



Windows Server Backup & Recovery Toolkit

Administrator & Engineer Reference Manual

For Windows Server 2016 / 2019 / 2022 / 2025

Version: 2026.05.01-1

Liquidware™

© 2026 Liquidware, Inc. All rights reserved. Liquidware, FlexApp and FlexApp™ One are trademarks of Liquidware, Inc. All other marks mentioned herein may be trademarks of their respective companies.

Table of Contents

1. Overview.....	4
1.1 Design Principles.....	4
1.2 Menu Structure.....	4
2. Prerequisites.....	6
2.1 Operating System.....	6
2.2 PowerShell.....	6
2.3 Disk Configuration.....	6
2.4 Windows Server Backup Feature.....	7
2.5 WinRE (Required for Options 6–10 Only).....	7
3. First-Time Setup.....	8
3.1 Launching the Toolkit.....	8
3.2 New Server — Recommended First-Time Order.....	8
3.3 Servers Without a Recovery Partition.....	8
4. Backup Setup Options (1–5).....	10
4.1 Option [1] — Install Windows Server Backup.....	10
4.2 Option [2] — Choose or Create Backup Drive.....	10
Using an Existing Drive.....	10
Creating a New Partition from C:.....	10
4.3 Option [3] — List Available Backups.....	11
4.4 Option [4] — Run Backup.....	11
Pre-Flight Space Check.....	11
Backup Command.....	11
4.5 Option [5] — Delete Backup(s).....	12
5. WinRE Recovery Options (6–9).....	13
5.1 Option [6] — Install Auto-Restore Script into WinRE.....	13
What Option [6] Does.....	13
5.2 Option [7] — Trigger Automated Restore + Reboot.....	13
Pre-Trigger Checks.....	13
Confirmation.....	14
Automated Recovery Sequence.....	14
5.3 Option [8] — Check Status / View Log.....	14
5.4 Option [9] — Cancel Pending Recovery.....	15
6. Recovery Partition (Option 10).....	16
6.1 Prerequisites for Option [10].....	16
6.2 What Option [10] Does.....	16
6.3 Technical Notes.....	17
7. How WinRE Identifies Drives.....	18

7.1 Finding the Windows C: Partition (Restore Target)	18
7.2 Finding the Backup Drive	18
7.3 Recommended Disk Layout	18
8. Post-Restore Checklist	20
9. Troubleshooting	21
10. File & Log Locations	23
11. Limitations & Known Constraints	24
12. Versioning	25
12.1 Version Format	25
12.2 Where the Version Appears	25
12.3 Version History	25
13. Quick Reference	27
New Server Setup	27
Before Packaging a FlexApp™ Application	27
After Packaging a FlexApp™ Application	27
Key Commands	27
Support & Feedback	28

1. Overview

The Windows Server Backup & Recovery Toolkit was created to address a specific challenge faced by Liquidware™ customers running Windows Server for their **FlexApp™ Packaging Console Operating System**. Unlike Windows 10/11, Windows Server does not include System Restore — leaving administrators without a simple built-in mechanism to roll a server back to a known-good state after packaging work.

For customers running on-premises without snapshot capabilities, or in cloud environments where snapshot-equivalent features may not be available or practical, this toolkit fills that gap. It provides a straightforward, script-only method to restore the server to a pristine state without requiring third-party software, hypervisor-level snapshots, or advanced infrastructure.

The toolkit combines backup management with automated WinRE-assisted recovery of all critical volumes in a single unified console. It uses the built-in **wbadmin** engine to create backups and boots into the Windows Recovery Environment (**WinRE**) to execute an unattended restore after the administrator confirms the operation from within Windows.

1.1 Design Principles

- All actions require administrator elevation — no silent background operations.
- Backup and recovery are independent; backup options (1–5) work without a recovery partition.
- Recovery uses embedded WIM hooks and volume GUIDs to remain portable across disk reconfiguration.
- Every destructive operation shows a confirmation prompt before proceeding.
- Post-restore automation (WinRE re-registration, scheduling) is handled automatically.

1.2 Menu Structure

The toolkit presents a numbered console menu. All options are available from a single screen:

Option	Name	Description
[1]	Install Windows Server Backup	Installs the WSB feature via Server Manager if not already present. Safe to re-run; reports if already installed.
[2]	Choose or Create Backup Drive	Selects an existing non-C: local drive or shrinks C: and creates a new dedicated NTFS backup partition.
[3]	List Available Backups	Scans all non-C: local drives and displays all valid wbadmin backup versions with date, target, and recoverability details.
[4]	Run Backup	Executes a full wbadmin backup of C: and all critical volumes. Performs space checks and deletes old backups first.

