# NGINX Plus Ingress Controller Upgrade Plan

**From**: 3.4.3  
**To**: 3.5.2  
**Helm chart**: 1.2.2 (per release notes)  
**Cluster(s)**: AKS 1.30.0 (Linkerd-injected)  
**Image registry change**: ACR01 → **ACR02** (acrappcommoneus202)

## 1) Scope & Objectives

* Upgrade only the **nginx-ingress** deployment from 3.4.3 to **3.5.2**.
* Redirect image pulls for nginx-ingress **to ACR02**.
* Keep all other workloads on ACR01 until separately migrated.
* Preserve all VirtualServer/VirtualServerRoute (VS/VSR) and Ingress functionality.
* Zero/minimal downtime via rolling deployment.

## 2) Current State Summary

* Controller version: **3.4.3**
* Values indicate **Plus** build (controller.nginxplus: true), Linkerd inject enabled.
* Helm is installed from a local chart folder; image repository & tag are overridden in Terraform.
* ACR02 (acrappcommoneus202) is provisioned and will be the new source of images.

## 3) Target State Summary

* Controller image: acrappcommoneus202.azurecr.io/nginx-plus-ingress:3.5.2
* Helm chart: **1.2.2** (controller 3.5.2)
* CRDs: **updated to v3.5.2** prior to Helm upgrade
* Remote state output consumed by Terraform: container\_registry\_02.fqdn

## 4) Pre‑Upgrade Checklist

**Platform & compatibility** - [ ] AKS 1.30.0 is supported by NIC 3.5.2. - [ ] Linkerd sidecar compatible with probe settings and PodSecurity contexts.

**Registry & permissions** - [ ] Image nginx-plus-ingress:3.5.2 is pushed to **ACR02**. - [ ] Kubelet identity (or UAI used for pulls) has **AcrPull** on **ACR02**.

**CRDs & webhooks** - [ ] Apply NIC **v3.5.2 CRDs** (Helm does **not** upgrade CRDs).  
kubectl apply -f https://raw.githubusercontent.com/nginx/kubernetes-ingress/v3.5.2/deploy/crds.yaml - [ ] Confirm apiextensions.k8s.io objects updated for: virtualservers, virtualserverroutes, transportservers, policies. - [ ] Admission webhook certs remain valid (or are managed by the chart).

**Backups & drift checks** - [ ] Export all custom NIC resources: VS/VSR/Policies/GlobalConfiguration/TransportServers. - [ ] Snapshot current Helm release values and rendered manifests for diff/rollback.

## 5) Risk Register — “What could break?”

| Area | Potential Breakdown | Symptom | Prevention / Fix |
| --- | --- | --- | --- |
| **Image pull** | ACR02 not reachable / no AcrPull | ImagePullBackOff | Ensure MI AcrPull on ACR02; verify image exists/tag correct. |
| **CRD mismatch** | CRDs not updated to 3.5.2 | Validation errors, controller log errors | Apply v3.5.2 CRDs **before** Helm upgrade. |
| **Webhook** | Patch job/cert issues | Helm upgrade hangs, admission failures | Ensure webhook patch job image also available (if used) or disable if not required. |
| **Linkerd** | Probe/sidecar interaction | Readiness drops, slow rollout | Keep linkerd.io/inject: enabled; confirm health probes; allow init delays. |
| **Config/annotations** | Deprecated/renamed fields (rare in 3.5.x) | Rejections, ignored config | Review release notes; run kubectl explain on VS/VSR; adjust if needed. |
| **WAF bundles** | NAP WAF upgraded to v4.10 in 3.5.2 | Incompatible WAF bundles (≤ v4.8.x) | Recompile bundles for v4.9+; JSON-based policies unaffected. |
| **Chart drift** | Local chart path vs value overrides | No change after apply | Confirm helm\_release chart path exists for desired tag or set unversioned folder plus image tag override. |
| **Traffic policy** | Listener or TLS changes | 4xx/5xx spikes | Compare generated NGINX; pre-prod canary test; validate TLS material. |

## 6) Deprecations & Annotation Notes (3.4.3 → 3.5.2)

* No blocking **annotation deprecations** between 3.4.x and 3.5.2.
* Recommended **hygiene**: audit for legacy nginx.org/\* annotations, ensure they are supported in Plus.
* VirtualServer/VirtualServerRoute schemas stable; ensure **v3.5.2 CRDs** are applied so validation stays correct.

