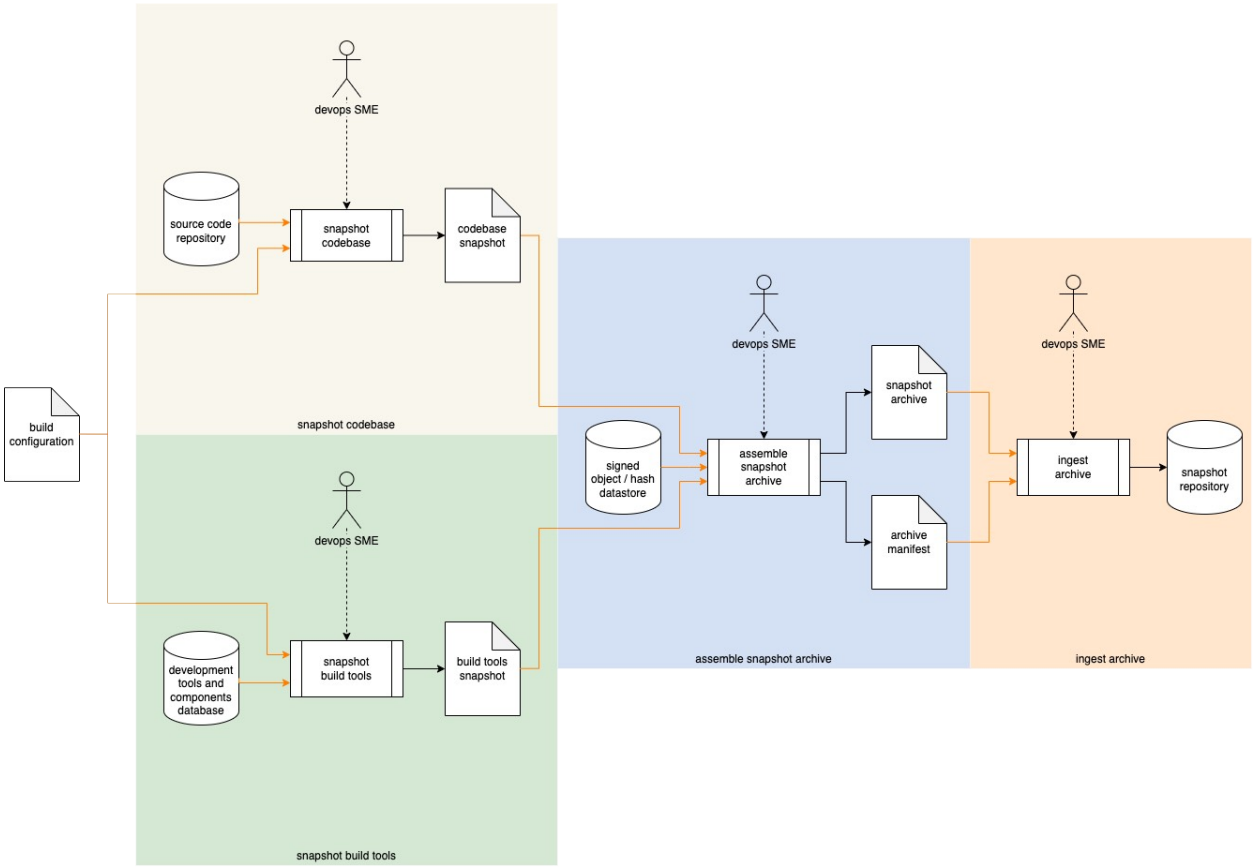
This document is motivated by the need to have formal processes in place tracking the tools used and products generated in creation of safety-critical, cyber-physical systems for certification of compliance to standards such as **ISO 21434** and **ISO 26262**.

License

Overview

It is critical that all elements and information necessary to regenerate a product are archived and that a manifest of these materials is created. This serves to ensure that the product release can be updated should the need arise and provides a mechanism to allow for the quick determination of vulnerability for any of the elements used in the product's creation.

