

Release Checklist

The following checklist helps to identify the series of actions that need to be done leading up to a normal EdgeX release. The examples below were taken from the Kamakura release in May 2022.

1. One week prior to code freeze
 - a. Stop merged 3rd party Dependabot PRs (unless critical bug fixes)
 - b. Pin 3rd party docker images to patch level in compose builder
2. Code Freeze
3. Update [edgex-global-pipelines](#) to enable new release name. ([example](#))
4. Update all modules & services to use latest go-mod dev tags
 - a. Update go-mod-secrets with latest version of go-mod-core-contracts ([example](#))
 - b. Update go-mod-bootstrap with latest versions of other go-mods ([example](#))
 - c. Update SDKs with latest versions of go-mods (Examples: [app-functions-sdk-go](#), [device-sdk-go](#))
 - d. Update all services with latest versions of their go-mods and SDKs ([example](#))
5. Verify TAF Tests all pass ([Jenkins Pipelines](#))
6. Perform Manual Testing
7. Fix bugs if needed, repeat steps #3 thru #5 as necessary.
8. Generate and prepare changelogs. Ask WG Chairs to make necessary changes if necessary. ([example](#))
9. Release go-mods
 - a. Verify release jobs have next **semver** for main set appropriately.
 - b. Update changelogs for Go Mods ([example](#))
 - c. Release go-mod-configuration, go-mod-core-contracts, go-mod-registry, ([example](#))
 - d. Update go-mod-secrets and go-mod-messaging with release version of go-mod-core-contracts ([example](#))
 - e. Release go-mod-secrets and go-mod-messaging ([example](#))
 - f. Update go-mod-bootstrap with release version of other go-mods ([example](#))
 - g. Release go-mod-bootstrap ([example](#))
10. Release SDKs ([app-functions-sdk-go](#), [device-sdk-go](#))
 - a. Verify release jobs have next **semver** for main set appropriately.
 - b. Update SDKs with released versions of go-mods ([app-functions-sdk-go](#), [device-sdk-go](#))
 - c. Update changelogs for SDKs ([example](#))
 - d. Update App Template go.mod with SDK version ([example](#))
 - e. Release SDKs ([example](#))
11. Release [edgex-go](#)
 - a. Verify release jobs have next **semver** for main set appropriately.
 - b. Update with released versions of go-mods ([example](#))
 - c. Update changelog ([example](#))
 - d. Release Core/Support/Security services ([example](#))
12. Release App Services ([app-service-configurable](#), [app-rfid-llrp-inventory](#) & [app-record-replay](#))
 - a. Verify release jobs have next **semver** for main set appropriately.
 - b. Update with released version of SDK and go-mods ([example](#))
 - c. Update changelogs for App Services ([example](#))
 - d. Release App Services ([example](#))
13. Release Device Services (Confirm which devices with Device Services WG)
 - a. Verify release jobs have next **semver** for main set appropriately.
 - b. Update with released versions of SDK and go-mods ([device-mqtt-go example](#))
 - c. Update changelogs for Device Services ([device-mqtt-go example](#))
 - d. Release Device Services ([example](#))
14. Release UI ([edgex-ui-go](#))
 - a. Verify release jobs have next **semver** for main set appropriately.
 - b. Update with released versions of go-mods (Ex: UI [part1](#), [part2](#), [part3](#))
 - c. Update changelog ([ui-example](#))
 - d. Release UI([example](#))
15. Release Compose files
 - a. Add any new services to the compose job code.
 - b. Run CI Job ([Jenkins Job](#))(Run without parameters for dry-run.)
 - i. Creates commit / branch ([example](#))
 - ii. Removes Jenkins file
 - c. Update README's ([example](#))
 - d. Make any needed version tweaks ([example](#))
 - e. Test that all images can be pulled and started appropriately
 - i. make run no-secty [all service options]
 - f. Merge tweaks PR ([example](#))
 - g. Tag commit on newly created branch
16. Release Helm Chart
 - a. Update Helm Chart to use current release images ([example](#))
 - b. Tag edgex-helm repo
17. Release examples ([edgex-examples](#))
 - a. App Service examples use current released SDK ([example](#))
 - b. Update security examples for released images & SDKs
 - c. Tag edgex-examples repo ([example](#))
18. Release Docs ([edgex-docs](#))
 - a. Update example links to use release tag ([example](#))
 - b. Release/Run CI Job ([example](#))
 - c. Merge PR's from CI Job ([example](#))

19. Update Docker image descriptions
 - a. Update Docker Hub descriptions.
 - i. Do not manually generate overviews. Jenkins will do that in the next step. ([example](#))
 - b. Manually trigger build to publish changes ([Jenkins Job](#))
20. Tag TAF
 - a. TAF team switches to release branch ([example](#))
 - b. Tag TAF ([example](#))
21. Update links **example** links above to current PRs
22. Verify semver on **main** branch is set to next expected release version with "-dev.1" appended
23. Prepare edgex-compose for next release
 - a. Update compose builder versions for 3rd party dependencies to not pin to patch level
 - b. Also update release name in the **gen-header** file to be the upcoming name.
 - c. Run **make build** to regenerate all the WIP compose files.
24. Update swagger versions to expected next release number.
 - a. edgex-go ([example](#))
 - b. app-functions-sdk-go ([example](#))
 - c. device-sdk-go ([example](#))
 - d. device-onvif-camera ([example](#))
 - i. Change version in JSON swagger generation config file
 - ii. Run `make openapi` to generate swagger
25. Create Dev tags for next minor/major release
 - a. Update all semver to create next dev tags for the new minor/major version
 - b. If next release is a major release,
Update all modules to be the next major version - ie v2 v3, update all import statements for the new major version, and update all mocks in the following order
 - i. Update go-mod-configuration, go-mod-core-contracts, go-mod-registry
 - ii. Update go-mod-secrets and go-mod-messaging (Note that the mocks in go-mod-messaging must be updated to ensure that the nats.go package is in the import statement)
 - iii. Update go-mod-bootstrap
 - iv. Update all go-sdks to be the next major version - ie v2 v3, update all import statements for the new major version, and update all mocks
 1. Update go.mod to use latest dev go-mod-bootstrap
 - v. Update edgex-go import statements for its dependent EdgeX modules
 1. Update go.mod to use latest dev go-mod-bootstrap
 - vi. Update all app services' import statements for its dependent EdgeX modules
 1. Update go.mod to use latest dev App SDK
 - vii. Update all device services' import statements for its dependent EdgeX modules
 1. Update go.mod to use latest dev Device SDK
 - c. If next release is a minor release,
update to generate and use initial dev tags for next minor release
 - i. Update go-mod-configuration, go-mod-core-contracts, go-mod-registry
 1. simple README tweak to dev tag generated
 - ii. Update go-mod-secrets and go-mod-messaging to use latest dev go-mod tags
 - iii. Update go-mod-bootstrap to use latest dev go-mod tags
 - iv. Update all go-sdks to use latest dev go-mod-bootstrap
 - v. Update edgex-go to use latest dev go-mod-bootstrap
 - vi. Update all app services to use latest dev App SDK
 - vii. Update all device services to use latest dev Device SDK
26. Make adjustments to the checklist for items that have changed since last release