

Project Documentation

Repackage Software with Predefined Registry Settings.

Prepared By:

User ID- 34851

Name- Krishna Verma

Email – kv979539g@gmail.com

Activity

Objective: Repackage Software with Predefined Registry Settings.

Application Repackaging:

Application repackaging is the process of converting an existing software installer, such as an EXE, into a new, customized installer format like an MSI.

This is a key practice for IT administrators and is separate from the original software development process.

Benefits of Repackaging

- **Standardization:**
 - Repackaging allows IT to create silent, consistent installations that can be deployed to thousands of computers without manual interaction.
- **Customization:**
 - It gives administrators the ability to modify installations to meet organizational needs.
 - It can include adding or removing features, applying specific registry keys, and defining custom shortcuts or file locations.
- **Troubleshooting & Compatibility:**

- This process provides a way to fix issues with installers that may have compatibility problems with specific operating systems or enterprise deployment tools.

Registry Settings and Their Usage

The **registry** is a hierarchical database that stores low-level settings for the Windows operating system and its applications. It acts as the central nervous system for a machine.

The repackaging process allows administrators to modify an installation by applying specific registry keys or settings. This is used to meet organizational needs.

In the activity, a new user-level registry key named **MyRegistry-Project** with a value of **MyValue** will be added to the **HKCU\Software** path.

The final repackaged installer then automatically creates this customized registry key and its value upon installation.

To Perform our Activity, we are using **Advanced Installer**:

Advanced Installer is a professional and user-friendly software packaging tool for Windows.

It is primarily used by developers and IT professionals to create reliable and customized installation packages, in the MSI format.

Prerequisites:

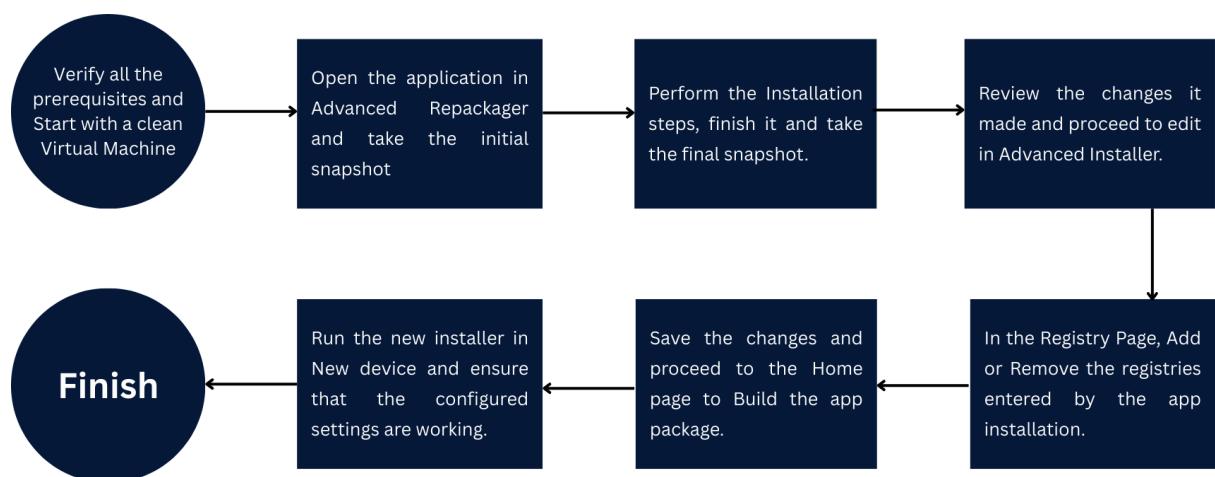
-
1. **Advanced Installer & Re-Packager** Software.

2. A clean **Virtual Machine** with the following requirements to Perform the Activity:
- CPU:** A Core 2 class processor or equivalent is the baseline.
 - RAM:** You need at least 1 GB of RAM to run the application.
 - Storage:** A minimum of 2 GB of free hard drive space is required for installation and operation.
 - Display:** The minimum supported screen resolution is 1366x768.
 - Operating System:** Advanced Installer requires **Windows 10 or newer** to run.

3. The Application you need to Customize- We will be using **CloudFlare WARP**.

4. Access to **Windows Registry Editor**.

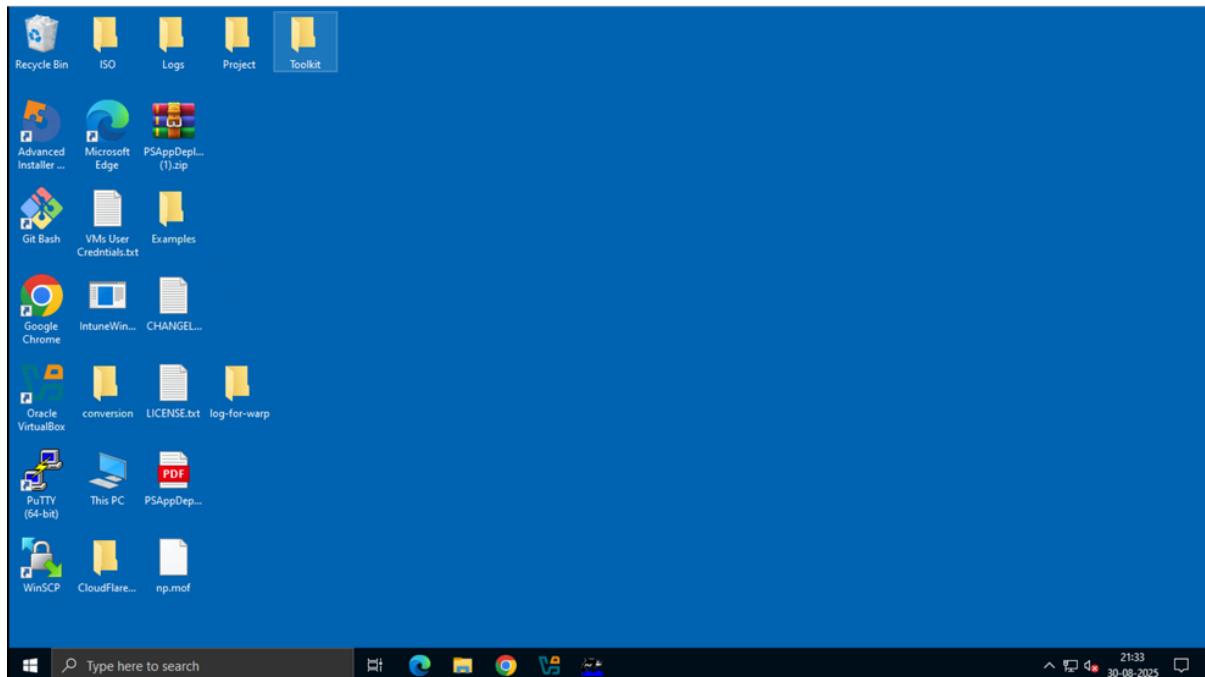
Workflow Diagram:



Execution Steps:

