Azure Image Lifecycle & Versioning – Immutable CIO Images

# Executive Summary

Azure Compute Gallery (ACG) does not provide an AWS‑style, auto‑enforcing lifecycle policy for customer images. To achieve predictable, safe deprecation, we stamp each image version with End‑of‑Life (EOL), set Exclude From Latest for staged releases, and enable Block Deletion Before End‑of‑Life for protection. Enforcement and cleanup are automated with Azure DevOps (ADO) pipelines and an Azure Automation runbook.

# Image Lifecycle in Azure (CIO Pattern)

Key stages:

1) Build with AIB + ADO → publish to ACG with replication to target regions.

2) Tag with EOL, Exclude From Latest, and Block Deletion Before EOL.

3) Promote through rings (Dev → DIT → SIT → BAT/PPD → Prod) using ADO Environment gates.

4) Consumers update IaC to consume the new explicit version; VMSS rolls forward via rolling upgrades.

5) After grace period, Automation runbook deletes versions past EOL.

## No Built‑In Lifecycle Policy (What We Use Instead)

EOL (End of Life): Signals deprecation and drives policy/automation to block new deployments after the date.

Exclude From Latest: Keeps a version out of “latest” selectors during testing or staged rollout.

Block Deletion Before EOL: Prevents accidental deletion of a version before its EOL date.

## Automation in ADO Pipeline

During copy/publish from Sandbox to target subscriptions/regions, the pipeline sets:

az sig image-version create \  
 -g <rg> -r <gallery> -i <imageDef> -e <version> \  
 --image-version "/subscriptions/<srcSub>/resourceGroups/<srcRg>/providers/Microsoft.Compute/galleries/<gallery>/images/<imageDef>/versions/<version>" \  
 --target-regions "canadacentral=2 canadaeast=1" \  
 --target-region-encryption "<your\_DES\_mapping>" \  
 --end-of-life-date "2026-09-30T00:00:00Z" \  
 --exclude-from-latest true \  
 --block-deletion-before-end-of-life true

If you already created the version earlier in the job, update it right after:

az sig image-version update \  
 -g <rg> -r <gallery> -i <imageDef> -e <version> \  
 --set publishingProfile.endOfLifeDate="2026-09-30T00:00:00Z" \  
 publishingProfile.excludeFromLatest=true \  
 --block-deletion-before-end-of-life true

## Deletion via Azure Automation Account (Past EOL)

A Runbook lets authorized users delete an image version by submitting its ACG Image Version Resource ID. The runbook checks that the EOL date is in the past, adjusts the Block Deletion flag if required, and deletes the version.

# PowerShell Runbook (Az.Accounts, Az.Compute modules)  
param(  
 [Parameter(Mandatory=$true)]  
 [string] $ImageVersionId  
)  
  
# Parse resource path  
$parts = $ImageVersionId -split '/'  
$sub = $parts[2]; $rg = $parts[4]; $gal = $parts[8]; $img = $parts[10]; $ver = $parts[12]  
Select-AzSubscription -SubscriptionId $sub  
  
$iv = Get-AzGalleryImageVersion -ResourceGroupName $rg -GalleryName $gal -GalleryImageDefinitionName $img -Name $ver  
$eol = $iv.PublishingProfile.EndOfLifeDate  
$block = $iv.SafetyProfile.BlockDeletionBeforeEndOfLife  
  
if (-not $eol) { throw "EOL not set. Refusing to delete." }  
if ($eol -gt (Get-Date)) { throw "EOL ($eol) is in the future. Refusing to delete." }  
  
if ($block) {  
 Update-AzGalleryImageVersion -ResourceGroupName $rg -GalleryName $gal -GalleryImageDefinitionName $img -Name $ver -BlockDeletionBeforeEndOfLife:$false | Out-Null  
}  
Remove-AzGalleryImageVersion -ResourceGroupName $rg -GalleryName $gal -GalleryImageDefinitionName $img -Name $ver -Force