Option	Name	Description
[5]	Delete Backup(s)	Scans all drives for backups and allows selective or bulk deletion with a final confirmation before any data is removed.
[6]	Install Auto-Restore Script into WinRE	Mounts winre.wim, injects recovery hooks (SR_AutoRecover.cmd, SR_Launcher.cmd, winpeshl.ini), commits changes, and records a SHA256 marker. Run once, and re-run after every restore.
[7]	Trigger Automated Restore + Reboot	Verifies WIM hash, discovers valid backup drives, writes RunRestore.cmd, schedules a one-shot WinRE boot, and reboots. Type RECOVER to confirm — machine restores automatically.
[8]	Check Status / View Log	Shows current WinRE status (reagentc /info), pending recovery config, and the last 15 entries from winre-restore.log.
[9]	Cancel Pending Recovery	Deletes RunRestore.cmd trigger files from all drives so the next reboot is normal. Does not clear the boot-to-RE flag — see Section 5.4.
[10]	Create Recovery Partition	Shrinks C:, creates a new NTFS Recovery partition, copies winre.wim, and registers WinRE. Use on servers that have no existing recovery partition.
[Q]	Quit	Exits the toolkit and automatically removes any temporary T: drive letter mapping assigned during the session.

2. Prerequisites

⚠ WARNING: All prerequisites must be confirmed before use. The toolkit enforces Administrator rights on startup and on every menu action. Running without elevation will produce an error and exit.

2.1 Operating System

Windows Server 2025	Primary development and test platform
Windows Server 2022	Fully compatible
Windows Server 2019	Fully compatible
Windows Server 2016	Compatible — wbadmin supported on all versions

2.2 PowerShell

- PowerShell 5.1 or later (ships inbox on all supported OS versions).
- Session must be launched as Administrator (Run as Administrator).
- Execution policy must allow script execution, or use the bypass flag described in Section 3.1.

2.3 Disk Configuration

C: drive	Windows OS volume. Backed up and restored during automated recovery. Volume GUID is resolved at install time for portability.
Backup drive	A separate locally attached fixed or removable drive (e.g., E:). Cannot be C:. If no separate drive exists, option [2] can shrink C: and create a dedicated backup partition automatically. Recovery partitions are excluded from the drive list. On cloud VMs (Azure, AWS, GCP), both approaches work reliably — either attach a dedicated data disk via the cloud portal, or use option [2] to repartition C:. If repartitioning C:, reboot after all partitioning is complete (options [2] and [10]) before running option [4] for the first time. After that reboot, backups succeed consistently.
Recovery partition	Required for options [6]–[10] only. A partition of GPT type Recovery containing winre.wim. If one does not exist, option [10] will create it by shrinking C:. Options [1]–[5] (backup functionality) work fully without one. On Azure VMs, a small unformatted partition (typically 450 MB) may appear at the front of Disk 0 — this is an Azure-managed partition and must not be used as the WinRE recovery partition. Use option [10] to create a proper 1 GB recovery partition at the end of the OS disk instead.

■ IMPORTANT: The backup drive and C: drive cannot be the same volume. The toolkit explicitly blocks C: from appearing as a backup target. Recovery partitions are also excluded from all backup drive lists. If neither a backup drive nor a recovery partition exists, use option [2] to create a backup drive and option [10] to create a recovery partition before using any other options. On cloud VMs

(Azure, AWS, GCP), both a dedicated data disk and a repartitioned C: work reliably — if repartitioning C:, reboot after all partitioning is complete before running option [4] for the first time.

2.4 Windows Server Backup Feature

The Windows Server Backup feature must be installed before options [3], [4], and [5] will function. Use option [1] to install it. Options [4] and [5] explicitly check for wbadmin.exe at startup and display a clear error if it is missing.

2.5 WinRE (Required for Options 6–10 Only)

- A recovery partition must exist on the primary disk.
- WinRE must be enabled. Verify with: `reagentc /info` — look for "Windows RE status: Enabled".
- The toolkit attempts to auto-enable WinRE when a disabled-but-present partition is found.
- If no recovery partition exists, use option [10] to create one before using options [6]–[9].
- Options [1]–[5] (backup functionality) operate fully without a recovery partition.

3. First-Time Setup

3.1 Launching the Toolkit

Open an elevated PowerShell window (right-click PowerShell → Run as Administrator), navigate to the directory containing the script, and execute it:

```
powershell.exe -ExecutionPolicy Bypass -File .\Windows_Server_Backup_&_Recovery_Toolkit.ps1
```

i NOTE: The -ExecutionPolicy Bypass flag allows the script to run without permanently changing the system execution policy. It applies only to the current process. The filename and path shown above may differ from your deployment — the script may have been renamed to include a version number or placed in a different directory. Substitute the actual filename and path as appropriate.

3.2 New Server — Recommended First-Time Order

Follow this sequence the first time the toolkit is deployed on a server. On single-disk systems where both the backup partition and recovery partition will be created from C:, it is important to run option [2] before option [10] — this ensures the backup partition is placed before the recovery partition on the disk, keeping the partition layout clean and avoiding unallocated space fragmentation issues. On cloud VMs (Azure, AWS, GCP), a reboot is recommended after all partitioning work (options [2] and [10]) is complete and before running option [4] for the first time — disk partitioning operations can leave VSS in a dirty state that causes the first backup attempt to fail, and a clean reboot resolves this.

