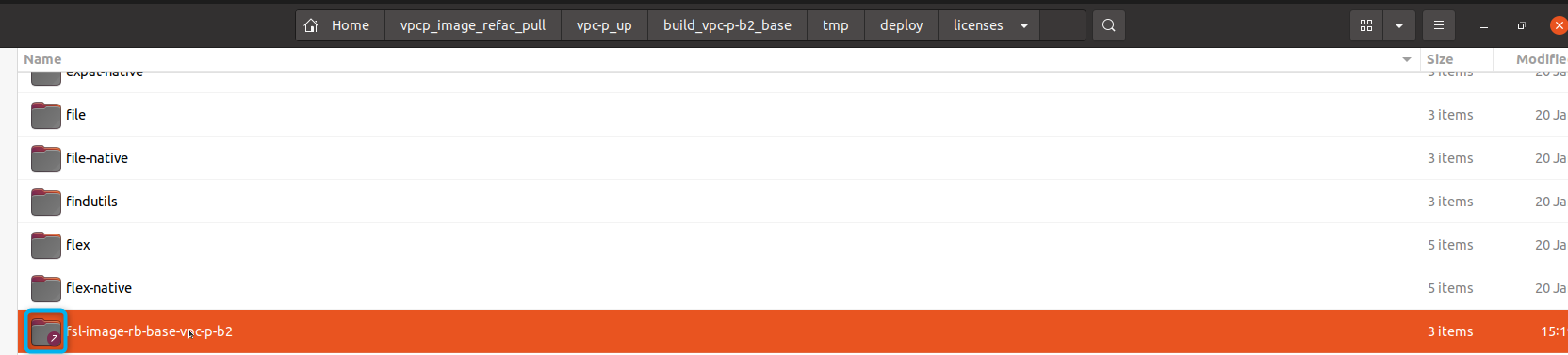
Base Image Refactoring

# Package List Location

* All package lists are available at: vpc-p\_up/build\_folder/tmp/deploy/licenses/fsl-image



# Package Manifest

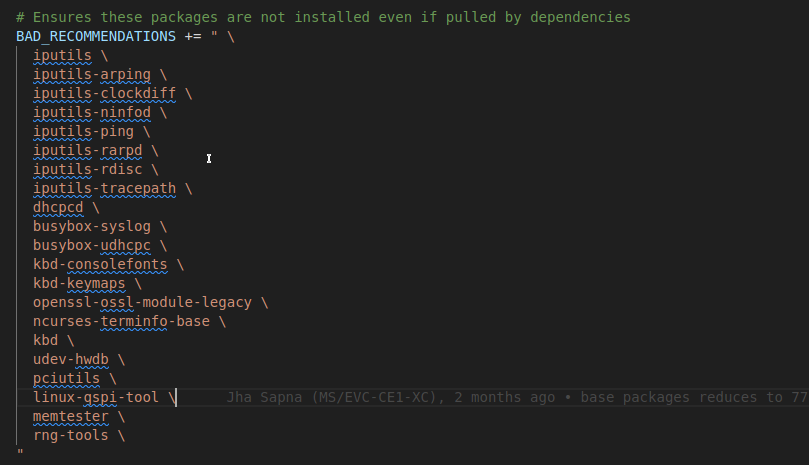
* The file 'package.manifest' provides a list of the current packages present in your image.

# Options to Reduce Packages

To reduce the number of packages in your image, you can use the following directives:

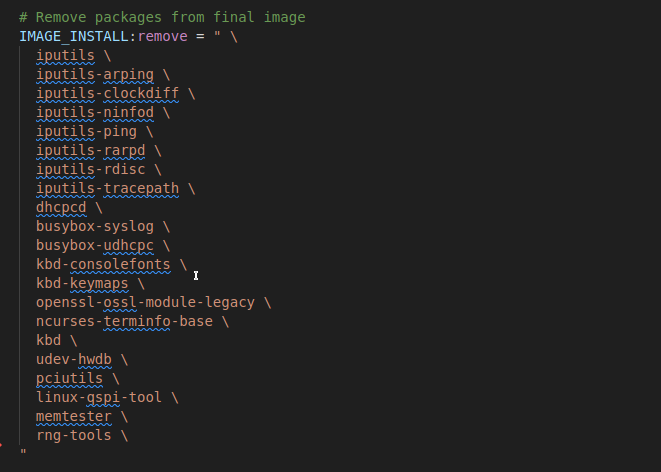
## 1. BAD\_RECOMMENDATIONS

* Purpose:
* This variable is used to flag a set of packages as 'bad recommendations'. In Yocto, some recipes recommend extra packages even if they are not strictly required. By marking these packages as bad recommendations, you instruct the build system to ignore or drop them, even if suggested by another package’s metadata.
* How It Works:
* It influences the dependency resolution system to prevent these packages from being automatically added based on recommended relationships. It doesn't remove them directly but ensures they are not suggested in the first place.