# Versioning Strategy (CGIP Base & CIO Overlay)

We use semantic versioning and explicitly record the Base→CIO dependency:

• Base CGIP Image = MAJOR.MINOR.PATCH

• CIO Image = MAJOR.MINOR.PATCH, with metadata baseVersion=X.Y.Z

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Scenario | Base MAJOR | Base MINOR | Base PATCH | CIO MAJOR | CIO MINOR | CIO PATCH |
| OS family/edition change; hardening baseline overhaul | ✅ |  |  | ✅\* |  |  |
| Monthly OS/tool refresh (non‑breaking) |  | ✅ |  |  | ✅† |  |
| Base zero‑day security hotfix |  |  | ✅ |  |  | ✅‡ |
| Add/update platform tool in Base (non‑breaking) |  | ✅ |  |  | ✅† |  |
| New non‑breaking CIO feature/tool |  |  |  |  | ✅ |  |
| CIO security/config hotfix (non‑breaking) |  |  |  |  |  | ✅ |

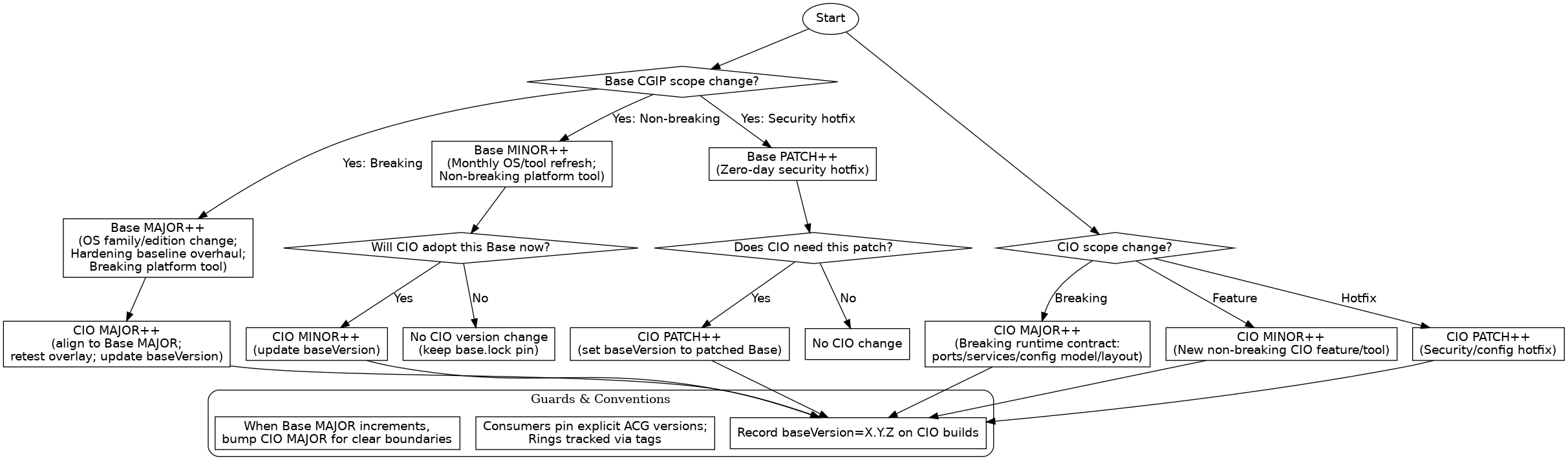
\* Keep CIO MAJOR aligned to Base MAJOR for clear approval boundaries.

† CIO MINOR when adopting the new Base MINOR in the same cycle.

‡ If CIO needs the Base PATCH, rebuild CIO as PATCH and set baseVersion to the patched Base.

# Flowchart – Version Bump Decision Matrix

Use this flow to determine which version number to increment on Base vs CIO images.



