

Getting Started With New KernelCI CLI Tools

Automating Linux Kernel Testing and Validation

Arisu Tachibana
KernelCI / Gentoo

Motivation: Why Kernel QA Is Hard

- Many trees, branches and configurations
- Multiple architectures, boards and toolchains
- CI dashboards are powerful but not ergonomic
- Developers still click UIs or write ad-hoc scripts
- Command-line tools keep context in the terminal

KernelCI in One Slide

- Upstream, open testing for the Linux kernel
- Builds, boots and tests across distributed labs
- Great data, but interaction is:
 - Mostly via web UI
 - Raw REST APIs
- Need something ergonomic in the terminal

Real-World Context: gentoo-sources

- Fast-moving tree with many patches
- High pressure to avoid regressions
- Need quick, scriptable visibility into CI results
- KernelCI CLI tools are built with this reality in mind

Introducing the KernelCI CLI Tools

- **kci-dev**
 - Developer-focused CLI to interact with KernelCI dashboards and Maestro
 - Installable from PyPI (`pip install kci-dev`) with shell completions for bash/zsh/fish
 - Ships `results` (dashboard) and Maestro-centric commands in one binary
- **kci-deploy** (*work in progress*)
 - Tool for deploying local / internal KernelCI maestro stacks
- **Goal:** make KernelCI a first-class tool in your terminal

kci-dev: What You Can Do Today

Dashboard (results) commands

```
kci-dev results summary --giturl https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git --branch master --history  
kci-dev results compare --giturl <git url> --branch <branch> <older> <newer>  
kci-dev results tests --giturl <git url> --branch <branch> --commit <sha>
```

Maestro commands (need config/token)

```
kci-dev checkout --branch <branch> --giturl <git url> --commit <sha>  
kci-dev testretry --id maestro:<node-id>  
kci-dev watch --id maestro:<node-id> --json
```

kci-dev: Everyday Workflow

1. Install & configure

- `pip install kci-dev` ; optional `virtualenv` + completions
- `kci-dev config` creates `~/.config/kci-dev/kci-dev.toml`
- Request API tokens from KernelCI GitHub issue template

2. Discover context

- `kci-dev results trees` to see tracked repos/branches
- `kci-dev results summary --history` for recent pass/fail trends

3. Inspect and compare

- `kci-dev results build|boot|test --id ... --download-logs`
- `kci-dev results compare` to spot regressions between commits

4. Retrigger or watch jobs

kci-dev: Quality-of-Life Details

- Rich output modes: table, JSON, quiet for scripts
- `--history` summaries with pass/fail/inconclusive color coding
- `compare` highlights regressions with dashboard/log links
- `results hardware` views boards + per-board summaries
- Profiles via `--instance` / `--settings` to switch endpoints
- Designed for piping into `jq`, `fzf`, notebooks

kci-deploy: For Lab Owners

- Simplifies standing up a Maestro stack
- Encodes best practices for networking and storage
- Shares the same UX patterns as kci-dev
- Early previews welcome: help shape the roadmap!

Automation Patterns

- **Pre-submit checks:** gate merges on KernelCI signal
- **Nightly reports:** email/Matrix summaries via cron
- **Release readiness:** track blockers for RCs
- **Local sanity tests:** run focused boards before shipping

Demo Script (Idea)

```
# Configure once
virtualenv .venv && source .venv/bin/activate
pip install kci-dev
kci-dev config # writes ~/.config/kci-dev/kci-dev.toml

# Morning health check
kci-dev results summary --giturl https://git.kernel.org/pub/scm/linux/kernel/git/torvalds/linux.git --branch master --history

# Investigate a failure
kci-dev results boot --id maestro:<node-id> --download-logs | less

# Nudge infra
kci-dev testretry --id maestro:<node-id>
```

Roadmap & Collaboration

- Deeper git integration (auto-pick branch/commit)
- Better diffing between runs
- Inline links back to dashboards
- kci-deploy installer previews in Q2
- Looking for testers, lab partners, and feedback

Getting Started After This Talk

- Try **kci-dev** with your tree this week
- Share your top 3 pain points in Kernel QA
- Join the KernelCI community calls / Matrix
- Contribute docs, plugins, and issue repros

Thank You!

Slides: <https://example.com>

@yourhandle

Questions welcome