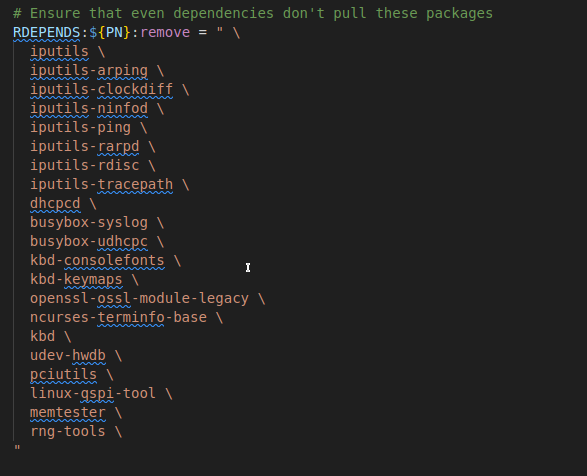
## 2. IMAGE\_INSTALL:remove

* Purpose:
* This directive explicitly removes the listed packages from the image installation list. Regardless of any other factors, if a package appears in IMAGE\_INSTALL, the build system will strip it out of the final image.
* How It Works:
* During the image creation phase, the specified packages are purged from the installation list, ensuring that even packages added earlier do not end up in the final image.



## 3. RDEPENDS:${PN}:remove

* Purpose:
* This directive targets the runtime dependencies (RDEPENDS) for the package defined by ${PN} (i.e., the current package or image recipe). It ensures that even if one of the unwanted packages is indirectly pulled in as a dependency, it is removed.
* How It Works:
* By removing these packages from RDEPENDS, you guarantee that the final runtime dependency list does not include any undesired packages, preventing them from being installed when the image runs.



# Build and Verification

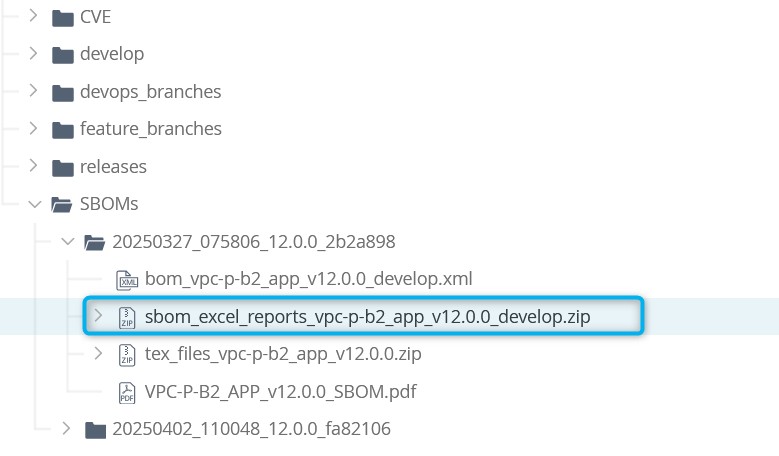
* After applying the above directives, run the build process and verify if it completes successfully.
* If the build fails due to packages specified in BAD\_RECOMMENDATIONS, IMAGE\_INSTALL:remove, or RDEPENDS:${PN}:remove, consider removing the dependency package or adjusting the package list. For example, if 'iproute-devlink' depends on the kernel, do not remove the kernel; simply remove 'iproute-devlink' from the list.
* Once the build is successful, review the updated 'package.manifest' to ensure that the unwanted packages have been removed.
* Finally, flash the new binaries and confirm that flashing is successful by verifying login, release version, ping tests, and the status of the COM port.

# Excel Link –

[Base\_Image\_refactoring](https://bosch.sharepoint.com/:x:/r/sites/msteams_3731583/_layouts/15/Doc.aspx?sourcedoc=%7B03164615-FDB8-4212-9CD6-F50ECD757F2A%7D&file=base_packages_updated.xlsx&action=default&mobileredirect=true)

# SBOM Path–

* Inside the artifactory images folder, you will see a SBOM folder
* Download the zip file, inside the zip file you will see a excel file named HMC\_SBOM\_REPORT.xlsx
* This file will have all the packages for the particular image version, with all the dependencies on and dependencies for



HMC\_SBOM\_REPORT.xlsx