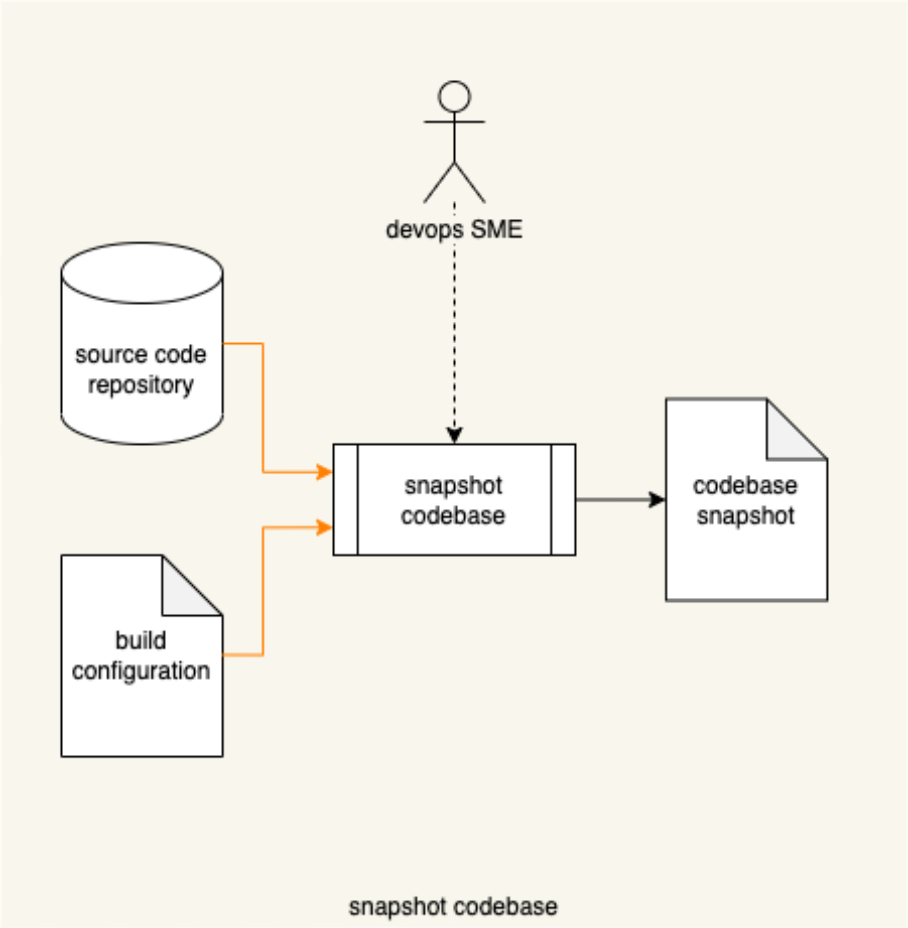
The following diagram shows the workflow to be used.



Process

Snapshot Codebase

Inputs	Source code repository Build configuration
Outputs	Codebase snapshot
Participants	None



Using the **build configuration**, the **source code repository** is accessed to generate a **codebase snapshot**.

Note: Depending on the implementation, this activity may be performed either by a devops SME or via automation.

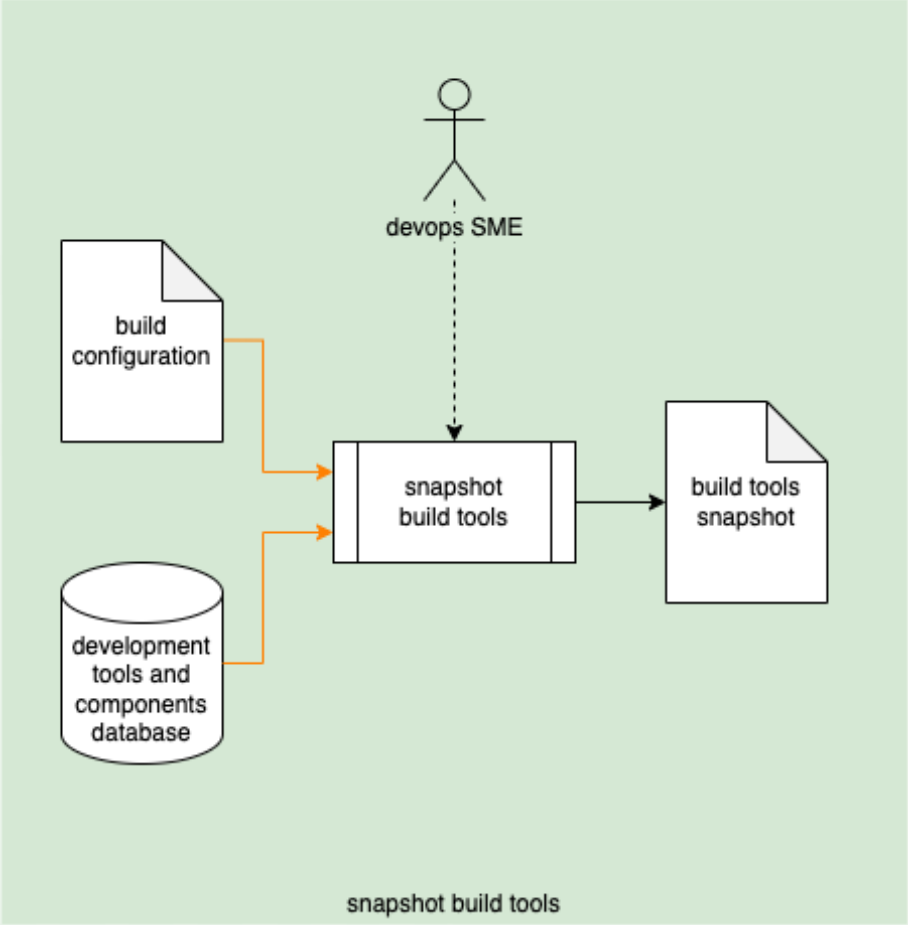
Note: The conformation (full repo vs. tip) of the snapshot is the responsibility of the organization.