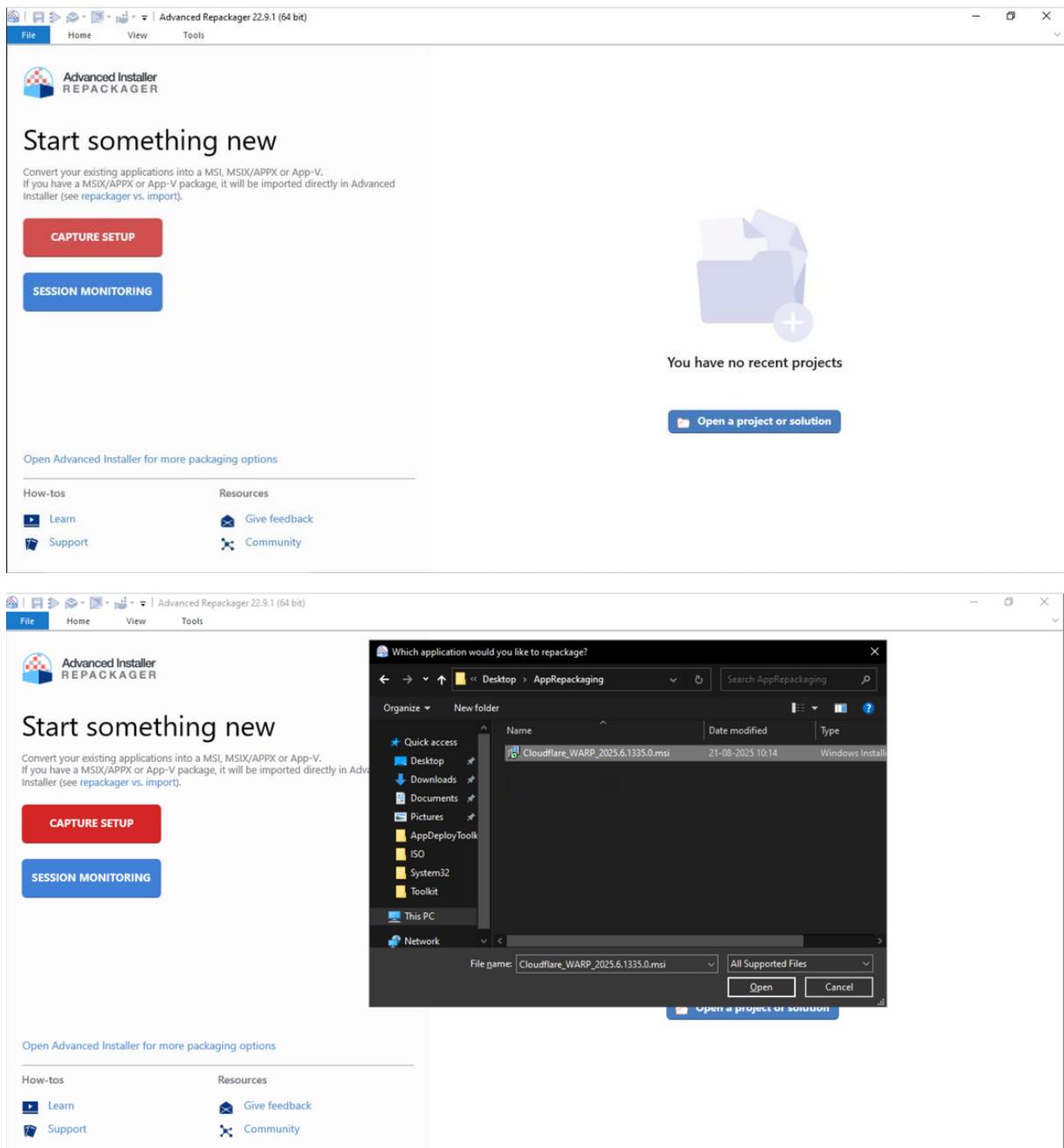
1. Preparation of the Virtual Machine

- Ensure you have a **clean virtual machine (VM)**.
- We are using the VM which is provided by our Team.



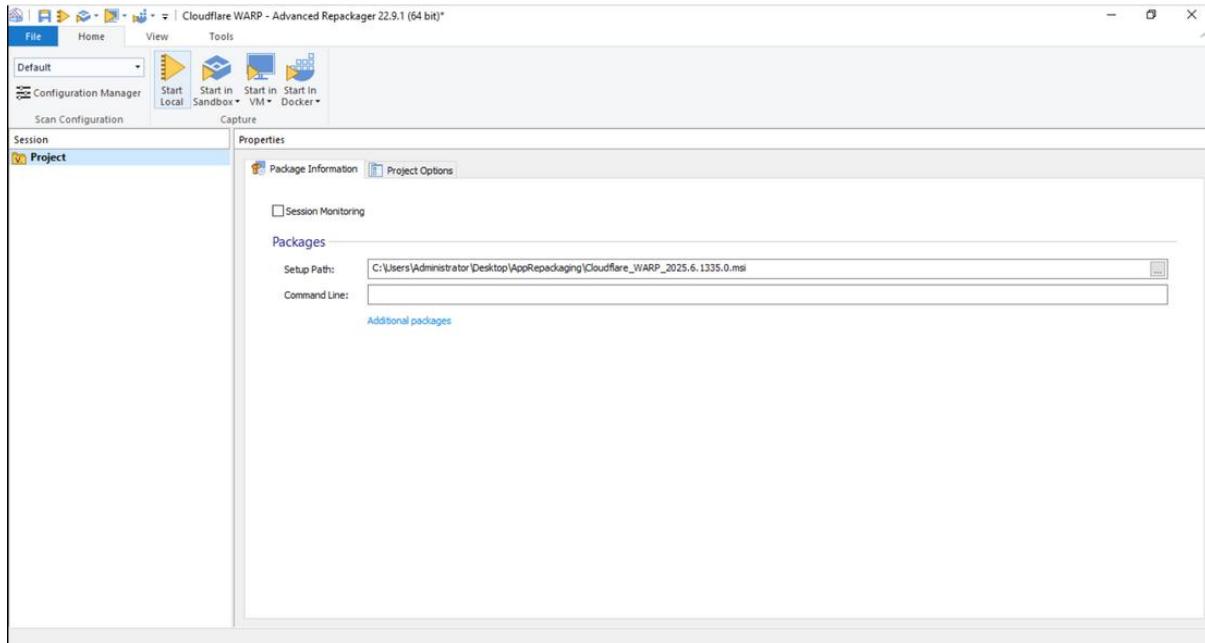
Step 2: Take the Before Snapshot

- Launch **Advanced Re-packager**.
- Select the **Capture Setup** option.
- Select the Application which you want to Customize.