- ☐ Install and configure the FlexApp™ Packaging Console software on the server
- ☐ Option [1] — Install Windows Server Backup (if not already installed)
- ☐ Option [2] — Choose an existing backup drive, or create one by shrinking C:
- ☐ Option [10] — Create a recovery partition (if one does not already exist)
- ☐ Option [4] — Run the first full backup of the server in its current clean state
- ☐ Option [6] — Install the WinRE auto-restore script into the recovery image
- ☐ Option [8] — Verify WinRE shows Enabled and no errors appear in the log

■ IMPORTANT: The FlexApp™ Packaging Console software must be fully installed and configured before running the first backup. The backup captures the server in its ready-to-package state — this is the pristine baseline the server will be restored to after each application is packaged.

i NOTE: The preferred backup drive selected in option [2] is remembered for the session and displayed in the menu header. It becomes the default in option [4] so you can press Enter to accept it without retyping the drive letter.

3.3 Servers Without a Recovery Partition

If the server has no recovery partition (reagentc /info shows "not available" or "disabled"), complete this additional sequence first. On single-disk systems, option [2] must run before option [10] so the backup

partition is positioned on disk before the recovery partition — running option [10] first would place the recovery partition at the end of the disk, leaving any unallocated space from a subsequent option [2] shrink stranded in the middle of the layout:

- ☐ Place a valid winre.wim at C:\winre\winre.wim (see Section 6.1 for how to obtain it)
- ☐ Option [2] — Configure the backup drive (on single-disk systems, do this before option [10])
- ☐ Option [10] — Create a recovery partition (shrinks C:, creates partition, registers WinRE)
- ☐ Reboot the server (recommended on cloud VMs (Azure, AWS, GCP) — partitioning operations can leave VSS in a dirty state that causes the first backup to fail; a clean reboot resolves this)
- ☐ Option [6] — Install the WinRE auto-restore script into the new recovery image
- ☐ Verify with: `reagentc /info` — should show Windows RE status: Enabled

⚠ WARNING: Option [10] shrinks C: to make room for the recovery partition. Ensure the server has sufficient unallocated space or enough headroom on C: before running it. This operation requires manual diskpart intervention to reverse and should be treated as a permanent change for planning purposes. On cloud VMs (Azure, AWS, GCP), repartitioning C: works reliably — reboot after all partitioning is complete before running option [4] for the first time, and backups will succeed consistently.

4. Backup Setup Options (1–5)

4.1 Option [1] — Install Windows Server Backup

Checks whether the Windows Server Backup feature is installed using Get-WindowsFeature and installs it if absent. If already installed, this is reported and no changes are made. A restart may be required after installation on some OS versions — the toolkit will report if so.

✓ **TIP:** Run this option first on any newly built server. Options [3], [4], and [5] will fail immediately with a clear error if wbadmin.exe is not found.

4.2 Option [2] — Choose or Create Backup Drive

Configures the backup target drive for the current session. Before listing drives, the toolkit displays the used space on C: and the minimum recommended backup drive size (used space + 15% buffer) so you can make an informed choice.

Using an Existing Drive

Lists all local non-C: fixed and removable drives that are not recovery partitions. Each entry shows drive letter, label, free space, and total size. Select the letter of the drive to use. If only one qualifying drive exists, it is selected automatically.

Creating a New Partition from C:

If no separate drive exists, the toolkit can shrink C: and create a new NTFS backup partition from the freed space. On cloud VMs (Azure, AWS, GCP) this works reliably — simply reboot after all partitioning is complete before running option [4] for the first time. The process:

- Displays the current C: size, used space, and allowed shrink range.
- Prompts for the new size of C: in whole GB.
- Shrinks C:, creates a new partition using all remaining unallocated space, formats it NTFS with label "Backup", and sets it as the preferred drive.
- Enter M at the size prompt to cancel without making any changes.

⚠ **WARNING:** Shrinking C: requires manual diskpart intervention to reverse and should be treated as a permanent change for planning purposes. Ensure the new C: size comfortably accommodates the OS, page file, installed applications, and anticipated growth. The toolkit shows the minimum and maximum allowed sizes before proceeding. On cloud VMs (Azure, AWS, GCP), repartitioning C: works reliably as long as you reboot after all partitioning is complete before running option [4] for the first time.

i **NOTE:** Recovery partitions (such as the WinRE partition) are automatically excluded from all drive lists and can never be selected as a backup target.

4.3 Option [3] — List Available Backups

Automatically scans all local non-C: removable and fixed drives and displays every valid wbadmin backup version it finds. Output is grouped by drive exactly as reported by wbadmin. Typical details include:

- Drive letter and backup target path
- Backup time (human-readable date and time)
- Version identifier (used by wbadmin when restoring)
- What can be recovered (Volumes, Files, Bare Metal Recovery, etc.)

No input is required — the scan runs automatically. Use this option to confirm which drives contain valid backups before triggering a restore with option [7].

4.4 Option [4] — Run Backup

Runs a full wbadmin backup of all critical volumes — including C: and the System Reserved / EFI partition — to the selected backup drive.

Pre-Flight Space Check

Before starting, the toolkit calculates:

- Space needed: used space on C: plus 15% buffer.
- Effective available space: current free space on the target drive plus the size of any existing WindowsImageBackup folder (since existing backups are deleted before the new one starts).
- If effective space is still insufficient, the operation hard-blocks with a clear error.