Note: The codebase snapshot is understood to include associated data (such as configuration files).

Note: The snapshot may take the form of an actual copy of the code used to produce the build or permalinks to the same.

Snapshot Build Tools

Inputs	Development tools and components database Build configuration
Outputs	Build tools snapshot
Participants	None



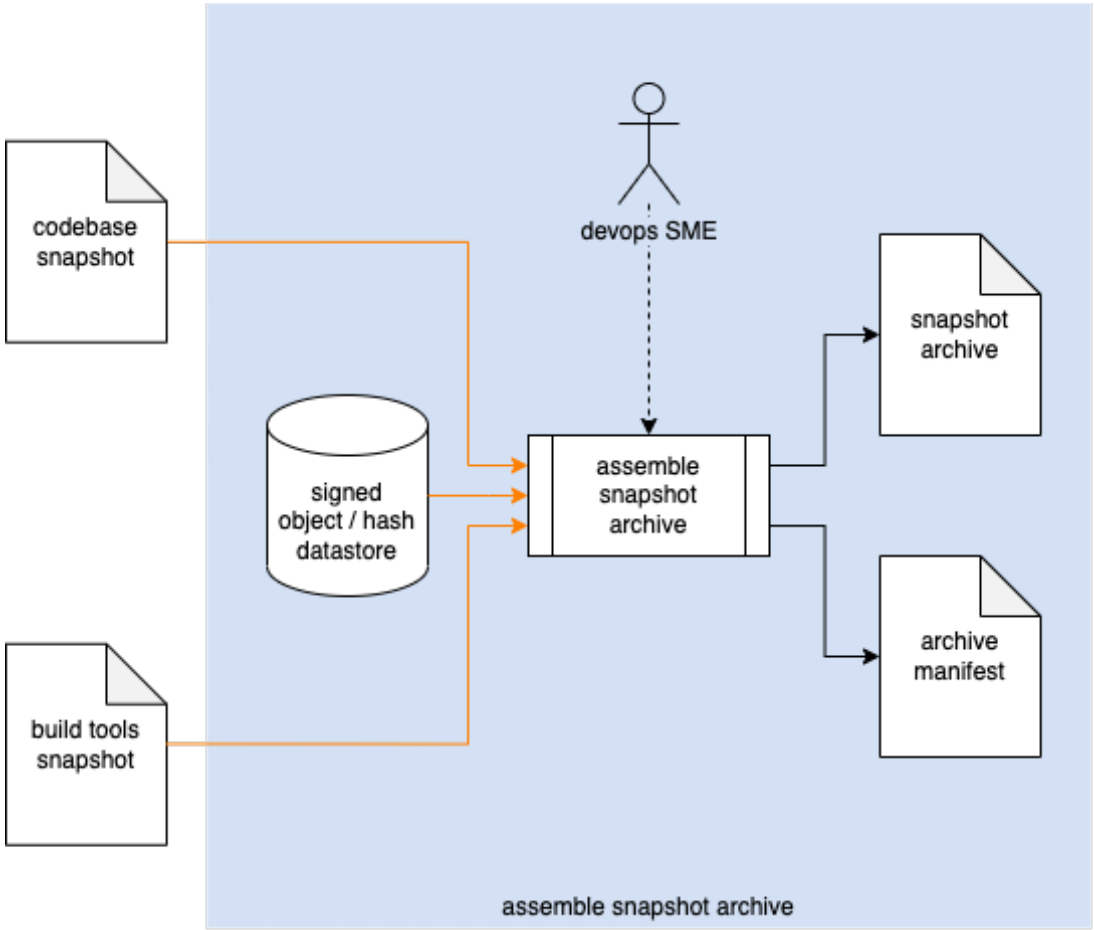
Using the **build configuration**, the **development tools and components database** (established in **List of Approved Tools** [1]) is accessed to generate a **build tools snapshot**.

