

Ops Agent Runbook

“ Last Verified: 2026-02-17 | Owner: John

Runbook: Ops Agent

Service: com.john.ops-agent **Type:** LaunchAgent daemon (Node.js) **Interval:** Every 5 minutes (300s) **Location:** ~/system/daemons/ops-agent.js **Plist:** ~/Library/LaunchAgents/com.john.ops-agent.plist **Owner:** John **Cost:** \$0 (Ollama local AI)

What It Does

Autonomous operations agent that runs 24/7:

1. **MM Monitoring** — reads all 4 Mattermost teams (basic, wizard, rendrom, riad)
2. **Message Classification** — Ollama llama3.1:8b classifies: ROUTINE / TASK / INCIDENT
3. **Intelligent Response** — Ollama qwen2.5-coder:32b generates contextual MM replies
4. **Task Creation** — creates MC tasks with BILLABLE/INTERNAL tag + Planka cards
5. **Health Monitoring** — runs health-check.js (Docker, HTTP, system, daemons)
6. **Auto-Fix** — auto-fix.js for known issues (max 3 attempts/hour/service)
7. **Escalation** — creates HIGH priority MC task + MM alert when it can't resolve

Status Check

```
# Is it running?
launchctl list | grep ops-agent

# Recent activity
tail -50 ~/system/logs/ops-agent.log

# LaunchAgent stdout/stderr
tail -20 ~/system/logs/ops-agent-launchd.log
```

```
tail -20 ~/system/logs/ops-agent-launchd-error.log
```

```
# State file
```

```
cat /tmp/ops-agent-state.json
```

```
# Stats
```

```
cat ~/system/agents/state/ops.json
```

Restart

```
# Graceful restart (unload + load)
```

```
launchctl unload ~/Library/LaunchAgents/com.john.ops-agent.plist
```

```
launchctl load ~/Library/LaunchAgents/com.john.ops-agent.plist
```

```
# Verify
```

```
launchctl list | grep ops-agent
```

Manual Run (Testing)

```
# Run one cycle manually
```

```
node ~/system/daemons/ops-agent.js
```

```
# Watch output in real-time
```

```
node ~/system/daemons/ops-agent.js 2>&1 | tee /dev/tty
```

Troubleshooting

Ops agent not running

```
# Check if loaded
```

```
launchctl list | grep ops-agent
```

```
# Expected: "- 0 com.john.ops-agent"
```

```
# If not loaded:
launchctl load ~/Library/LaunchAgents/com.john.ops-agent.plist

# If load fails, check plist:
plutil -lint ~/Library/LaunchAgents/com.john.ops-agent.plist
```

Not processing messages

```
# Check state – is last_check_ms recent?
cat /tmp/ops-agent-state.json | python3 -m json.tool

# Check MM connectivity
node ~/system/tools/mm.js status

# Check MM token
cat /tmp/mm-token.json | python3 -m json.tool

# Verify Mattermost is up
curl -s http://localhost:8065/api/v4/system/ping
```

Classification wrong (Ollama issues)

```
# Check Ollama is running
curl -s http://localhost:11434/api/tags | python3 -m json.tool

# Test classification manually
curl -s http://localhost:11434/api/generate -d '{
  "model": "llama3.1:8b",
  "prompt": "Classify: ROUTINE, TASK, or INCIDENT. Reply ONE word.\n\nMessage: Can you fix the
login page?",
  "stream": false,
  "options": {"temperature": 0.1, "num_predict": 10}
}' | python3 -c "import sys,json; print(json.load(sys.stdin)['response'])"

# If Ollama down, ops-agent falls back to keyword heuristics (still works)
```

Health check reporting false positives

```
# Run health check directly
node ~/system/tools/health-check.js

# JSON output for debugging
node ~/system/tools/health-check.js --json 2>/dev/null | python3 -m json.tool

# Check specific service
docker ps --format '{{.Names}} {{.Status}}' | grep <service>
```