Backup Command

```
wbadmin start backup -backupTarget:<drive> -include:C: -allCritical -quiet
```

⚠ WARNING: Any existing backup on the selected target drive will be deleted before the new backup runs. If you decline deletion, no backup is started. The toolkit is designed to maintain one current backup per target drive.

i NOTE: The -allCritical flag backs up C: and all other volumes required for a complete recovery — including the System Reserved or EFI partition. If this command fails because the disk contains a partition that cannot be shadow-copied (which occurs on cloud VMs including Azure, AWS, and GCP, and some Hyper-V configurations), the toolkit automatically retries with -include:C: only. A warning is shown when the fallback fires. A C:-only backup is sufficient for the automated restore in option [7]. On cloud VMs, if the first backup attempt fails after disk partitioning work (options [2] or [10]), reboot the server and retry option [4] — partitioning operations can leave VSS in a dirty state that a clean reboot resolves.

4.5 Option [5] — Delete Backup(s)

Scans all local non-C: drives for backups and displays a numbered list. Each entry shows the drive, backup time, version identifier, and recovery capabilities.

Selection options at the prompt:

- Enter a number — select and delete a specific backup version.
- A — delete all backups found on all drives.
- M — return to the main menu without any changes.

After selecting, a WARNING callout shows exactly what will be deleted, followed by a final Y/N confirmation. Invalid entries loop with a guidance message. This action cannot be undone.

5. WinRE Recovery Options (6–9)

5.1 Option [6] — Install Auto-Restore Script into WinRE

This option injects the recovery automation files into the winre.wim image. It must be run once per machine after initial setup, and re-run after every successful restore. The script detects automatically if winre.wim has been replaced since option [6] last ran — if it has (for example after a Windows Update or a restore), option [7] will block with a clear error and prompt you to re-run option [6] before proceeding.

What Option [6] Does

- Auto-detects the winre.wim location via `reagentc /info`.
- Creates a timestamped backup of the original WIM to `C:\RecoveryConfig\`.
- Captures the Windows C: volume GUID for portable WinRE-side drive identification.
- Mounts the WIM using DISM and injects four files:
 - `SR_AutoRecover.cmd` — mounts the Windows volume as `W:`, locates `RunRestore.cmd`, and calls it.
 - `SR_Launcher.cmd` — runs `wpeinit`, calls `SR_AutoRecover.cmd`, then reboots.
 - `winpeshl.ini` — hooks WinRE to launch `SR_Launcher.cmd` instead of the normal WinRE UI.
 - `startnet.cmd` — secondary hook for WinRE builds that ignore `winpeshl.ini`.
- Commits and unmounts the WIM.
- Re-registers WinRE via a `reagentc disable/setreimage/enable` cycle.
- Writes a marker file to `C:\RecoveryConfig\winre-scripts-installed.marker` containing the SHA256 hash of the WIM.
- Removes the temporary `T:` drive letter on completion.

■ IMPORTANT: Option [6] must be re-run after every successful restore. For all other cases — such as a Windows Update replacing winre.wim — option [7] will detect the change automatically via SHA256 hash check and block with a clear error prompting you to re-run option [6]. You do not need to proactively re-run it after every update.

5.2 Option [7] — Trigger Automated Restore + Reboot

Triggers an automated, unattended restore of all critical volumes — including C: and the System Reserved / EFI partition — from the most recent backup. This is the primary recovery action, typically run after completing a FlexApp™ application packaging session to return the server to its pristine baseline state.

Pre-Trigger Checks

- Verifies the `winre-scripts-installed.marker` file exists (option [6] has been run).
- Computes the current SHA256 hash of winre.wim and compares it to the stored hash. Blocks if they differ.
- Scans all non-C: local drives for drives that contain both a `WindowsImageBackup` folder and at least one valid `wbadmin` backup version.

- If a preferred backup drive is set and still valid, selects it automatically. Otherwise prompts you to choose from the list of valid backup sources.

Confirmation

After verifying all prerequisites and selecting a backup source, the toolkit displays the backup target, version to be restored, and a final warning. You must type exactly:

```
RECOVER
```

Pressing Enter without typing RECOVER cancels the operation and returns to the menu.

⚠ WARNING: This operation will restore all critical volumes — including C: and the System Reserved / EFI partition — to the state they were in when the backup was taken. All changes made to the server since that backup will be lost. After typing RECOVER, the machine reboots automatically in 10 seconds.

Automated Recovery Sequence

After confirmation, the following sequence runs automatically without further interaction:

1. Option [7] writes RunRestore.cmd to both the selected backup drive and C:\RecoveryConfig\ on the Windows volume.
2. Copies the currently installed winre.wim to <BackupDrive>:\RecoveryConfig\winre.wim.backup for post-restore WinRE rebuild.
3. Calls reagentc /bootore to schedule a one-shot WinRE boot.
4. Reboots the machine into WinRE.
5. WinRE launches SR_Launcher.cmd via winpeshl.ini (with startnet.cmd as a fallback hook).
6. SR_AutoRecover.cmd mounts the Windows partition as W: using its volume GUID (disk/partition fallback if GUID fails).
7. RunRestore.cmd scans drive letters for the WindowsImageBackup folder to identify the backup drive.
8. wbadm restores all critical volumes (C: and System Reserved / EFI) to W: from the latest backup version on the backup drive.
9. On success: WinRE copies winre.wim.backup to the recovery partition, writes RebuildWinRE.cmd to W:\RecoveryConfig\, registers a SYSTEM on-logon scheduled task, and reboots.
10. On failure: WinRE displays an error, waits 30 seconds, and reboots back to Windows. Check C:\RecoveryConfig\winre-restore.log.