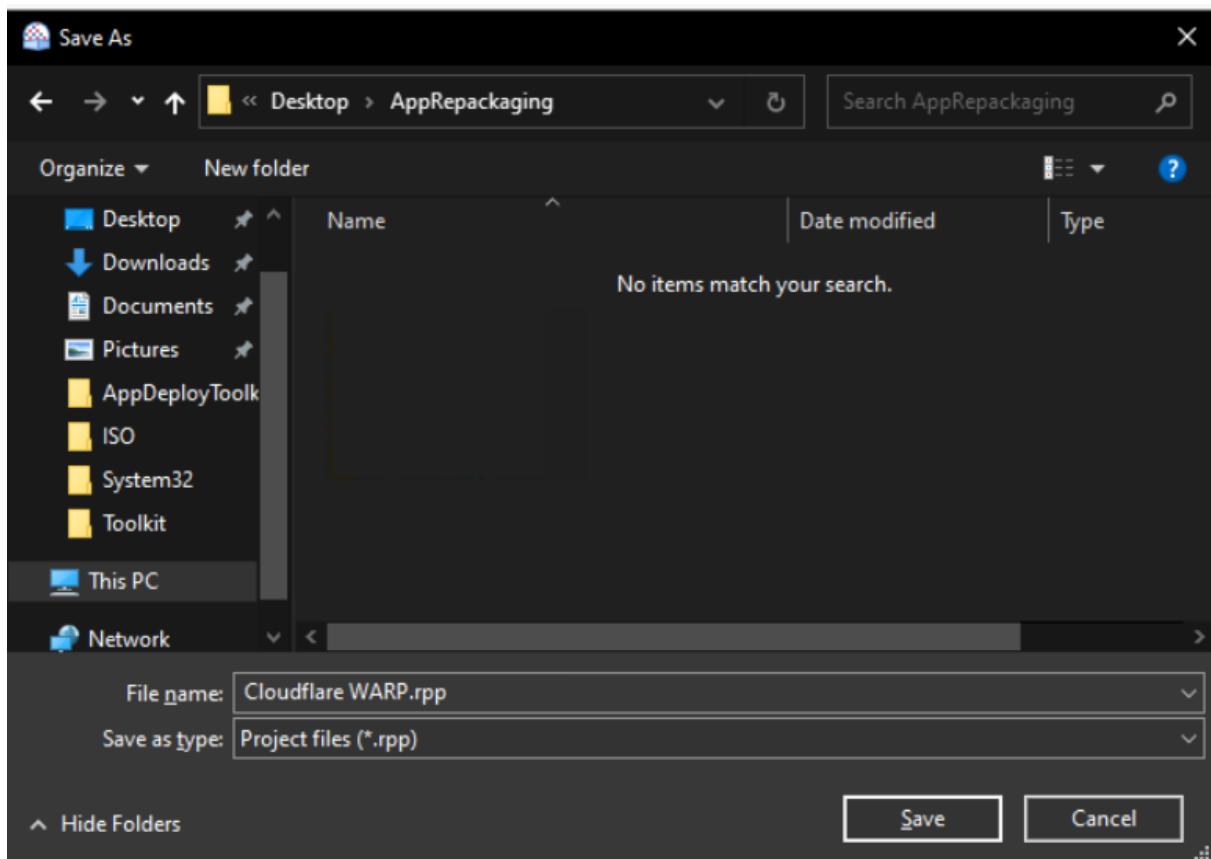


- The tool will begin by taking a **before snapshot** of the system. This process scans the file system, registry, and other system components to record the initial state.
- In the next steps, you'll be prompted to choose the repackaging environment.

- **Start Local:** Select the Setup Path and choose the Start Local option in Home Tab.



- Choose the Location where you want to place the unpacked project file. It save the Project with a .rpp extension.

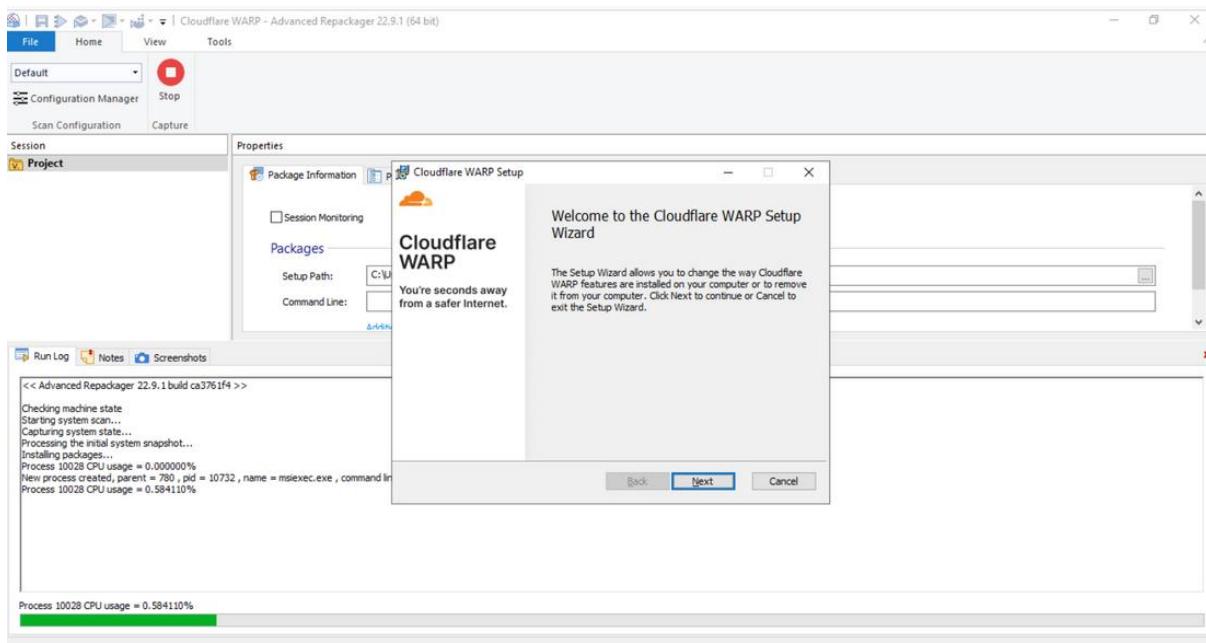


- **Start the Capture:** Click the **Start Local** button. The Repackager will take an initial snapshot of the VM's file system and registry.

