Cost Ceiling Doctrine — UserPromptSubmit Main-Session Gate

Cost Ceiling Doctrine — UserPromptSubmit Main-Session Gate

Status: DRAFT — Awaiting Skillforge BookStack publication **MC:** #101419 **Author:** FlowForge / Kelsey Hightower **Date:** 2026-05-18

Why This Exists

On May 11, 2026, a single-day Opus spend of $377,487 occurred. The existing `opus-cost-guard.sh` hook was wired only to `PreToolUse[Task]` — it gated sub-agent dispatches but had zero visibility into main-session Opus usage. The cost events table in `costs.db` recorded everything post-session via the Stop hook (`claude-cli-cost-hook.sh`), creating a full-session lag before any gate could fire.

The 8-day cumulative burn at the time of this writing: $742K. This hook closes the main-session gap.

How It Works

The `userprompt-cost-guard.sh` hook fires on **every user message** via the `UserPromptSubmit` event — before Claude processes anything.

Data source: `~/system/databases/costs.db` (read-only, never written by this hook).

Query executed on each call:

```
SELECT COALESCE(SUM(cost_usd), 0)
FROM cost_events
WHERE date(timestamp,'localtime') = date('now','localtime')
  AND model LIKE 'claude-opus%'
```

Config file: `~/system/config/cost-ceilings.json`

The hook pins a sha256 of `cost-ceilings.json` in its script header and verifies integrity on every invocation. If the file is missing or tampered, the hook **fails open** (logs ERROR, exits 0) to avoid locking out the CEO.

Thresholds

Level	Threshold	Behavior
WARN	$400 (80% of $500 ceiling)	stdout injection — Claude sees the warning; session continues
BLOCK	$500 (100% of daily ceiling)	exit 2 — message blocked; JSON reason to stderr
KILLSWITCH	$1000 (200% of daily ceiling, multiplier=2.0)	BLOCK + `touch ~/system/state/killswitch` + reason JSON file

---

Alert-Only Grace Period (48h, per CEO D8)

Until the file `~/system/state/cost-guard-enforced` exists, the hook operates in **alert-only mode**:

- All blocking branches (BLOCK, KILLSWITCH) still log to the JSONL audit file
- Blocking branches print a WARN message to stdout instead of exiting 2
- The killswitch file is still written (as a paper trail), but exit code is 0

To activate enforcement:

```
touch ~/system/state/cost-guard-enforced
```

To deactivate enforcement (CEO override):

```
rm ~/system/state/cost-guard-enforced
```

This converts the hook back to alert-only mode without any code change.

How to Override Permanently

Two override mechanisms:

1. **Alert-only mode** (remove enforce marker, see above) — logging continues, no blocking.
2. **Raise ceiling** — edit `~/system/config/cost-ceilings.json` then update the `CEILINGS_SHA256` pin in the hook header to match the new file's sha256. Run: `shasum -a 256 ~/system/config/cost-ceilings.json`

Do NOT delete `cost-ceilings.json` — that triggers fail-open with an ERROR log entry.

Audit JSONL Schema

Every hook invocation appends one line to: `~/.cache/userprompt-cost-guard-YYYYMMDD.jsonl`

Schema:

```
{
  "timestamp": "2026-05-18T12:34:56Z",
  "verdict": "ALLOW | WARN | BLOCK | KILLSWITCH | SKIP | ERROR",
  "reason": "within_ceiling | daily_opus_warn_threshold_pct80 |
daily_main_session_ceiling_breach | daily_opus_killswitch_multiplier_breach | costs_db_missing
| ceilings_file_missing | ceilings_sha256_mismatch_actual=<hash> | spend_parse_error",
  "spend_usd": 423.50,
  "ceiling_usd": 500
}
```

Files

File	Purpose
`~/.claude/hooks/userprompt-cost-guard.sh`	Hook script (chmod 755)

File	Purpose
`~/system/config/cost-ceilings.json`	Ceiling thresholds (chmod 644)
`~/system/config/opus-allowlist.json`	Historical Opus subagent types (docs only)
`~/system/state/cost-guard-enforced`	Presence = enforcement active
`~/system/state/killswitch`	Presence = killswitch triggered
`~/system/state/killswitch.reason.json`	Killswitch trigger metadata
`~/.cache/userprompt-cost-guard-YYYYMMDD.jsonl`	Per-day audit JSONL
`~/system/tests/userprompt-cost-guard-test.sh`	D2 Proveo test harness

Registration in settings.json

Hook is registered under `hooks.UserPromptSubmit[].hooks`:

```
{
  "type": "command",
  "command": "bash ~/.claude/hooks/userprompt-cost-guard.sh",
  "timeout": 8000
}
```

Related Systems

- `opus-cost-guard.sh` — PreToolUse[Task] gate (sub-agent level; still active)
- `claude-cli-cost-hook.sh` — Stop hook writes cost_events to costs.db post-session
- `spend-limits.json` — separate spend limit config (infra-level, not hook-level)
- MC #101419 — implementation task

Revision #2
Created 2026-05-19 15:54:55 UTC by John
Updated 2026-06-21 20:03:48 UTC by John