5.3 Option [8] — Check Status / View Log

Displays a status summary without making any changes:

- Current WinRE status from reagentc /info.
- Whether a pending RunRestore.cmd trigger file exists.
- Selected backup target and restore details if a recovery is pending.
- Last 15 entries from C:\RecoveryConfig\winre-restore.log.

✓ **TIP:** Run option [8] after any WinRE operation to verify the environment is clean and no stale trigger files remain.

5.4 Option [9] — Cancel Pending Recovery

If option [7] has been triggered but the machine has not yet rebooted, option [9] cancels the pending recovery by deleting any RunRestore.cmd files found on all mounted filesystem drives.

i NOTE: Option [9] removes the RunRestore.cmd trigger files but does NOT clear the one-shot boot-to-RE flag set by reagentc /bootore. If that flag was already set, also run the following to prevent the next restart from entering WinRE:

```
reagentc /disable  
reagentc /enable
```

6. Recovery Partition (Option 10)

Option [10] is a one-time setup option for servers that have no existing WinRE recovery partition. It automates the full partition creation and WinRE registration sequence. If a recovery partition already exists, option [10] will block with an error and direct you to option [6] instead — it will not create a second partition.

6.1 Prerequisites for Option [10]

Before running option [10], you must manually place a valid winre.wim at the following path:

```
C:\winre\winre.wim
```

The script checks for this file first and will exit with an error immediately if it is not found — before making any disk changes.

To obtain winre.wim:

- Copy it from an existing recovery partition on another server of the same OS version.
- Extract it from Windows installation media (mount the ISO, then copy from sources\boot.wim or the recovery image).
- Export it from the current system if WinRE is still partially enabled: use Add-PartitionAccessPath to temporarily letter the recovery partition, then copy the file out.

⚠ WARNING: Do not use a winre.wim from a different Windows Server version or edition. The WIM must match the OS on the target server. A mismatched WIM will result in a non-functional recovery environment.

6.2 What Option [10] Does

11. Verifies C:\winre\winre.wim exists — exits immediately with a clear error if not found.
12. Resolves C:'s disk number dynamically (does not assume disk 0).
13. Shrinks the last shrinkable partition on that disk by 1 GB to free space for the recovery partition. On single-disk systems, ensure the backup partition has already been created via option [2] before running option [10] — otherwise the recovery partition will be placed at the end of the disk before the backup partition exists, leaving the partition layout in a suboptimal state.
14. Creates a new partition in the unallocated space and assigns drive letter R: temporarily.
15. Formats R: as NTFS with label RECOVERY and creates the R:\Recovery\WindowsRE\ folder.
16. Copies C:\winre\winre.wim to R:\Recovery\WindowsRE\winre.wim.
17. Runs reagentc /disable (non-fatal if already disabled), deletes ReAgent.xml, then runs reagentc /setreimage and reagentc /enable to register the new partition.
18. Stamps the new partition with the Windows Recovery GPT type GUID via diskpart.
19. Removes the R: drive letter — the partition is letterless after setup.

i NOTE: After option [10] completes, run option [6] to inject the auto-restore scripts into the new WIM before the toolkit can be used for recovery.

6.3 Technical Notes

- reagentc commands must be run via cmd.exe /c rather than directly from PowerShell to avoid exit code handling issues.
- reagentc /disable runs before deleting ReAgent.xml — this is the required sequence to avoid leaving WinRE in an inconsistent state.
- The disk number for C: is resolved dynamically via Get-Partition -DriveLetter C rather than assuming disk 0.
- Remove-PartitionAccessPath is unreliable for removing drive letters in some configurations. The toolkit uses diskpart (select volume / remove letter / exit) as the robust alternative.

7. How WinRE Identifies Drives

When the machine boots into WinRE for recovery, drive letters are not guaranteed to match their normal Windows assignments. The toolkit uses stored volume identity data and scripted mounting logic to locate the correct partitions reliably.

7.1 Finding the Windows C: Partition (Restore Target)

When option [6] runs, it captures the unique volume GUID of C: and embeds it permanently into SR_AutoRecover.cmd inside the WinRE image. In WinRE, the script uses mountvol with this GUID to mount the correct partition as W: regardless of how many drives are present or how WinRE assigns drive letters.

If mountvol fails (which can happen in rare configurations), the script falls back to disk and partition numbers recorded at install time using diskpart.

i NOTE: Moving the OS disk to a different physical slot or adding/removing disks does not normally prevent the toolkit from finding C:, because the WinRE script first attempts to mount by volume GUID, which is persistent across disk changes.