3. Installation and Customization

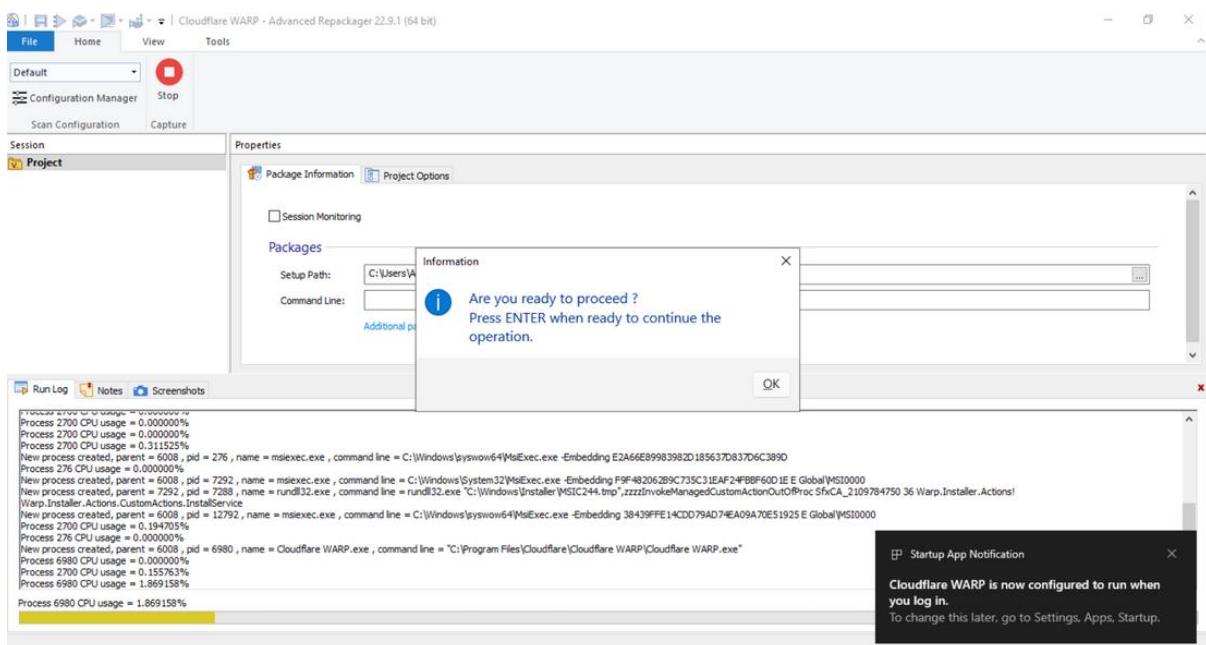
After the initial snapshot is complete, the repackaging wizard will launch the installer inside the VM.

- **Install the Application:** Go through the application's installation wizard as you normally would. **Crucially, this is where you perform the actions that you want to capture.**



- **Complete the Capture:**

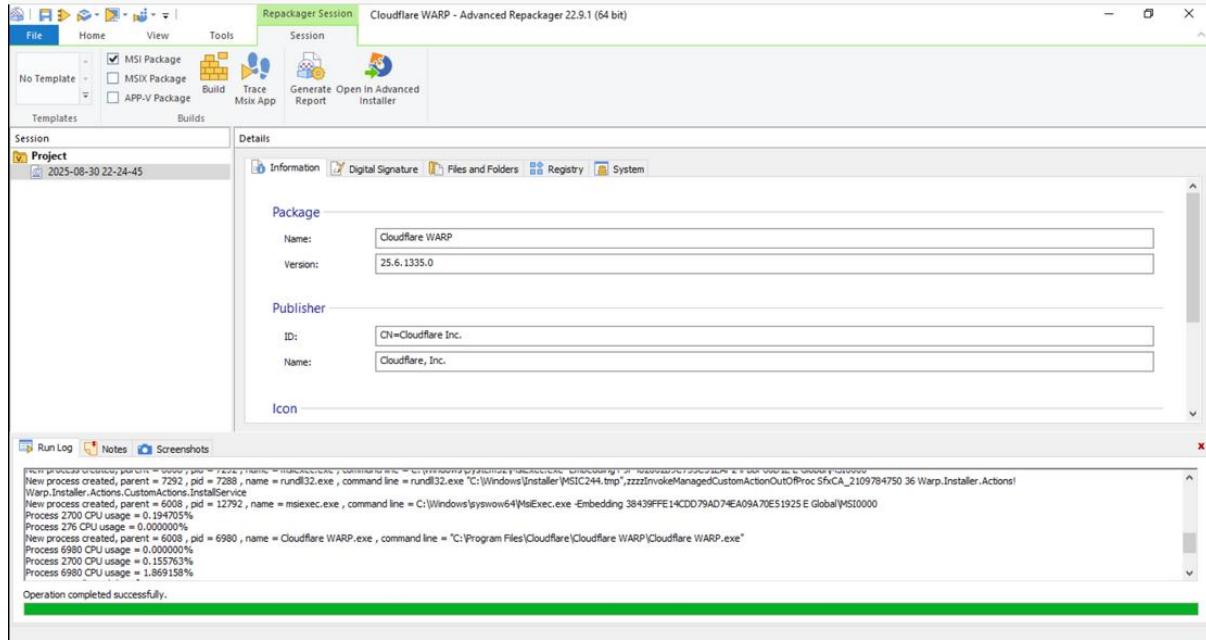
- Once the installation is finished, close the application.
- In the Repackager wizard on your host machine, click **OK** to signal that you are done.
- The Repackager will now take a second snapshot.



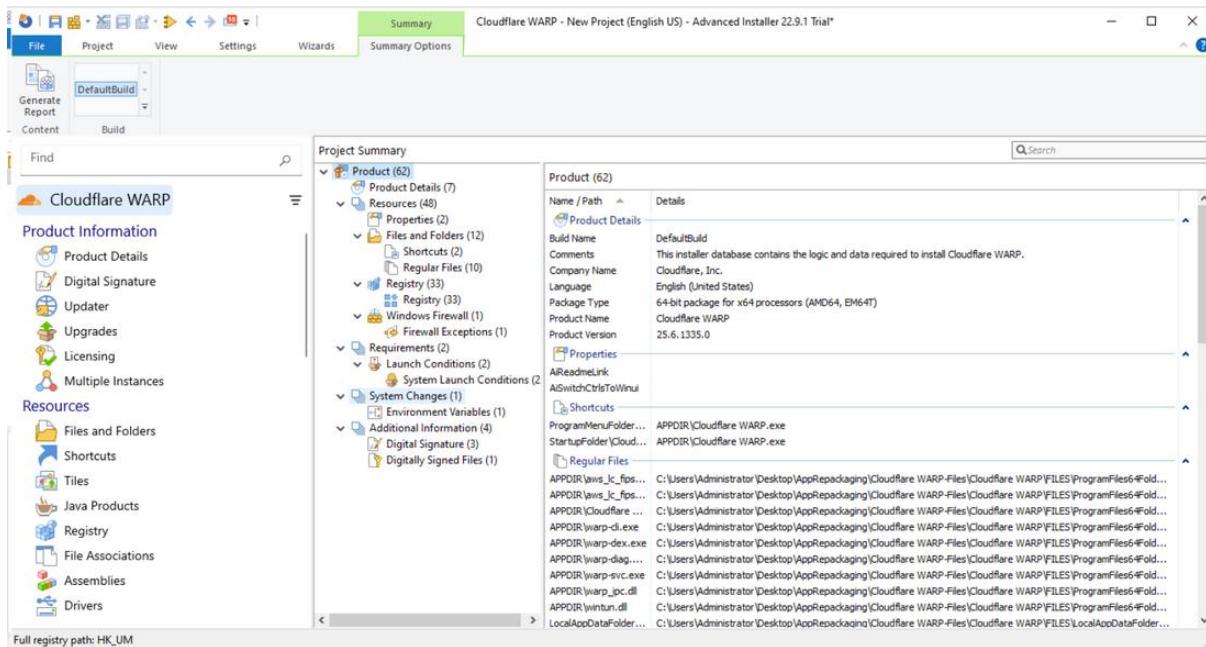
4. Refining the Project and Building the Package 🔧

The Repackager will compare the "before" and "after" snapshots and present a report of all changes.