## 7) VirtualServer & VirtualServerRoute Handling

* **No spec changes required** for 3.4.3 → 3.5.2.
* After CRD update, run:
  + kubectl get vs -A and kubectl get vsr -A to ensure all objects are Valid.
  + kubectl describe vs <name> -n <ns> to catch any warnings after reconciliation.
* If you use **Policies** (e.g., OIDC/JWT/WAF/RateLimit), validate they still bind to the VS/VSR as expected.

## 8) Change Implementation Steps (Terraform + Helm)

1. **CRDs first** (cluster-admin run once):  
   kubectl apply -f https://raw.githubusercontent.com/nginx/kubernetes-ingress/v3.5.2/deploy/crds.yaml
2. **Ensure image in ACR02**: push nginx-plus-ingress:3.5.2 to acrappcommoneus202.
3. **Permissions**: confirm kubelet/UAI has AcrPull on ACR02.
4. **Terraform edits**:
   * Use data.terraform\_remote\_state.common.outputs.container\_registry\_02.fqdn for the image repository host.
   * Set controller.image.repository → ${FQDN}/nginx-plus-ingress.
   * Set controller.image.tag → 3.5.2 (or via env override map).
5. **Plan & apply**:
   * terraform plan
   * terraform apply
6. **Rollout watch**:
   * kubectl -n nginx-ingress rollout status deploy/nginx-ingress
   * kubectl -n nginx-ingress get pods -l app=nginx-ingress-nginx-ingress

## 9) Validation Plan (pre/post)

**Before** - Capture current image, ready replicas, QPS/latency baseline. - Export current VS/VSR and Policies.

**After** - Verify deployment image string equals acrappcommoneus202.azurecr.io/nginx-plus-ingress:3.5.2. - All VS/VSR status show Valid (or previous health state). - Ingress/VS endpoints return 200 OK; smoke tests on representative apps. - No ImagePullBackOff, CrashLoopBackOff, or admission errors in controller logs.

## 10) Rollback Plan

* **Fast rollback**: re-apply previous image tag 3.4.3 via Terraform; or helm rollback to prior revision.
* CRDs: backward compatible across 3.4.x → 3.5.2; no CRD rollback expected. If needed, re-apply the 3.4.3 CRDs.
* Keep 3.4.3 image available in **ACR02** (or ACR01) for immediate fallback.

## 11) Communication & Change Windows

* Change window aligned with lowest external traffic.
* Announce a **read-only/at-risk** interval for apps fronted by nginx-ingress.
* Stakeholders: Platform (AKS), Networking/SRE, Security (if using WAF), App owners.

## 12) Appendix — Commands

# Show controller image used  
kubectl -n nginx-ingress get deploy nginx-ingress -o jsonpath='{.spec.template.spec.containers[0].image}{"\n"}'  
  
# Watch rollout  
kubectl -n nginx-ingress rollout status deploy/nginx-ingress  
  
# List VS/VSR statuses  
kubectl get vs -A; kubectl get vsr -A  
  
# Tail controller logs  
kubectl -n nginx-ingress logs deploy/nginx-ingress --tail=100

## 13) Open Items / Decisions

* Confirm Helm chart folder selection is compatible with 1.2.2.
* Confirm whether **App Protect WAF** is in use; if so, recompile bundles (NAP v4.10).
* Confirm ACR02 remote‑state output is available as container\_registry\_02 and applied in the **common** stack.