7.2 Finding the Backup Drive

The backup source is chosen in Windows before the reboot (by option [7]), not by WinRE. RunRestore.cmd scans all drive letters in WinRE looking for a drive that contains a WindowsImageBackup folder, which inherits its letter from the Windows assignment. This scanning approach is more reliable than GUID-based mounting in WinRE, where mountvol with volume GUIDs can return "The parameter is incorrect." Important: only one drive should have a WindowsImageBackup folder at the time of recovery. If multiple drives contain a WindowsImageBackup folder, the scan may select the wrong drive and the restore will fail. Before triggering option [7], ensure only the intended backup drive has a WindowsImageBackup folder — remove or rename any others.

⚠ WARNING: Multiple backup drives are supported for storing backups, but only one drive should have a WindowsImageBackup folder present at the time of recovery. If more than one drive has a WindowsImageBackup folder when option [7] triggers, RunRestore.cmd may select the wrong drive in WinRE and the restore will fail. Before triggering option [7], verify only the intended backup drive has a WindowsImageBackup folder. Use option [5] to delete backups from any unintended drives first.

7.3 Recommended Disk Layout

C:	Windows OS volume only — no backup data.
E:	Dedicated backup drive — single WindowsImageBackup folder, labeled "Backup". Only one drive with a WindowsImageBackup folder should be present at time of recovery.

(Recovery)	Unlabeled, letterless GPT Recovery partition containing winre.wim. Assigned T: temporarily by the toolkit as needed.
-------------------	--

8. Post-Restore Checklist

After WinRE completes a successful restore and reboots back to Windows, complete the following steps in order:

- ☐ Verify WinRE status: run "reagentc /info" — should show Windows RE status: Enabled.
- ☐ If WinRE shows Not Enabled, re-register manually (see Section 9.7).
- ☐ Option [6] — Reinstall the WinRE auto-restore scripts into the freshly restored WIM. (The marker file on C: was wiped when the volumes were overwritten — option [7] will block until this is done.)
- ☐ Option [8] — Confirm no pending recovery trigger files remain and the log shows a clean install.
- ☐ Option [4] — Run a fresh backup of the restored server state once validation is complete.

The following post-restore automation runs automatically in WinRE before the final reboot:

- winre.wim.backup is copied from the selected backup drive back to the recovery partition as R:\Recovery\WindowsRE\winre.wim.
- W:\RecoveryConfig\RebuildWinRE.cmd is written and a SYSTEM scheduled task named RebuildWinRE is registered to run it at next logon.
- RebuildWinRE.cmd attempts to re-register the WinRE image using reagentc and then deletes its own scheduled task.

■ **IMPORTANT:** Always follow the post-restore checklist in the order shown above. Option [7] will block with an error until option [6] is re-run because the marker file is destroyed when the volumes are overwritten during restore.

9. Troubleshooting

Use the table below to diagnose common issues. For any problem not listed, check C:\RecoveryConfig\winre-restore.log and C:\RecoveryConfig\SR_winre_boot.log for detailed output from the most recent operation.

Symptom / Error	Cause & Resolution
wbadmin.exe Not Found	Run option [1] to install the Windows Server Backup feature, then retry the failed option.
No Valid Backup Target Drive (option [4])	A separate local drive other than C: is required. Use option [2] to configure one. Recovery partitions are excluded automatically and cannot be selected.
Not Enough Space on Backup Drive	The toolkit calculates needed space (C: used + 15%) and credits any existing backup that will be deleted. If still insufficient, use option [2] to select or create a larger drive.
WinRE Scripts Not Installed (option [7])	Option [7] blocks with "WinRE recovery scripts have not been installed yet." Run option [6] first. This also occurs after every restore because the C: marker file is wiped when the volumes are overwritten.
WinRE Image Has Changed (option [7])	Option [7] blocks with "The WinRE image has changed since option [6] was last run." This happens after a restore or after Windows Updates replace winre.wim. Re-run option [6].
No Backup Found on Any Drive	Run option [3] to list current backups. Run option [4] to create a fresh backup. Ensure the backup drive is connected and visible in Windows before triggering option [7].
Restore Fails with wbadmin Exit Code -2 (Specified Backup Not Found)	This almost always means RunRestore.cmd found and used the wrong backup drive in WinRE. Check C:\RecoveryConfig\winre-restore.log after reboot — if it shows "Found backup drive at" for more than one drive letter, multiple drives have a WindowsImageBackup folder. Only one drive should have a WindowsImageBackup folder at the time of recovery. Use option [5] to delete backups from any unintended drives, then re-trigger option [7].
WinRE Is Disabled	The toolkit attempts auto-enable when a partition exists. If it fails, run manually: <code>Add-PartitionAccessPath -DiskNumber 0 -PartitionNumber 4 -AccessPath T:\ reagentc /setreimage /path T:\Recovery\WindowsRE reagentc /enable reagentc /info</code>
Recovery Boots Into WinRE UI (no restore)	Option [6] was not run recently, or winre.wim was replaced after injection. Re-run option [6]. Then check: <code>Get-Content C:\RecoveryConfig\SR_winre_boot.log</code>
Server Stays in WinRE After Failed Restore	If wbadmin fails in WinRE, the server reboots automatically after 30 seconds. After reboot, review: <code>Get-Content C:\RecoveryConfig\winre-restore.log</code>
T: Drive Letter Left Mapped	The toolkit removes T: automatically on exit, on completion of options [6] and [7], at the start of options [4] and [6], on error exits from option [6], when the user declines to continue in the WinRE disabled prompt, and at script startup. If T: is still