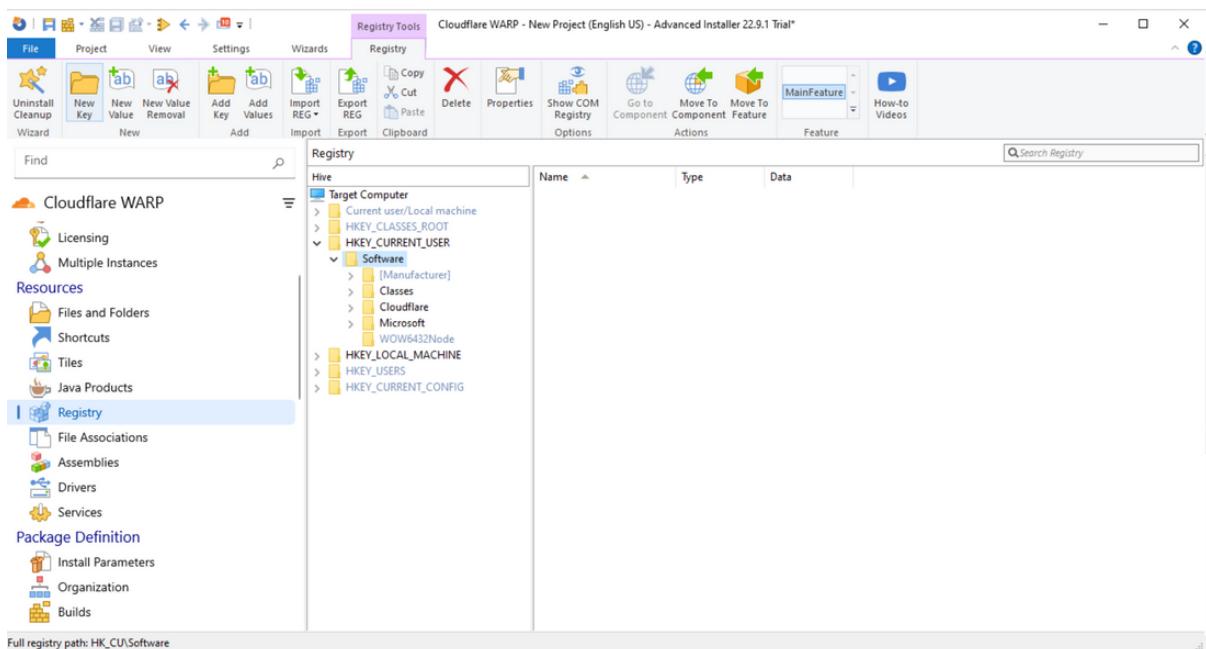
- **Review Changes:** Carefully review the captured changes. You'll see a list of new files, folders, and registry keys and values that were created or modified. This is your chance to clean up the project by removing any "noise" or unwanted changes.



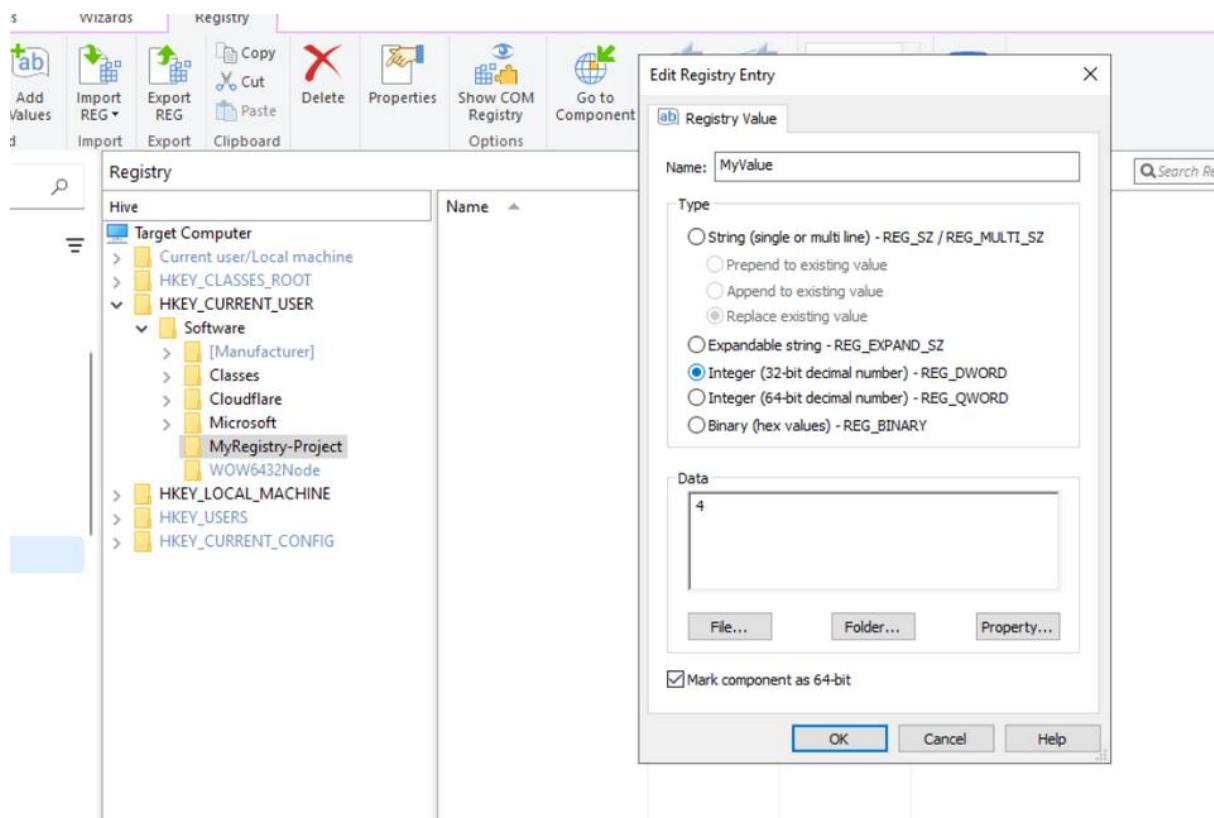
- **Open in Advanced Installer:** Once you're satisfied with the captured data, click the **Open in Advanced Installer** button. This will create a new .aip (Advanced Installer Project) file.

