

4 Deterministic Probes (MCs #101133-#101136)

4 Deterministic Probes (MCs #101133-#101136)

Parent MC: #101065 (Deterministic Session Compiler — expanded scope)

Owner: CodeCraft

Date Shipped: 2026-05-17

Registry: `~/system/probes/registry.json`

Overview

These 4 probes are the foundation of the Reality Anchor doctrine: external, non-LLM, deterministic tools that produce structured JSON output as evidence. The LLM cannot write probe output. The LLM is removed from the evidence chain entirely.

“ **Petter Graff (Unified Fix):** "Before any agent can mark evidence as valid, require invocation of an external, non-LLM, deterministic probe against the actual system. The probe output IS the evidence."

Probe Registry

All probes are registered in `~/system/probes/registry.json` with:

- **claim_class** — category of claim (login_works, commit_verified, ally_count, test_count)
- **script** — absolute path to probe executable
- **invocation** — command template with parameter placeholders
- **output_schema** — JSON schema for probe output

- **exit_codes** — meaning of 0/1/2/3 exit codes
- **smoke_test** — path to test script

Probe 1: login-probe.sh (MC #101133)

Claim Class: login_works

Script: ~/system/probes/login-probe.sh

Purpose: Deterministic login verification against a URL

Invocation

```
bash ~/system/probes/login-probe.sh \  
  --url https://demo.bilko.cloud/api/auth/login \  
  --user test@example.com \  
  --pass-bw "Bilko Demo Login"
```

Or with credentials from Bitwarden item:

```
bash ~/system/probes/login-probe.sh \  
  --url https://demo.bilko.cloud/api/auth/login \  
  --credentials "Bilko Demo Login"
```

Output Schema

```
{  
  "claim_class": "login_works",  
  "timestamp": "2026-05-17T10:30:45Z",  
  "url": "https://demo.bilko.cloud/api/auth/login",  
  "success": true,  
  "http_status": 200,  
  "latency_ms": 342,  
  "session_cookie_set": true,  
  "me_endpoint_check": true  
}
```

Exit Codes

Code	Meaning
0	Login success (HTTP 2xx + session cookie present)
1	Login failed (non-2xx or no session cookie)
2	Network error (timeout, DNS failure)
3	Invalid arguments (missing --url or credentials)

Test

```
bash ~/system/probes/test-login-probe.sh
```

Probe 2: git-diff-probe.sh (MC #101134)

Claim Class: `commit_verified`

Script: `~/system/probes/git-diff-probe.sh`

Purpose: Deterministic commit verification against baseline

Invocation

```
bash ~/system/probes/git-diff-probe.sh \  
  --repo /Users/makinja/projects/bilko \  
  --baseline main \  
  --expected-shas a3f8bc4,d9e2f01,c5b7a93
```

Or enumerate all commits without expected list:

```
bash ~/system/probes/git-diff-probe.sh \  
  --repo /Users/makinja/projects/bilko \  
  --baseline v1.2.0
```

Output Schema

```
{  
  "claim_class": "commit_verified",  
  "timestamp": "2026-05-17T10:32:18Z",  
  "repo": "/Users/makinja/projects/bilko",
```

```
"baseline": "main",
"actual_shas": ["a3f8bc4", "d9e2f01", "c5b7a93"],
"expected_shas": ["a3f8bc4", "d9e2f01", "c5b7a93"],
"missing": [],
"unexpected": [],
"match": true
}
```

Exit Codes

Code	Meaning
0	Exact match or enumeration complete (no expected list)
1	Mismatch: missing or unexpected SHAs
2	Git error (repo not found, invalid SHA)

Test

```
bash ~/system/probes/test-git-diff-probe.sh
```

Probe 3: playwright-a11y-probe.js (MC #101135)

Claim Class: `ally_count`

Script: `~/system/probes/playwright-a11y-probe.js`

Purpose: Deterministic accessibility violation count via Playwright + axe-core

IMPORTANT: Requires `npm install` in `~/system/probes/` directory (Playwright + axe dependencies).

