How We Test: Controlled Host-Failure Drills, With the Timings Published

Our test method, the first two results, and the limits of what they show.

PREPARED BY KILAWATT CLOUD | OCT. 14, 2026 | DATA: KILAWATT PRODUCTION DATABASE | INTERNAL TEST ACCOUNT, TWO RUNS, COMPANY-PREPARED, NOT INDEPENDENT REPORTING

On Oct. 14, 2026 we conducted a controlled failover drill on an internal Kilawatt test account. A GPU workload wrote a checkpoint roughly once per minute. We then terminated its host machine at the supplier mid-run and measured the time to detect the failure, the time to provision a replacement on a different supplier, and the time to resume from the last checkpoint.

THE METHOD

Method

1. **Launch.** Start a short GPU workload on one supplier from an internal test account.
2. **Checkpoint.** The workload writes a numbered checkpoint about every 60 seconds.
3. **Inject the fault.** After checkpoint 4, we terminate the host machine at the supplier.
4. **Detect and replace.** Kilawatt detects the failure, requests a replacement on a different supplier, and the replacement is provisioned.
5. **Resume.** The workload reads back its latest checkpoint and continues.
6. **Record.** Every timestamp comes from the platform's own records, and we publish them.

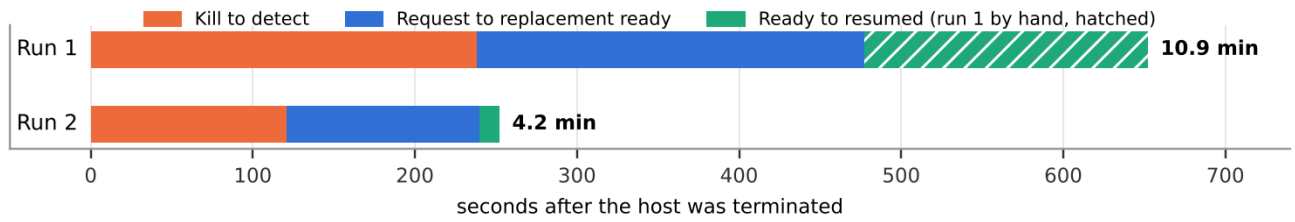
RESULTS

Two Counted Runs, One Excluded Run

Run	Path	Time to detect	Time to provision	Time to resume	Total recovery	Result
1	RunPod to Hyperstack	238 s	239 s	175 s (manual)	10.9 min	Failure detected, replacement provisioned, checkpoint intact. Resume was performed manually because of a defect in our test script, since corrected.
2	RunPod to Vast.ai	121 s	119 s	12 s (automatic)	4.2 min	Failure detected, replacement provisioned, resumed from checkpoint 4 automatically.
Excluded	Vast.ai, no kill	n/a	n/a	n/a	n/a	Stopped before the drill. Machine boot took 14.6 minutes, consuming most of the 20-minute job window. Not a test result.

Time figures are the platform's own timestamps. Source: Kilawatt production database.

FIGURE 1. Recovery time by phase, in seconds



Run 1's resume step was performed manually; run 2 resumed automatically. Source: Kilawatt production database.

WHAT THE RESULTS SHOW

Detection, Replacement and Recovery

Kilawatt detected both host failures without intervention and re-provisioned the workload on a different supplier. The latest checkpoint was intact on the replacement in both runs. In run 2 the workload resumed from checkpoint 4 automatically and went on to write checkpoints 5, 6 and 7. Billing was paused while the workload was down.

WHAT THE RESULTS DO NOT SHOW

The Limits

- **Sample size.** Two runs do not establish a reliability rate.
- **Recovery time.** Recovery is measured in minutes, not milliseconds, because a replacement must be provisioned and verified.
- **Detection latency.** 121 to 238 seconds is longer than our target. Reducing it is our first priority.
- **Workload.** A simple step-counting job on an internal test account. It is not a training run or a customer workload.
- **Independence.** We ran the test and prepared this report ourselves.

NEXT STEPS

Wider Coverage and Lower Latency

We will repeat the drill across additional supplier pairs, work to reduce detection latency, and publish each run in the same table, including unfavorable results.

DISCLOSURES

This document was prepared by Kilawatt Cloud about its own service and is not independent reporting. Both runs used an internal Kilawatt test account. No outside customer is involved. It is two runs, not a reliability rate, and contains no forecast and no offer. Kilawatt Cloud is a trade name of Right Recruit LLC.