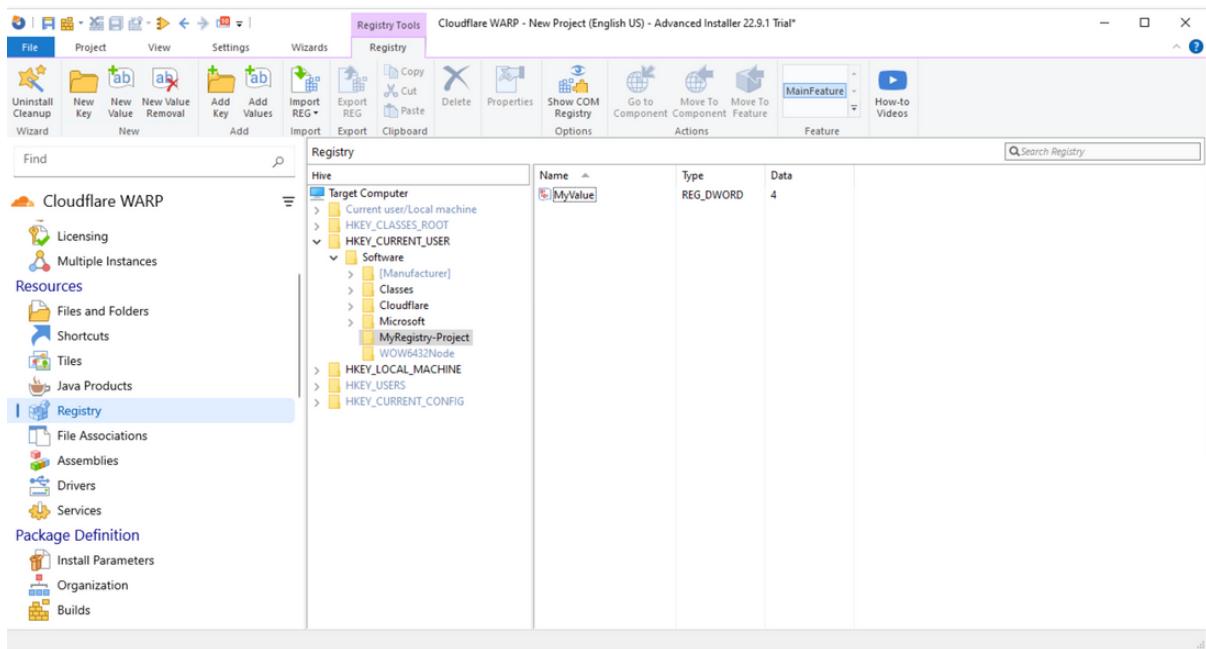


- **Refine Registry Settings:** Navigate to the **Registry** tab to review the captured registry changes.
 - You can manually Add or delete any other keys or values.



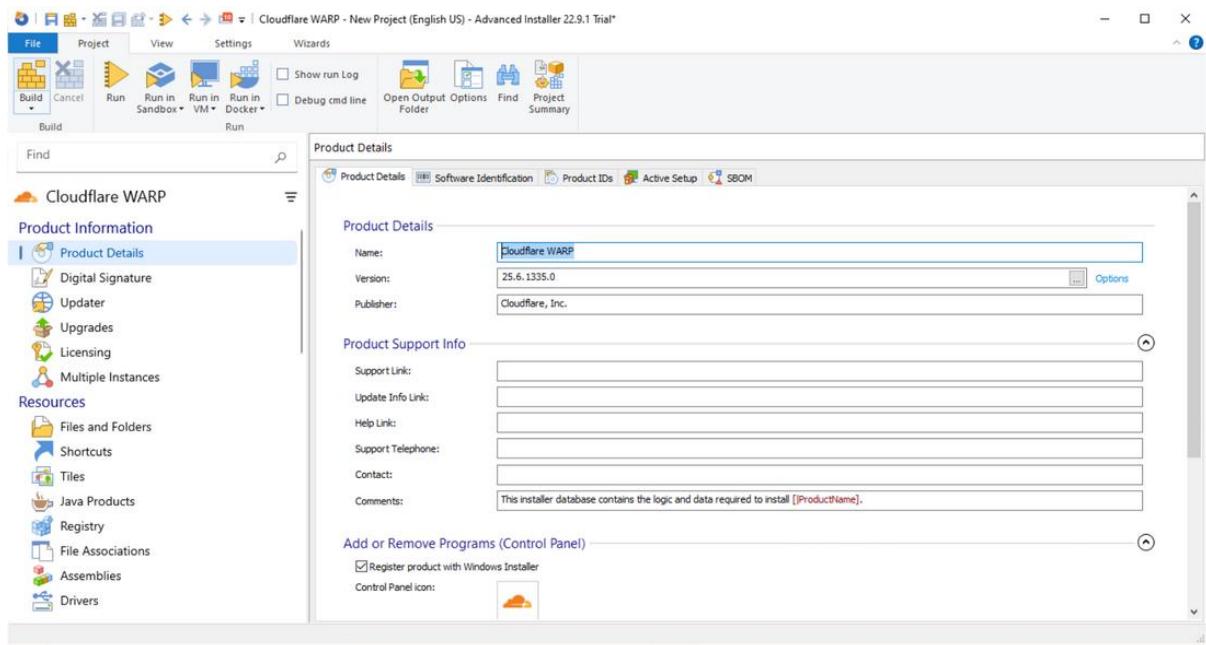
- Here, we are going to add our customized User level registry keys and values in the HKCU Key Hive.
- Navigate to the specific Hive you want to create your new key and click on **New Key** option from the **Registry Tab**.
- We have created the new Key with name **MyRegistry-Project** with Value **MyValue** and Data is **4** (Integer 32 bit decimal type) in the **HKCU \ Software** path.



