# RXPatch overview

The patcher takes two input directories, OLD and NEW. NEW contains all files that should be available after installation, i.e. the new version. OLD includes all files of the previous version. These files are used for incremental patching and removal. Any files in the OLD directory that are not in the NEW directory will be removed.

A patch can be created using RXPatch create args.json, where args.json is a file specifying the OLD, NEW, and PATCH directory. (An example is in the appendix.) The PATCH directory is the output directory and should be empty before creation of the patch. After creating the patch, all files in this directory should be uploaded to a HTTP or FTP location.

A patch can be applied by setting it up for the updater, or by invoking RXPatch apply\_web ... to install it from an update URL or rxpatch apply\_filesystem ... to install it from a local path.

A special version file, located at <http://renegade-x.com/launcher_data/version.json> describes the latest versions of the game and the launcher. (An example is in the appendix.) The launcher downloads this file to see if a new version is available.

# Preparing a typical patch

There are some things to note when creating a patch.

1. Prepare a directory with all files from the old install. These files will be replaced or removed by the patcher (OLD).
   1. Be sure to include the UDK\*.ini configuration files, as these should be removed by the patcher.
   2. However, do not include the UDKSystemSettings.ini file, as this file should persist between patches.
   3. Remove Binaries/InstallInfo.xml, as it depends on the user’s machine.
   4. Do not include the launcher, as the launcher cannot be patched simultaneously with the game. The launcher has a separate updater.
   5. Make sure that the directory does not contain extraneous files, and only the files that you wish to patch.
2. Prepare a directory with all files that should be installed (NEW).
   1. Do not include the launcher, for the same reason as above.
   2. Do not include the UDK\*.ini files as they are automatically generated on the client on the first run. (Do not include the UDKSystemSettings.ini file either.)
   3. Remove Binaries/InstallInfo.xml, see step 1.c.
   4. Make sure that the DefaultRenegadeX.ini contains the correct version info. In particular:  
      GameVersion=”Open Beta 4”  
      GameVersionNumber=400  
      GameVersion can be anything, and is used only for displaying the version to the player. GameVersionNumber is the number that is used to check for updates, and should be higher than any previously distributed version.
3. Run RXPatch create args.json, where args.json specifies the paths to the OLD and NEW directory that were prepared above, and the desired output (PATCH) directory. This might take a while. RXPatch does not report on its progress[[1]](#footnote-1).
4. Upload everything in the PATCH directory to a location that is accessible via HTTP or FTP. It is a good idea to name this directory after the new version, e.g. [http://renegade-x.com/patches/Open Beta 4](http://renegade-x.com/patches/Open%20Beta%204)
5. Update <http://renegade-x.com/launcher_data/version.json> to match the version info specified in step 2.c and the URL of the directory that was uploaded to in step 4.
6. Run the launcher and see if the patch works.

# Appendix

## args.json

{

"OldPath": "c:\\rxpatch\\Open Beta 3",

"NewPath": "c:\\rxpatch\\Open Beta 4",

"PatchPath": "c:\\rxpatch\\Open Beta 4 patch",

}

## version.json

{

"launcher": {

"version\_name": "0.51",

"version\_number": 51,

},

"game": {

"patch\_url": "http://renegade-x.com/patch/Open Beta 4",

"version\_name": "Open Beta 4",

"version\_number": 400,

}

}

1. A rough estimate can be made by comparing the size of the full directory in the PATCH directory, with the size of the NEW directory. The former grows until it is approximately the size of the latter, at which point the patch is completed. [↑](#footnote-ref-1)