

Command	Explanation
<code>zarf COMMAND</code>	DevSecOps for Airgap
<code>zarf bundle</code>	Zarf commands for creating, deploying, removing, pulling, and inspecting bundles
<code>zarf bundle create [DIRECTORY]</code>	Create a Zarf bundle from a given directory or the current directory
<code>zarf bundle deploy [BUNDLE]</code>	Deploy a Zarf bundle from a local file or URL (runs offline)
<code>zarf bundle inspect [BUNDLE]</code>	Display the zarf.yaml of a compiled Zarf bundle (runs offline)
<code>zarf bundle pull [OCI_REF]</code>	Pull a Zarf bundle from a remote reigstry and save to the local file system
<code>zarf bundle remove [BUNDLE_NAME BUNDLE_TARBALL OCI_REF]</code>	Remove a Zarf bundle or sub-packages that have been deployed already
<code>zarf connect { REGISTRY   LOGGING   GIT   connect-name }</code>	Accesses services or pods deployed in the cluster
<code>zarf connect list</code>	Lists all available connection shortcuts
<code>zarf destroy --confirm</code>	Tears down Zarf and removes its components from the environment
<code>zarf init</code>	Prepares a k8s cluster for the deployment of Zarf packages
<code>zarf package</code>	Zarf package commands for creating, deploying, and inspecting packages

<code>zarf package create [ DIRECTORY ]</code>	Creates a Zarf package from a given directory or the current directory
<code>zarf package deploy [ PACKAGE ]</code>	Deploys a Zarf package from a local file or URL (runs offline)
<code>zarf package inspect [ PACKAGE ]</code>	Displays the definition of a Zarf package (runs offline)
<code>zarf package list</code>	Lists out all of the packages that have been deployed to the cluster (runs offline)
<code>zarf package publish { PACKAGE   SKELETON DIRECTORY } REPOSITORY</code>	Publishes a Zarf package to a remote registry
<code>zarf package pull REFERENCE</code>	Pulls a Zarf package from a remote registry and save to the local file system
<code>zarf package remove { PACKAGE_NAME   PACKAGE_FILE } --confirm</code>	Removes a Zarf package that has been deployed already (runs offline)
<code>zarf prepare</code>	Tools to help prepare assets for packaging
<code>zarf prepare find-images [ PACKAGE ]</code>	Evaluates components in a zarf file to identify images specified in their helm charts and manifests
<code>zarf prepare generate-config [ FILENAME ]</code>	Generates a config file for Zarf
<code>zarf prepare patch-git HOST FILE</code>	Converts all .git URLs to the specified Zarf HOST and with the Zarf URL pattern in a given FILE. NOTE: This should only be used for manifests that are not mutated by the Zarf Agent Mutating Webhook.

<code>zarf prepare sha256sum { FILE   URL }</code>	Generates a SHA256SUM for the given file
<code>zarf tools</code>	Collection of additional tools to make airgap easier
<code>zarf tools archiver</code>	Compresses/Decompresses generic archives, including Zarf packages
<code>zarf tools archiver compress SOURCES ARCHIVE</code>	Compresses a collection of sources based off of the destination file extension.
<code>zarf tools archiver decompress ARCHIVE DESTINATION</code>	Decompresses an archive or Zarf package based off of the source file extension.
<code>zarf tools clear-cache</code>	Clears the configured git and image cache directory
<code>zarf tools download-init</code>	Downloads the init package for the current Zarf version into the specified directory
<code>zarf tools gen-key</code>	Generates a cosign public/private keypair that can be used to sign packages
<code>zarf tools gen-pki HOST</code>	Generates a Certificate Authority and PKI chain of trust for the given host
<code>zarf tools get-creds</code>	Displays a Table of credentials for deployed components. Pass a component name to get a single credential
<code>zarf tools kubectl</code>	Kubectl command. See <a href="https://kubernetes.io/docs/reference/kubectl/overview/">https://kubernetes.io/docs/reference/kubectl/overview/</a> for more information.
<code>zarf tools monitor</code>	Launches a terminal UI to monitor the connected cluster using K9s.

<code>zarf tools registry</code>	Tools for working with container registries using go-containertools
<code>zarf tools registry catalog REGISTRY</code>	List the repos in a registry
<code>zarf tools registry copy SRC DST</code>	Efficiently copy a remote image from src to dst while retaining the digest value
<code>zarf tools registry login [OPTIONS] [SERVER]</code>	Log in to a registry
<code>zarf tools registry ls REPO</code>	List the tags in a repo
<code>zarf tools registry pull IMAGE TARBALL</code>	Pull remote images by reference and store their contents locally
<code>zarf tools registry push PATH IMAGE</code>	Push local image contents to a remote registry
<code>zarf tools sbom</code>	Generates a Software Bill of Materials (SBOM) for the given package
<code>zarf tools sbom attest --output [FORMAT] &lt;IMAGE&gt;</code>	Generate an SBOM as an attestation for the given [SOURCE] container image
<code>zarf tools sbom convert [SOURCE-SBOM] -o [FORMAT]</code>	Convert between SBOM formats
<code>zarf tools sbom login [OPTIONS] [SERVER]</code>	Log in to a registry
<code>zarf tools sbom packages [SOURCE]</code>	Generate a package SBOM

<code>zarf tools sbom version</code>	show the version
<code>zarf tools wait-for { KIND   PROTOCOL } { NAME   SELECTOR   URI } { CONDITION   HTTP_CODE }</code>	Waits for a given Kubernetes resource to be ready
<code>zarf version</code>	Shows the version of the running Zarf binary