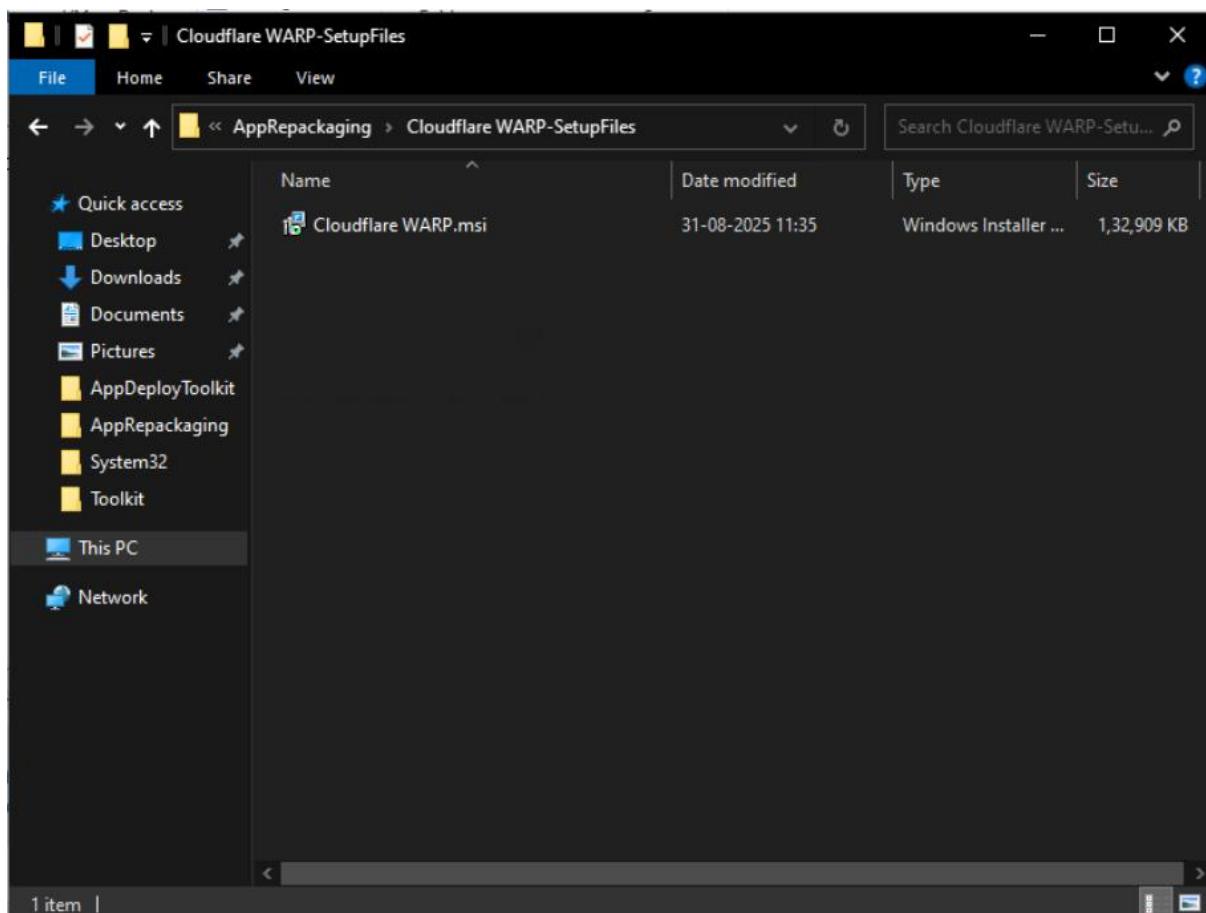


- **Build the Installer:**

- After verifying everything, goto the **Product Details** Page.
- At the **Product Details** page, click on **Build** and select the desired output format, typically an **MSI** package. Configure any build options, and then click the **Build** button.



- Select your export path and the final **MSI** Package will be build.



The final output is a new, customized installer package that includes the application and all the predefined registry settings you captured and refined.

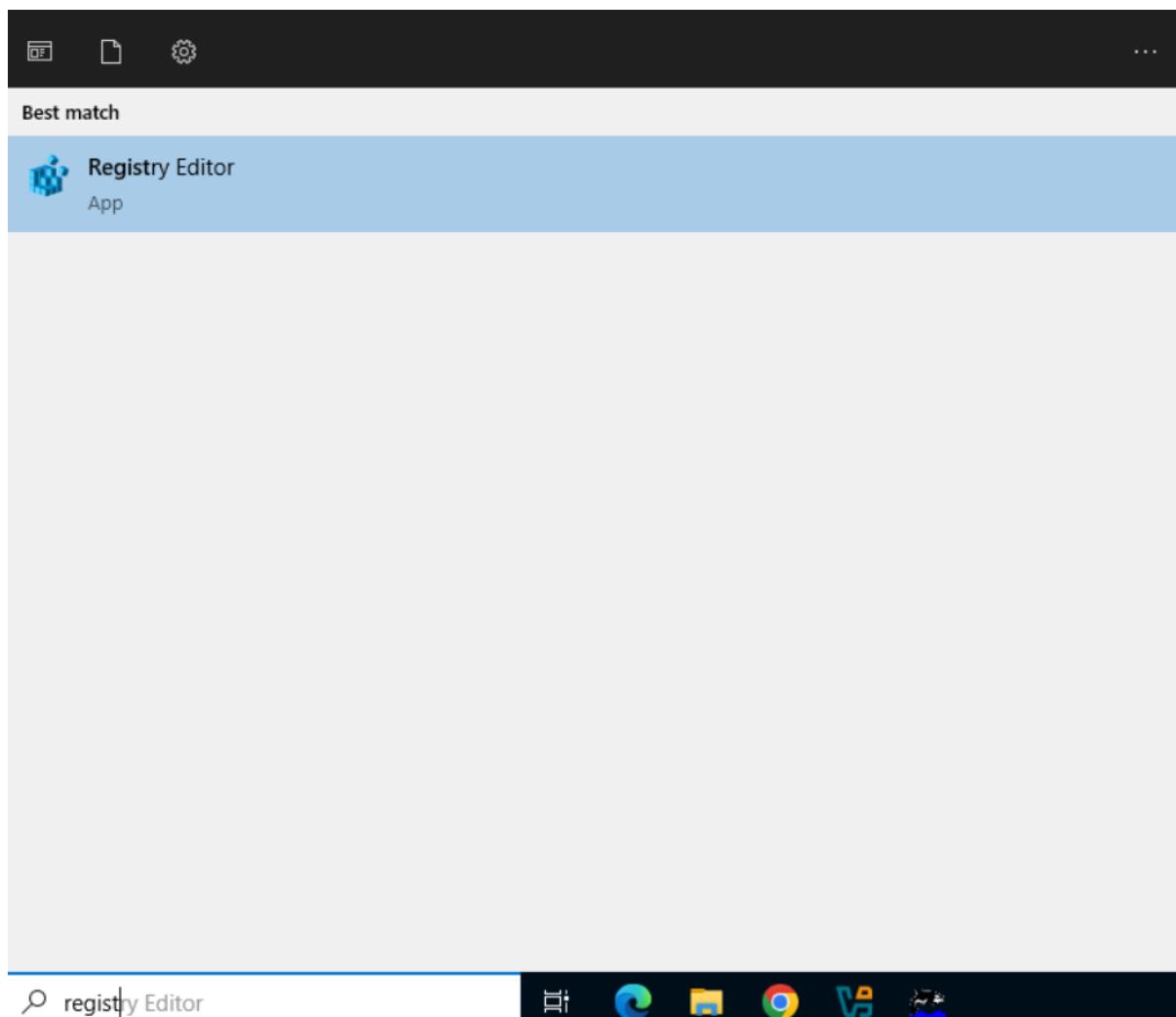
Result:

Verify that the customized registry keys are getting configured while installing the package.

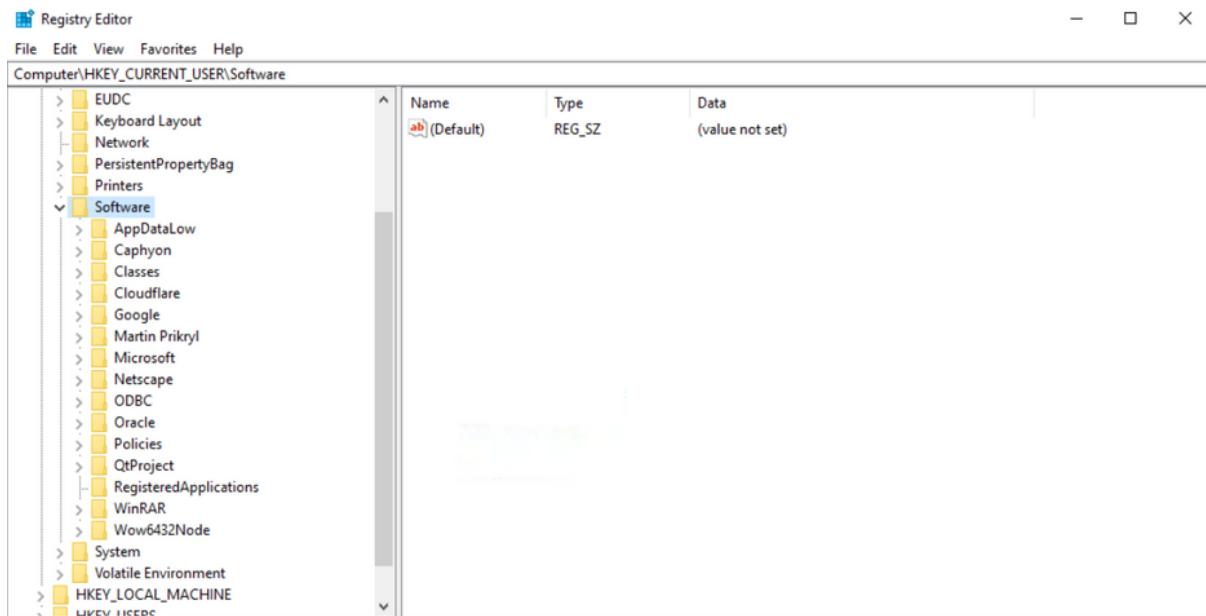
- Check the status of Registry before installation.