# Consumer Impact When a New Image Version is Published

• Existing VMs/VMSS using a specific ACG version are unaffected until IaC (or model) is updated.

• Avoid using “latest” in production; use explicit versions for reproducibility.

• If a version is marked Exclude From Latest, consumers referencing “latest” will not pick it up.

• VMSS: once the model references a new image version, use Rolling Upgrades with health probes and batch sizing (e.g., MaxSurge 20–33%) to cut over safely.

• Rollback: re‑deploy or update the model to the prior ACG version; retain at least 6–12 versions.

# Appendix – Copy & Stamp in az CLI

SRC\_SUB=<sandboxSubId>  
SRC\_RG=rg-sandbox  
GALLERY=acg-cio  
IMAGE\_DEF=CIO-Win-Std-Gen2  
VERSION=4.3.1  
TARGET\_SUB=<targetSubId>  
TARGET\_RG=rg-platform  
TARGET\_REGIONS="canadacentral=2 canadaeast=1"  
EOL\_DATE="2026-09-30T00:00:00Z"  
DES\_MAP="<your\_DES\_mapping>"  
  
az account set -s "$TARGET\_SUB"  
az sig image-version create -g "$TARGET\_RG" -r "$GALLERY" -i "$IMAGE\_DEF" -e "$VERSION" --image-version "/subscriptions/$SRC\_SUB/resourceGroups/$SRC\_RG/providers/Microsoft.Compute/galleries/$GALLERY/images/$IMAGE\_DEF/versions/$VERSION" --target-regions $TARGET\_REGIONS --target-region-encryption "$DES\_MAP" --end-of-life-date "$EOL\_DATE" --exclude-from-latest true --block-deletion-before-end-of-life true

# Notifications & Communications (CGIP → CIO)

The CGIP team will send an email notification to CIO stakeholders whenever a new MAJOR, MINOR, or PATCH version of the CGIP Base Image is published. This communication keeps CIO teams aligned on baseline changes, security posture, and required actions.

When notifications go out:

• At initial publish to ACG (Dev ring) for awareness and early testing.

• At Prod promotion, confirming availability for consumer cutover.

Email content (recommended template):

Subject: [CGIP Base Image] v{BaseVersion} ({ChangeType}) published – EOL {EOLDate}  
  
Hello CIO Team,  
  
A new {ChangeType} version of the CGIP Base Image has been published.  
  
Version: {BaseVersion}  
Change Type: {ChangeType} (Major/Minor/Patch)  
EOL: {EOLDate}  
Regions: {Regions}  
Exclude From Latest: {ExcludeFromLatest}  
Block Deletion Before EOL: {BlockDeletion}  
Replication Status: {ReplicationStatus}  
  
What changed:  
- {Highlights}  
  
Actions for CIO:  
- Review release notes and validate CIO overlay builds as needed.  
- If adopting this base, update base.lock and trigger CIO image build.  
  
References:  
- ACG Resource ID: {ACGVersionId}  
- Release Notes: {ReleaseNotesUrl}  
  
Thanks,  
CGIP Platform Team

Implementation notes:

• Automate notifications from the ADO pipeline stage that creates/updates the ACG image version.

• Populate the template fields from pipeline variables/tags (version, regions, EOL, excludeFromLatest, replication status).

• Send via your preferred channel: corporate SMTP/Graph (Logic App or Function), SendGrid, or a Teams/Slack webhook in addition to email.