Symptom / Error	Cause & Resolution
	mapped after an unexpected crash or process kill, remove it manually: <code>\$p = Get-Partition Where-Object { \$_.DriveLetter -eq 'T' } Select-Object -First 1 Remove-PartitionAccessPath -DiskNumber \$p.DiskNumber -PartitionNumber \$p.PartitionNumber -AccessPath T:\</code> Note: a mapped T: drive during a backup operation can cause wbadmin VSS failures. Always remove it before retrying option [4].
DISM Mount Fails / Stale Mount	If DISM mount fails, the toolkit retries once after running <code>dism /Cleanup-Mountpoints</code> . If still failing, run <code>dism /Cleanup-Mountpoints</code> manually from an elevated command prompt and retry option [6].
wbadmin Backup Fails ("file not found")	This is typically a VSS / shadow copy issue caused by a partition that cannot be shadow-copied — common on cloud VMs (Azure, AWS, GCP) and some Hyper-V configurations. The toolkit automatically retries with <code>-include:C:</code> only when <code>-allCritical</code> fails. If the retry also fails, check the VSS writer status with: <code>vssadmin list writers</code> . Also check that no temporary drive letters (such as T:) are mapped to the recovery partition when the backup runs. If the failure occurs immediately after disk partitioning work (options [2] or [10]), reboot the server and retry — partitioning operations can leave VSS in a dirty state that a clean reboot resolves consistently on cloud VMs.

10. File & Log Locations

All runtime files are stored under `C:\RecoveryConfig\` on the Windows volume. Some files are recreated or wiped during a restore — see the "Survives Restore" column below.

Path	Purpose
<code>C:\RecoveryConfig\winre-restore.log</code>	Main activity log. Records install actions, restore trigger details, and WinRE-side output copied back to Windows after recovery. Wiped during restore.
<code>C:\RecoveryConfig\SR_winre_boot.log</code>	Copy of the WinRE boot log written by <code>SR_AutoRecover.cmd</code> before and after calling <code>RunRestore.cmd</code> . Re-created after each WinRE boot.
<code>C:\RecoveryConfig\winre-scripts-installed.marker</code>	Written by option [6]. Contains the SHA256 hash of <code>winre.wim</code> at install time. Option [7] compares against this. Wiped during restore — re-run option [6] after every restore.
<code>C:\RecoveryConfig\recover.json</code>	Written by option [7]. Stores the selected backup target, restore drive, backup version, trigger timestamp, and staged script paths.
<code>C:\RecoveryConfig\RunRestore.cmd</code>	Staged trigger script on the Windows volume. <code>SR_AutoRecover.cmd</code> prefers this copy when WinRE boots.
<code>C:\RecoveryConfig\winre.wim.backup_*</code>	Timestamped backups of <code>winre.wim</code> created before each option [6] reinstall. Safe to delete once option [6] completes successfully.
<code>C:\RecoveryConfig\RebuildWinRE.cmd</code>	Post-restore script written in WinRE. Runs at next logon via SYSTEM scheduled task and re-registers WinRE, then deletes itself and its task.
<code><BackupDrive>:\RecoveryConfig\RunRestore.cmd</code>	Primary recovery trigger script written by option [7] to the selected backup drive. <code>RunRestore.cmd</code> on the backup drive is the first source tried in WinRE.
<code><BackupDrive>:\RecoveryConfig\winre.wim.backup</code>	Copy of the injected <code>winre.wim</code> saved by option [7]. Copied back to the recovery partition by WinRE after a successful restore.
<code><RecoveryPart>:\Recovery\WindowsRE\winre.wim</code>	The WinRE boot image. Option [6] mounts this file, injects recovery scripts, and commits it. Must be re-injected after any replacement.

11. Limitations & Known Constraints

Critical volume restore	The restore flow recovers all critical volumes captured by the backup — including C: and the System Reserved / EFI partition. It is not a general-purpose multi-volume recovery wizard.
Local backup drive only	The backup drive must be a locally attached fixed or removable drive. UNC network share paths are not compatible with the automated WinRE recovery workflow.
Recovery partition required	Options [6]–[9] require a recovery partition. Use option [10] to create one on servers that lack one.
Re-run option [6] after restore	Option [6] must be re-run after every restore and after any change that replaces winre.wim. Option [7] verifies the stored WIM hash and blocks if it has changed.
One backup set per target drive	The wbadmin local-disk workflow deletes existing backups before creating a new one. Plan for one active backup set per target drive.
Multiple backup drives	Supported, but option [7] picks the backup source before reboot. Verify the selected drive carefully on the confirmation screen before typing RECOVER.
wbadmin format only	Only backups created by wbadmin are supported. Third-party backup formats are not compatible.
PowerShell 5.1 minimum	Older PowerShell versions are not supported. All supported OS versions ship with 5.1 inbox.
VSS / shadow copy edge case	On cloud VMs (Azure, AWS, GCP) and some Hyper-V configurations, -allCritical may fail because the disk contains a partition that cannot be shadow-copied. The toolkit automatically retries with -include:C: only. A C:-only backup is sufficient for the option [7] restore workflow. On cloud VMs, a reboot after all partitioning work is complete resolves VSS issues consistently — both dedicated data disks and repartitioned C: drives work reliably after that reboot.

