Manually Trigger a Windows Feature Upgrade

> **Warning:** Proceed only if you understand the risks. Manually forcing a Windows Feature Upgrade can bypass important compatibility checks, potentially leading to system instability or hardware issues. Always back up your data beforehand. This process requires administrative privileges and is recommended only for advanced users in testing or controlled environments. Do not use on critical production systems.

Sometimes, Windows Feature Updates fail due to Group Policies, Windows Hello for Business (WHfB), or unknown errors. In such cases, manually triggering the upgrade can help standardize versions, fix issues, or bypass restrictions. This guide shows how to do that using PowerShell.

Use Cases

- **Remediate Upgrade Rollbacks:** Force reinstallation after automatic rollback due to compatibility checks or silent failures.
- **Bypass WHfB or Group Policy Blocks:** Manually install updates when Windows Hello for Business or update policies prevent standard delivery.
- **Troubleshoot Upgrade Issues:** Isolate and resolve problems by cleanly forcing the upgrade outside the normal update pipeline.
- **Manually Align Device Versions:** Standardize Windows versions across machines for testing, support, or software compatibility.

PowerShell Script

This script downloads the official Windows Upgrade tool to a temp folder and silently starts the upgrade in the background with logging enabled — no user input required.

To run the script on a device where you have access to, follow these steps:

1. **Download PSExec from Microsoft: PsExec - Sysinternals | Microsoft Learn**
2. **Open PowerShell as Administrator and run ".\PsExec.exe -i -s powershell.exe"**

3. **Paste and run the script:**

```
$dir = "$($env:Temp)\FU"
if (Test-Path $dir) {
    Remove-Item "$($dir)\Win11Upgrade.exe" -Force
} else {
    New-Item -Path $dir -ItemType Directory
}

$webClient = New-Object System.Net.WebClient
$url = 'https://go.microsoft.com/fwlink/?linkid=2171764'
$file = "$($dir)\Win11Upgrade.exe"



$webClient.DownloadFile($url, $file)

Start-Process -FilePath $file -ArgumentList "/quietinstall /skipeula /auto upgrade /copylogs
$($dir)"
```

After the script was deployed to the clients, you can get the status of the process using the following command:

```
Get-Process Windows10UpgraderApp
```

Revision #3
Created 2025-06-11 09:21:35 UTC by Luca Noah Caprez
Updated 2025-06-24 07:14:27 UTC by Admin Luca Noah Caprez