# CIO Adoption Policy & Deadlines (Base → CIO Uptake)

To keep the fleet secure and consistent, CIO teams must adopt new CGIP Base Image versions on the timelines below. Deadlines are measured from the CGIP publish date (Dev ring). Older versions are marked with EOL and eventually blocked via policy.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Version Type | Initial Notify | Dev Deadline | Non‑Prod Deadline (DIT/SIT) | Prod Deadline | Freeze Old Version (deny new deploys) | EOL Old Version |
| MAJOR | Day 0 (Dev publish) | ≤ Day 14 | ≤ Day 30 | ≤ Day 60 | Day 90 | Day 120 |
| MINOR (monthly) | Day 0 | ≤ Day 7 | ≤ Day 21 | ≤ Day 30 | Day 45 | Day 60 |
| PATCH – Critical | Immediate | Same day | ≤ 24–48 h | ≤ 72 h | 96 h | 7 days |
| PATCH – High | Day 0 | ≤ 2 days | ≤ 5 days | ≤ 7 days | 10 days | 14 days |
| PATCH – Standard | Day 0 | ≤ 7 days | ≤ 14 days | ≤ 30 days | 45 days | 60 days |

Notes: “Freeze” uses Azure Policy/ADO gates to deny deployments using the old ACG version. EOL marks the version as deprecated and eligible for automated deletion by the Automation runbook.

## How CIO Teams Update to a New Base Image

• Read the CGIP notification and release notes; identify breaking vs non-breaking changes.

• Create a PR in the CIO overlay repo updating the pinned base (e.g., base.lock → baseVersion=X.Y.Z).

• Run the CIO ADO pipeline to build the new CIO image on the updated Base; produce SBOM + test artifacts.

• Dev Ring: deploy test VM/VMSS; run smoke and compliance tests; fix overlay issues if any.

• Promote through DIT/SIT; run workload validation tests; capture sign-offs from app owners.

• Update IaC to reference the new CIO ACG image version explicitly (avoid “latest”).

• Prod cutover: execute Rolling Upgrade with health probes and MaxSurge (e.g., 20–33%); monitor telemetry.

• Post-cutover: confirm health, update dashboards, and archive prior version according to retention.

## Exception & Deferral Process

If a deadline cannot be met, file a deferral RFC before the freeze date, including risk assessment, compensating controls, target remediation date, and approvers (Security, Platform, CIO Engineering). Deferrals typically expire within 30–60 days.

## Tracking & Enforcement

• Dashboard: subscription/resource-group rollup showing image version per VM/VMSS and drift from policy.

• Azure Policy: deny creates and VMSS scale-outs using ACG versions past Freeze/EOL windows.

• ADO Gates: block production releases if the target service is behind required Base/CIO versions.

• Automation: daily/weekly sweep to remove versions past EOL after grace (and flip block deletion flag when EOL is past).

## Email Templates to CIO Teams

Subject: ACTION REQUIRED – Adopt CGIP Base {BaseVersion} ({ChangeType}) by {ProdDeadline}  
  
Hello CIO Team,  
  
CGIP Base Image {BaseVersion} ({ChangeType}) has been published.  
  
Deadlines from publish:  
- Dev by {DevDeadline}  
- Non‑Prod (DIT/SIT) by {NonProdDeadline}  
- Prod by {ProdDeadline}  
Freeze old version on {FreezeDate}; EOL on {EOLDate}.  
  
What changed:  
- {Highlights}  
  
Required actions:  
1) Update base.lock to {BaseVersion} and rebuild the CIO image.  
2) Validate in Dev → DIT/SIT, then cut over Prod before {ProdDeadline}.  
3) Confirm completion via the adoption dashboard and reply to this email.  
  
References:  
- Release Notes: {ReleaseNotesUrl}  
- ACG Version ID: {ACGVersionId}  
  
Thanks,  
CGIP Platform Team

Subject: REMINDER – CGIP Base {BaseVersion} adoption due by {ProdDeadline}  
  
Hello CIO Team,  
  
This is a reminder to adopt CGIP Base {BaseVersion} ({ChangeType}). The old version freezes on {FreezeDate} and reaches EOL on {EOLDate}.   
If you need a deferral, submit an RFC before the freeze date with risk acceptance and a remediation plan.  
  
Status: {AdoptionStatus}  
  
Thanks,  
CGIP Platform Team