12. Versioning

12.1 Version Format

All releases of the toolkit use a date-based version number in the following format:

YYYY.MM.DD-rev

YYYY	Four-digit year (e.g., 2026)
MM	Two-digit month (e.g., 03 for March)
DD	Two-digit day (e.g., 27)
-rev	Sequential revision number for same-day releases, starting at 1

Examples:

- 2026.03.27-1 — first release on March 27, 2026
- 2026.03.27-2 — a same-day hotfix on March 27, 2026
- 2026.06.15-1 — next update released June 15, 2026

✓ **TIP:** Date-based versioning means customers can determine which version is newest simply by reading the version string — no changelog lookup required. A higher date is always a newer release.

12.2 Where the Version Appears

Script header	Defined as \$Script:Version near the top of Windows_Server_Backup_&_Recovery_Toolkit.ps1 and displayed in the menu header on launch.
Filename	When distributing the script, append the version to the filename: Windows_Server_Backup_&_Recovery_Toolkit_2026.03.27-1.ps1
This document	The document revision date in the cover page and version history table (Section 12.3) aligns with the script version date.

12.3 Version History

The table below tracks all released versions of the toolkit. Always reference the version number when reporting issues or requesting support.

Version	Date	Author	Summary of Changes
2026.03.27-1	Mar 27, 2026	Solutions Architecture	Initial release. Full backup and WinRE recovery toolkit with options 1-10. Includes option [10] recovery partition creation, SHA256 WIM hash verification, and drive-letter scan approach for WinRE restore.

Version	Date	Author	Summary of Changes
2026.05.01-1	May 1, 2026	Solutions Architecture	Cloud VM compatibility (Azure, AWS, GCP) and bug fixes: Added -allCritical retry fallback to -include:C: only for non-snapshotable partitions. Fixed T: drive letter cleanup — Remove-RecoveryDriveLetter rewritten to use Get-Partition to find T: directly; added cleanup at script startup, Assert-WinREEnabled return paths, and Install-WinRERecoveryScript early exit paths. Fixed WIM hash capture timing for marker file. Fixed bare return bug in Assert-WinREEnabled. Added Remove-RecoveryDriveLetter at start of options [4] and [6]. Added graceful error handling in Get-WinREWimPath. Added staged winre.wim detection message in Assert-WinREEnabled. Added version constant displayed in menu header. Fixed missing Pause-Toolkit in option [7] error paths. Added existing recovery partition check to option [10] with 500 MB size filter to exclude EFI partition. Confirmed cloud VMs work reliably with repartitioned C: after a single reboot post-partitioning — guidance updated throughout to reflect this. Updated first-time setup order: option [2] before option [10] before option [4] before option [6]. Added section 3.1 filename/path note. Updated all Azure-specific references to cover cloud VMs (Azure, AWS, GCP) broadly.

i NOTE: When distributing an updated version of the script, add a new row to this table before releasing. Include a brief but specific summary — engineers in the field need to know at a glance what changed and whether it affects their deployment.

13. Quick Reference

New Server Setup

- ☐ Option [1] — Install Windows Server Backup
- ☐ Option [2] — Configure backup drive
- ☐ Option [10] — Create recovery partition (if one does not already exist)
- ☐ Option [4] — Run first full backup of server in clean state
- ☐ Option [6] — Install WinRE auto-restore script
- ☐ Option [8] — Verify clean status

Before Packaging a FlexApp™ Application

- ☐ Option [4] — Create a fresh backup of the current pristine state
- ☐ Option [6] — Refresh WinRE injection (if winre.wim was updated)

After Packaging a FlexApp™ Application

- ☐ Option [7] — Review the backup source shown, type RECOVER to confirm
- ☐ Wait — server restores automatically and reboots to Windows
- ☐ Verify: reagentc /info shows Windows RE status: Enabled
- ☐ Option [6] — Reinstall WinRE auto-restore scripts
- ☐ Option [8] — Confirm no pending triggers remain
- ☐ Server is now back to pristine state and ready for the next packaging session

Key Commands

```
# Check WinRE status
reagentc /info
```

```
# View logs after a restore
Get-Content C:\RecoveryConfig\winre-restore.log
Get-Content C:\RecoveryConfig\SR_winre_boot.log
```

```
# Clean up stale DISM mounts
dism /Cleanup-Mountpoints
```

```
# Re-enable WinRE manually
Add-PartitionAccessPath -DiskNumber 0 -PartitionNumber 4 -AccessPath T:\
reagentc /setreimage /path T:\Recovery\WindowsRE
reagentc /enable
reagentc /info
```

```
# Remove T: manually if left mapped
$P = Get-Partition | Where-Object { $_.DriveLetter -eq 'T' } | Select-Object -First 1
Remove-PartitionAccessPath -DiskNumber $P.DiskNumber -PartitionNumber $P.PartitionNumber -
AccessPath T:\
```

Support & Feedback

You must contact SparkTools@liquidware.com for support for this toolkit.