Open Registry Editor:

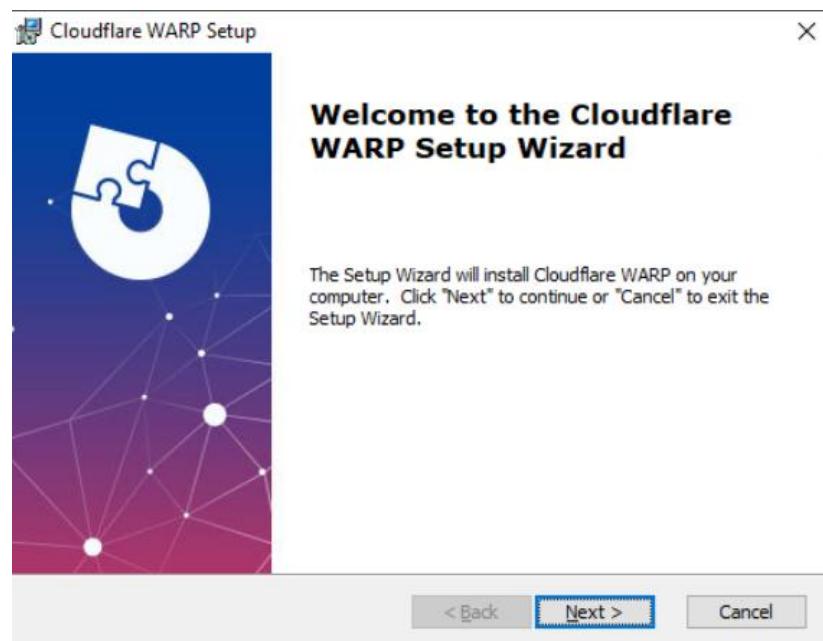
- Simply search for the registry editor in start menu and open it.

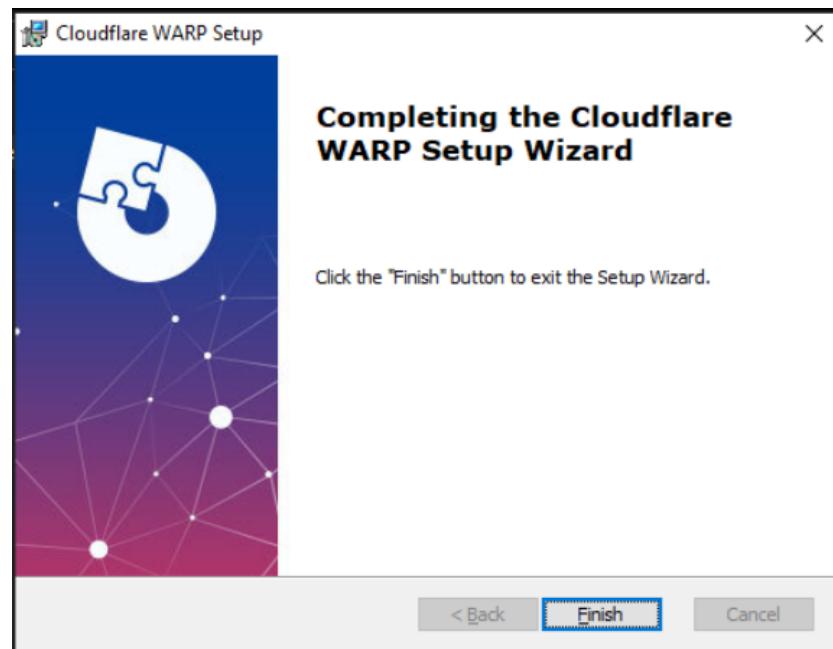


- Look up the desired hierarchy key.



- Launch the Application installer and install it.





- After finishing the installation, verify the status of Registry again.

A screenshot of the Windows Registry Editor. The title bar says "Registry Editor". The menu bar includes File, Edit, View, Favorites, and Help. The toolbar has icons for New Key, New Value, Find, and Delete. The address bar shows "Computer\HKEY_CURRENT_USER\Software\MyRegistry-Project". The left pane shows a tree view of registry keys under Software, including Control Panel, Environment, EUDC, Keyboard Layout, Network, PersistentPropertyBag, Printers, and Software. The Software key is expanded, showing AppDataLow, Caphyon, Classes, Cloudflare, Google, Martin Prikryl, Microsoft, MyRegistry-Project, Netscape, ODBC, Oracle, Policies, QtProject, RegisteredApplications, WinRAR, and Wow6432Node. The right pane displays a table with two rows. The first row is for the "(Default)" value, which is of type REG_SZ and has "Data" "(value not set)". The second row is for "MyValue", which is of type REG_DWORD and has "Data" 0x00000004 (4).

Name	Type	Data
(Default)	REG_SZ	(value not set)
MyValue	REG_DWORD	0x00000004 (4)

- Now we can see that our customized registry key and its value has automatically been created at the correct location.
 - Hence, with this activity we conclude that we **successfully** used **Advanced Installer & Repackager** to Repack an existing application and add customization like **pre-defined registry keys** to the new installer.
-

Precautions:

1. **Use a Clean Virtual Machine:**
 2. **Capture the "Before" State Accurately**
 3. **Perform Minimal Actions During Installation**
 4. **Test the Repackaged Installer Thoroughly.**
-

Tools & Technologies used:

Advanced Installer & Repackager: To perform the repackaging and customization.

Windows Registry Editor: To view the changes made by Installer.

CloudFlare WARP: The application which we customized.

Snipping Tool: To take screenshots of the activity.

Virtual Machine: The machine provided by our team to perform the Activity.

Canva: To create the Report and Presentation.

Sources and References:

The activity steps, theory documentation, and the system minimum requirements are taken from the official documentation of **Advanced Installer**.

Download: [Free Windows Installer - MSI Installer Tool - Advanced Installer](#)

Minimum Requirements: [Advanced Installer - Requirements](#)

Documentation: [Introduction to Advanced Installer](#)

CloudFlare WARP was downloaded from its official website to perform the activity on.

Download: [Download WARP stable releases](#)