Invocation

```
node ~/system/probes/playwright-a11y-probe.js \  
  --url https://snowit.ba \  
  --max-critical 0 \  
  --max-serious 2
```

Output Schema

```
{
  "claim_class": "ally_count",
  "timestamp": "2026-05-17T10:35:22Z",
  "url": "https://snowit.ba",
  "violations": {
    "critical": 0,
    "serious": 1,
    "moderate": 3,
    "minor": 5
  },
  "thresholds": {
    "critical": 0,
    "serious": 2
  },
  "gate_pass": true,
  "detail_path": "/tmp/ally-violations-101065.json"
}
```

Exit Codes

Code	Meaning
0	gate_pass true (violations within thresholds)
1	gate_pass false (violations exceed thresholds)
2	Playwright error (install missing, network, page load failure)

Test

```
bash ~/system/probes/test-playwright-ally-probe.sh
```

Setup

```
cd ~/system/probes
npm install
npx playwright install chromium
```

Probe 4: test-enumeration.sh (MC #101136)

Claim Class: `test_count`

Script: `~/system/probes/test-enumeration.sh`

Purpose: Deterministic test case enumeration across frameworks (Jest, Playwright, Vitest, JUnit)

Invocation

```
bash ~/system/probes/test-enumeration.sh \  
  --repo /Users/makinja/projects/bilko \  
  --pattern '**/*.test.ts' \  
  --framework jest
```

Or auto-detect framework:

```
bash ~/system/probes/test-enumeration.sh \  
  --repo /Users/makinja/projects/bilko
```

Output Schema

```
{  
  "claim_class": "test_count",  
  "timestamp": "2026-05-17T10:38:45Z",  
  "repo": "/Users/makinja/projects/bilko",  
  "framework": "jest",  
  "pattern": "**/*.test.ts",  
  "file_count": 23,  
  "test_count": 147,  
  "breakdown": {  
    "src/auth/auth.test.ts": 12,  
    "src/invoices/invoices.test.ts": 18,  
    "src/reports/reports.test.ts": 9  
  }  
}
```

Exit Codes

Code	Meaning
0	Enumeration complete
2	Repo not found or invalid path

Test

```
bash ~/system/probes/test-test-enumeration.sh
```

Probe Execution Wrapper

All probes can be executed via the universal wrapper:

```
node ~/system/probes/run-probe.js \  
  --claim-class login_works \  
  --url https://demo.bilko.cloud/api/auth/login \  
  --credentials "Bilko Demo Login"
```

The wrapper:

- Resolves probe script from registry.json
- Validates parameters against schema
- Runs probe in sandboxed environment (via `probe-sandbox.sb` if available)
- Captures JSON output + exit code
- Writes cryptographic seal to output (SHA-256 hash + timestamp + task_id)

Probe Output Sealing

Each probe output is sealed with:

```
{  
  "seal": {  
    "sha256": "a3f8bc4d9e2f01c5b7a93...",  
    "timestamp": "2026-05-17T10:40:12Z",  
    "task_id": 101065,  
  }  
}
```

```
"probe": "login-probe.sh",
"exit_code": 0
},
"data": { ... }
}
```

The seal is verified at `mc.js ready/done` via `~/system/probes/verify-seal.js`.

Related

- **Parent MC:** #101065 (Deterministic Session Compiler)
 - **Child MCs:** #101133 (login), #101134 (git-diff), #101135 (a11y), #101136 (test-enum)
 - **Schema Injector:** [Schema Stub Gate + Claim Schema Injector](#)
 - **Reality Anchor Doctrine:** [v1 Final](#)
-

Revision #2

Created 2026-05-17 12:05:06 UTC by John

Updated 2026-06-21 20:03:41 UTC by John