

EKS Launch Template Configuration Cheat Sheet

Scenario	update_default_version	version	Behavior	When to Use
Initial Creation	FALSE	"1" or "\$Default"	Creates LT v1, nodes use v1	First-time setup
Safe Updates	FALSE	aws_launch_template.example.latest_version	Creates new LT version, rolls out nodes	✅ Recommended for all updates
Dangerous	TRUE	"\$Latest"	❌ Race condition, suspended nodes	❌ Never use this
No Rollout	FALSE	"5" (pinned)	Creates new LT version but no rollout	Testing changes safely
Broken	TRUE	"5" (pinned)	Wasted change: updates default but no rollout	❌ Never use this

When we have new changes

- Do this:
 - update_default_version = false
 - version = aws_launch_template.example.latest_version
- Not This:
 - update_default_version = true
 - version = \$Latest

Rollout Behavior:

Configuration	Creates New Version	Rolls Out Nodes	Safe
update_default_version=false + version=latest_version	✅ Yes	✅ Yes	✅ Yes
update_default_version=false + version="5"	✅ Yes	❌ No	✅ Yes
update_default_version=true + version="5"	✅ Yes	❌ No	❌ No
update_default_version=true + version="\$Latest"	✅ Yes	✅ Yes	❌ No

Recovery from suspended state

1. Check status

```
aws eks describe-nodegroup --cluster-name my-cluster --nodegroup-name my-ng
```

2. If suspended, update to known good version

```
aws eks update-nodegroup-config --cluster-name my-cluster --nodegroup-name my-ng --launch-template version="32" # Previous known good
```

3. Wait for recovery