Auto-fix loop (service keeps restarting)

```
# Check fix history (max 3/hour enforcement)
cat /tmp/ops-fix-history.json | python3 -m json.tool

# Clear fix history (reset counter)
rm /tmp/ops-fix-history.json

# Check auto-fix directly
node ~/system/tools/auto-fix.js <service> <issue>
```

Planka card not created

```
# Check Planka is up
curl -s http://localhost:3100/api/access-tokens -X POST \
  -H "Content-Type: application/json" \
  -d '{"emailOrUsername":"john","password":"BasicAS2026!"}'

# Check ops-agent log for Planka errors
grep "Planka" ~/system/logs/ops-agent.log | tail -10
```

Dependencies

Service	Required	Fallback
Mattermost (8065)	YES	Agent skips MM check cycle
Ollama (11434)	NO	Falls back to keyword classification
MC (mc.js)	YES	Tasks not created (error logged)

Service	Required	Fallback
Planka (3100)	NO	Cards not created (task still created in MC)
HiveMind	NO	Intel not posted (ops still works)

Configuration

Monitored MM Teams

Defined in ops-agent.js. Currently: basic, wizard, rendrom, riad

Ignored Users (bots)

john, edita, system-bot, boards, calls, tester — defined by user ID in ops-agent.js

Billable Logic

- `basic` team = INTERNAL (not billable)
- `wizard`, `rendrom`, `riad` = BILLABLE (client teams)

Health Check Services

Defined in health-check.js:

- 8 Docker containers
- 6 HTTP endpoints
- 2 system checks (disk, memory)
- 4 LaunchAgent daemons

Files

File	Purpose
~/system/daemons/ops-agent.js	Main daemon code
~/Library/LaunchAgents/com.john.ops-agent.plist	LaunchAgent config
~/system/tools/health-check.js	Service health monitor

File	Purpose
~/system/tools/auto-fix.js	Automated recovery
~/system/agents/identities/ops.md	Agent identity card
~/system/agents/state/ops.json	Persistent state
/tmp/ops-agent-state.json	Runtime state (last check timestamp)
/tmp/mm-token.json	Cached MM auth token
/tmp/ops-fix-history.json	Auto-fix attempt tracking
~/system/logs/ops-agent.log	Activity log
~/system/logs/ops-agent-launchd.log	LaunchAgent stdout
~/system/logs/ops-agent-launchd-error.log	LaunchAgent stderr

Disaster Recovery

Complete reset

```
# 1. Stop daemon
launchctl unload ~/Library/LaunchAgents/com.john.ops-agent.plist

# 2. Clear state
rm -f /tmp/ops-agent-state.json /tmp/mm-token.json /tmp/ops-fix-history.json

# 3. Restart
launchctl load ~/Library/LaunchAgents/com.john.ops-agent.plist

# Note: First run will check messages from last 30 minutes only (default)
```

Rollback to mm-responder

```
# 1. Stop ops-agent
launchctl unload ~/Library/LaunchAgents/com.john.ops-agent.plist

# 2. Restore mm-responder
cp ~/system/archive/mm-responder.sh.archived-2026-02-10 ~/system/daemons/mm-responder.sh
chmod +x ~/system/daemons/mm-responder.sh
```

```
launchctl load ~/Library/LaunchAgents/com.john.mm-responder.plist
```

```
# 3. Update health-check.js daemon list (add mm-responder, remove ops-agent)
```

Metrics

Check via MC:

```
node ~/system/tools/mc.js stats          # Task creation stats
node ~/system/tools/mc.js list --owner ops # Tasks created by ops-agent
```

Check via state:

```
cat ~/system/agents/state/ops.json      # Cumulative stats
cat /tmp/ops-agent-state.json           # Current cycle stats
```

Created: 2026-02-10 **Last Updated:** 2026-02-10 **Next Review:** 2026-03-10

Revision #3

Created 2026-02-17 22:14:31 UTC by John

Updated 2026-05-31 20:00:42 UTC by John