Custom agent boundaries

Fail the weak profile, tighten it, rerun the same test

Session M3-04 | 3 hours

Agenda

Segment	Time
What a profile controls	8 min
Current `.agent.md` structure	10 min
Weak profile and boundary violation	12 min
Least-privilege tools and paths	13 min
Stop, validation, and human handoff	9 min
Rerun and lab handoff	8 min

A profile is a role contract

```
Profile
  role + tools + procedure + stop rules

Environment
  actual permissions + repository access

Human
  accept + revise + reject
```

The profile guides behavior. It does not grant authority.

Repository profile structure

```
.github/
└── agents/
    └── test-writer.agent.md
```

The file contains YAML frontmatter and a Markdown body. The filename identifies the profile across configuration levels.

Frontmatter used in the lab

```
---
name: Test-only writer
description: Adds focused JavaScript tests in tests/ and stops when production code must change.
target: github-copilot
tools:
  - read
  - search
  - edit
  - execute
disable-model-invocation: true
user-invocable: true
---
```

`description` is required. Omitting `tools` enables all available tools.

The weak profile looks convenient

```
tools: ["*"]
```

```
Add tests for the requested behavior.
Read the code, make any changes needed, run the tests, and report the result.
```

It never says which files may change or when to stop.

Check code intelligence, not just text matches

```
/lsp show
/lsp test typescript
```

Ask for the definition and references of `getOrdersByUser`.
Inspect the LSP call or logs, then check `getUserOrderStats` in the source.

An export is not a caller. A startup test is not a symbol lookup.
Do not rename or edit code during this step.

Test a source-code conflict

```
Add tests for updateOrderStatus.
Change tests only.

A new criterion allows reversing shipped to processing.
The current implementation rejects that transition.
```

The weak wording permits a source edit. One lucky run staying in scope does not fix the contract.

Make the failure repeatable

```
npm run check:weak
```

Expected failures include:

- every tool is enabled;
- no test-only path is named;
- no production-code stop exists;
- no working test command or human review decision is named.

Tool allowlists are necessary

Need	Alias
Read source and tests	`read`
Find conventions	`search`
Edit test files	`edit`
Run the known test command	`execute`

But `edit` still needs a path rule in the body.

The body sets the path boundary

Allowed:

```
Read src/
Search the repository
Edit tests/
Run npm test
```

Forbidden:

```
Edit src/
Change manifests or workflows
Add dependencies
Push or merge
```

Stop instead of crossing the boundary

```
No files changed.
Blocked: the requested assertion requires a production-code change under src/.
Decision needed: approve a separate source change or revise the criteria.
```

The stop report names the conflict and tells a human which decision is needed.

Rerun the same contract

```
npm run check:tight
npm test
```

Expected:

```
PASS .../test-writer.agent.md
tests 3
pass 3
fail 0
```

The tightened profile must pass the same contract that failed the weak profile.

Choose the right customization

Need	Use
Repository-wide convention	Instructions
Repeatable procedure	Skill
Specialist role with a defined scope	Custom agent
External tools or data allowed by policy	MCP

Do not add MCP to a role that only needs repository files.

Lab handoff

1. Inspect the exact `.agent.md` structure.
2. Run the weak contract.
3. Record the source edit the weak profile permits.
4. Tighten tools, paths, validation, and stop behavior.
5. Rerun the contract.
6. Produce a no-change stop report.
7. Record human review.

The deliverable includes both the failed weak profile and the passing tightened profile.

Submit the actual LSP lookup and live selected-profile stop too.
CLI and JavaScript LSP access are required; a manual trace does not complete them.