Note: Depending on the implementation, this activity may be performed either by a devops SME or via automation.

Note: The snapshot may take the form of an actual copy of the tools used to produce the build or permalinks to the same.

Assemble Snapshot Archive

Inputs	Codebase snapshot Build tools snapshot
Outputs	Snapshot archive Archive manifest
Participants	None

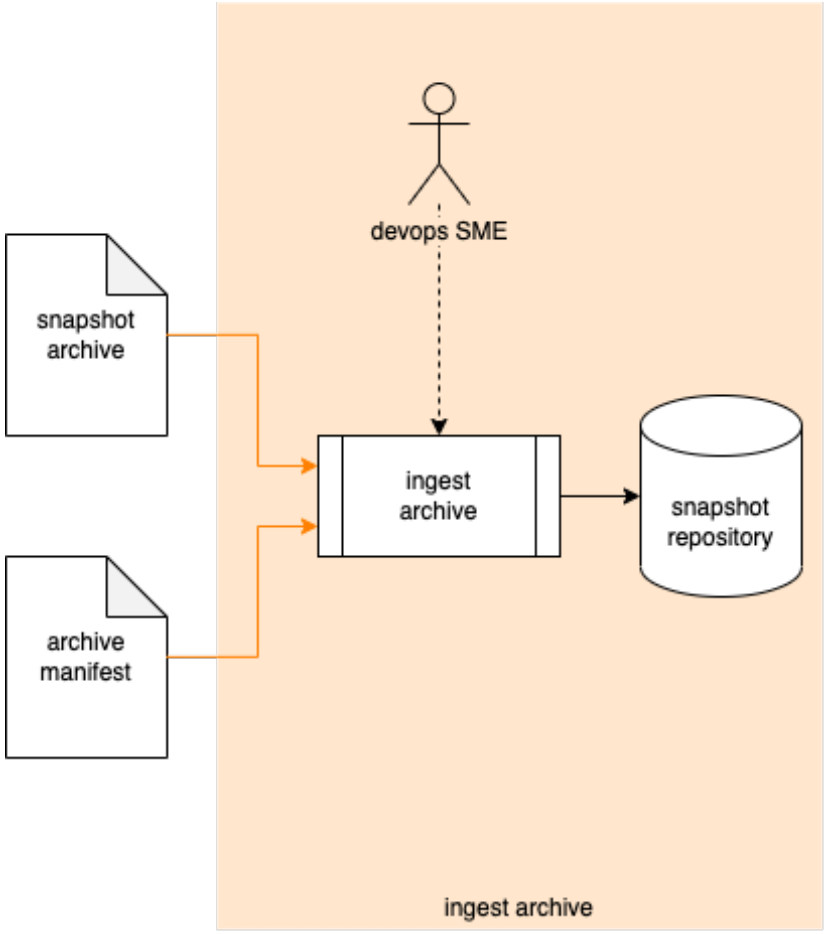


The **codebase snapshot**, **build tools snapshot**, along with the corresponding build product (extracted from the **signed object / hash datastore** [2]) are consolidated into a single **snapshot archive**. An **archive manifest** is generated.

Note: Depending on the implementation, this activity may be performed either by a devops SME or via automation.

Ingest Archive

Inputs	Snapshot archive Archive manifest
Outputs	Snapshot repository
Participants	None



The **snapshot archive** and **archive manifest** are associated and stored in the **snapshot repository**.

Note: Depending on the implementation, this activity may be performed either by a devops SME or via automation.

References

1. **List of Approved Tools** (AVCDL secondary document)
2. **Release Integrity Plan** (AVCDL secondary document)
3. The Case for Software Bill of Materials [video 37m]
   http://video.sonatype.com/watch/k1q2hYfAussHmetReM3Jbm
4. ISO 19770-2:2015 **Information technology - IT asset management - Part 2: Software identification tag**
   https://www.iso.org/standard/65666.html
5. NIST IR 8060 **Guidelines for the Creation of Interoperable Software Identification (SWID) Tags**
   https://nvlpubs.nist.gov/nistpubs/ir/2016/NIST.IR.8060.pdf