**Runbook: Rotate Telerik / KendoReact License in CI**

**Project:** MCPX-KendoBridge Admin Portal  
**Runbook ID:** TJ-MCPX-RB-04  
**Version:** 1.0.0  
**Status:** Draft  
**Date:** 2025-09-27  
**Owner:** DevSecOps / SRE (Technijian)  
**Confidentiality:** Technijian Internal

**Document Control**

| **Version** | **Date** | **Author** | **Change Summary** | **Status** |
| --- | --- | --- | --- | --- |
| 1.0.0 | 2025-09-27 | SRE Lead | Initial rotation procedure + CI snippets | Draft |

**Approvals**

| **Role/Team** | **Name** | **Signature/Date** | **Comment** |
| --- | --- | --- | --- |
| Director of Software Eng. |  |  |  |
| Systems Architect |  |  |  |
| Security & Compliance |  |  |  |
| QA Lead |  |  |  |

**1) Purpose**

Ensure the **KendoReact/Telerik** license is activated during CI builds without ever committing a license string/file to the repo. This runbook covers **secret rotation**, **CI activation**, **validation**, **rollback**, and **evidence** capture.

**2) Scope & Pre-Reqs**

* Applies to **web** builds that import KendoReact Fluent 2 theme/components.
* CI: GitHub Actions environments **Alpha / Beta / RTM / Prod**.
* You have the **new license key** (from vendor portal) and a designated **secret name**:
  + Recommended secret key: **KENDO\_UI\_LICENSE** (per-environment GitHub secret).
* Builds use the Kendo license activator in CI (no license text in code).

**Never** commit license text to source control. Treat the key as a secret at rest and in logs.

**3) Roles, Timing, Dependencies**

* **Responsible:** SRE/DevSecOps
* **Accountable:** Director of Software Engineering
* **Consulted:** Systems Architect, QA, Security
* **When:** Quarterly or on vendor key change; during a **low-traffic window** for Alpha first
* **Dependencies:** CI runner access to secrets; node and npm present; web project present under /web

**4) High-Level Flow**

1. **Prepare** new key (vendor portal).
2. **Rotate secret** in GitHub Environments (Alpha → Beta → RTM → Prod).
3. **Activate in CI** using the license CLI (before npm run build).
4. **Validate** build output + logs; ensure no license appears in artifacts.
5. **Promote** through envs; attach evidence to the release.

**5) Step-by-Step Procedure**

**5.1 Preparation**

* Obtain the **new license key** safely (out-of-band).
* Open a short **change ticket** with: reason, scope, affected envs, planned window, and rollback plan.

**5.2 Update GitHub Environment Secrets**

For each environment (**Alpha**, then **Beta**, **RTM**, **Prod**):

1. Navigate to **Repo → Settings → Environments → <Env> → Secrets**.
2. **Create/Update** secret **KENDO\_UI\_LICENSE** with the new license **(single line, no quotes)**.
3. Save; ensure **Secret Scanning** and **Dependabot** are enabled repo-wide.

If you previously used a different secret name, standardize on KENDO\_UI\_LICENSE and update CI step (below).

**5.3 CI Activation Step (GitHub Actions)**

Insert this step **before** the web build in .github/workflows/ci.yml (and in deploy.yml if it builds the web):

# activate Kendo license (no output of key)

- name: Activate Kendo UI license

if: hashFiles('web/\*\*/package.json') != ''

working-directory: web

env:

KENDO\_UI\_LICENSE: ${{ secrets.KENDO\_UI\_LICENSE }}

run: |

npx --yes kendo-ui-license activate

# Alternative if your project uses the scoped CLI:

# npx --yes @progress/kendo-licensing activate

Then build as usual:

- name: Build web

if: hashFiles('web/\*\*/package.json') != ''

working-directory: web

run: npm ci && npm run build

**Notes**

* The activator generates/validates a license artifact in the build context or runner profile; it **must run each clean build**.
* Masking: GitHub automatically masks ${{ secrets.\* }}; do **not** echo the key.
* Keep the step **silent**—no printing of the secret or license blob.

**5.4 Validate (Alpha)**

* Run CI on a feature branch; verify:
  + Web build passes (no “license not activated” error).
  + Logs **do not** show license text.
  + Artifacts **do not** contain the key (spot-check with strings or grep -i).
* Merge → Alpha deploy; verify app loads; run a smoke (login → dashboard).

**5.5 Promote (Beta → RTM → Prod)**

* Repeat the secret update (if per-env secrets differ) or **reuse** the same secret value across envs.
* Re-run CI; perform smoke on each env.

**6) Evidence (for 12 – Evidence Pack)**

Attach to the release:

* **CI logs** excerpt of “Activate Kendo UI license” step (no secrets shown).
* **Screenshot** of GitHub Environments → secret last updated timestamp (redact value).
* **Artifact scan** note: grepped build artifacts show no license string (commands + results).
* **Change ticket** link (request/approval).
* **Runbook link** (this doc) and date/time of rotation.

Retention: **≥ 1 year** with the release tag.

**7) Rollback Plan**

* If build fails post-rotation:
  1. Revert **KENDO\_UI\_LICENSE** to **previous value** in the affected environment(s).
  2. Re-run CI; confirm green.
  3. Open an incident if production was impacted; attach logs and revert diff to Evidence Pack.
* If a leak is suspected:
  1. **Rotate** the key **again**; invalidate compromised key via vendor portal.
  2. Purge caches/artifacts; re-run builds.
  3. Conduct secret-scan on repo and artifacts; document actions in incident report.

**8) Troubleshooting**

| **Symptom** | **Likely Cause** | **Fix** |
| --- | --- | --- |
| “License not activated / expired” during build | Activator step missing or wrong order | Ensure activation runs **before** npm run build; confirm secret present |
| CI logs print gibberish key text | Echoing env vars | Remove any echo/debug prints; rely on activator only |
| Works locally, fails in CI | Local license file present; CI clean | Always run activator in CI; do not commit license file |
| Activation step not found (npx error) | CLI package not resolved | Use npx kendo-ui-license activate **or** npx @progress/kendo-licensing activate per your project |
| Intermittent failure on forks/PRs | Secrets not available to external PRs | Limit web build for external PRs; use internal branches for full build |

**9) Security Notes & Policy**

* License key is **Confidential**; treat as a secret at rest and in transit.
* No secrets in repo, logs, or artifacts.
* Rotate on vendor cadence or upon personnel change/compromise.
* Include rotation in the **Quarterly controls review** (13 – Compliance).

**10) Related Docs**

* **10 – CI/CD**: evidence, gates, and path-casing checks
* **12 – Evidence Pack**: artifact list & retention
* **11 – Monitoring**: dashboard screenshots included at release
* **07a – DB Grants & SP Signature**: unrelated to license, but same Evidence Pack flow

**11) Acceptance Criteria**

* CI builds succeed in **Alpha → Beta → RTM → Prod** with activator step present.
* No license text in **logs** or **artifacts**.
* Evidence Pack updated with rotation proof.
* Rollback verified in a dry-run at least **once per year**.

**End of Runbook — TJ-MCPX-RB-04 v1.0.0**