**NetSuppRemove.bat**

This script is designed to uninstall the NetSupport software from a Windows system. It first locates the installation details of NetSupport in the Windows Registry, specifically its version and the uninstall command. The script searches two registry paths where uninstall information is typically stored. If it finds the NetSupport version and uninstall string, it prompts the user to confirm if they want to uninstall NetSupport. If the user agrees, the script executes several steps: it runs a batch file (`killtasks.bat`) to terminate relevant processes, executes a PowerShell script (`NetSuppRemoveRegCheck.ps1`), removes the NetSupport installation folder, and finally runs the uninstall command found in the registry. If the NetSupport version is not found, it notifies the user. The script uses error handling to check for issues during the execution of external scripts and displays appropriate messages.

Steps:

1. The script sets up two registry paths to search for uninstall information.

2. It initializes variables for storing NetSupport's version and uninstall string.

3. It searches the first registry path for NetSupport's installation details. If not found, it searches the second path.

4. If the NetSupport version is found, the script displays the version and asks the user if they want to uninstall NetSupport.

5. If the user confirms, the script:

- Notifies about terminating necessary processes.

- Executes `killtasks.bat` to terminate relevant processes.

- Runs a PowerShell script (`NetSuppRemoveRegCheck.ps1`)

- Attempts to delete the NetSupport installation folder from "C:\Program Files (x86)\NetSupport".

- Initiates the uninstallation process using the uninstall string obtained from the registry.

- Displays a completion message after the uninstallation and cleanup.

6. If the user chooses not to uninstall, the script displays a message aborting the uninstallation.

7. If the script cannot find the NetSupport version, it notifies the user of this.

8. The script ends, and it prompts the user to press a key to close the window.

**killtasks.bat**

This batch script is designed to check for the presence and terminate a series of specific executable processes on a Windows system. These processes include client32.exe, RevitAccelerator.exe, Runplugin64.exe, runplugin.exe, node.exe, msedge.exe (Microsoft Edge browser), smartscreen.exe, and CollaborationKeysController.exe. The script uses a combination of TASKLIST, FINDSTR, and TASKKILL commands. For each process, the script first checks if the process is running using TASKLIST and FINDSTR. If the process is found, it is forcefully terminated with TASKKILL /F. After attempting to terminate a process, the script outputs a message indicating whether the process was successfully terminated or if it was not running.

Step-by-Step Process:

1. The script begins by setting local environment options to enable extensions and delayed expansion.
2. It checks if client32.exe is running using TASKLIST piped into FINDSTR. If the process is found, TASKKILL /F /IM "client32.exe" is executed to forcefully terminate it, and a corresponding success or failure message is displayed.
3. The script repeats this check-and-terminate pattern for each of the following processes: RevitAccelerator.exe, Runplugin64.exe, runplugin.exe, node.exe, msedge.exe, smartscreen.exe, and CollaborationKeysController.exe.
4. For each process, the script uses TASKLIST | FINDSTR /I "<process\_name>.exe" to determine if the process is running. If it is, TASKKILL /F /IM "<process\_name>.exe" is executed to terminate it.
5. After attempting to terminate each process, the script outputs a message indicating whether the process was successfully terminated or if it was not running.
6. The script ends after checking and attempting to terminate all the listed processes.