@demotoprod · KEYS · also LEVERS

MCP security checklist

A local MCP server is a child process of your account. It may inherit your environment variables, can read any file the account can, and can open outbound connections anywhere. The authors of a 2026 study of 15,465 published servers found no guardrails and no review. Five afternoon checks.

1 Give each server its own narrow, read-only token

A local server may inherit the environment of whatever launched it, so tokens exported in a shell profile may be readable by every server. A stolen broad token opens unrelated tools; the protocol's guide lists wildcard scopes as a common mistake.

- Give each server its own token in its own config, smallest scope that works, read-only if possible.
- Grant one project directory, read-only, never your home directory. Rotate what a removed server held.

```
# the server's environment holds PATH and one token
env -i PATH="$PATH" SERVER_TOKEN="$SERVER_A_TOKEN" server-a --stdio
```

2 Pin the exact version and the image digest

A package version pinned through a launcher fixes only the top-level package; dependencies are typically resolved fresh at install time. An image pinned by digest freezes the whole tree. For a remote server, the study's authors say, review shows what was published, not what runs.

- Install an exact version, never latest, from a publisher tied to a source repository. Disable auto-update.
- Run the server from an image pinned by digest. Re-check pinned servers when advisories land.

3 Re-read tool descriptions and re-approve when they change

A model reads tool descriptions as instructions, and a server can change them after approval, so a version pin does not cover them. The protocol lets the tool list change over time and calls tool annotations untrusted unless the server is trusted.

- List each server's tools and descriptions with a tool inspector, no model involved. Remove any server whose descriptions carry unrelated instructions.
- Choose a client that asks again when tool definitions change. Keep the server list short.

4 Allow-list each server's network and log every call

A local process normally has unrestricted outbound access, which turns any overcollection or compromise into data loss. A server wrapping one API has a short, predictable set of destinations. Configuration-level allow-lists are cooperative; a sandbox or proxy enforces them.

- Run each server in a sandbox with no network, or behind an egress proxy with a domain allow-list.
- Log time, server, tool and inputs of every call. Alert on destinations outside the list.

5 Delete standing always-allow grants

In one security vendor's test on an older model, a single always-allow grant let a malicious server use a later prompt injection for privileged file access with no further confirmation. The same attack did not succeed against two newer models.

- Delete standing allow rules for servers that read untrusted content: web pages, mail, tickets.
- Show the inputs and ask a person before any call that writes, deletes, sends or spends.

Nitish Gautam · indica-tech.com · hello@indica-tech.com

INDICA TECH · @DEMOTOPROD

Nitish Gautam

Founder-led AI engineering. Demo to production, under NDA, in regulated industries, on fixed scope.

Paste into your project

A PER-SERVER POLICY TO FILL IN

```
# One block per server; names in capitals are yours to set
server-a:
  version: EXACT_VERSION          # never latest, no auto-update
  image: IMAGE@sha256:DIGEST      # pinned by digest
  env: [SERVER_A_TOKEN]           # its own narrow, read-only token
  files: {path: PROJECT_DIR, mode: read-only}   # never home
  network: none                   # or a proxy with ALLOWED_DOMAINS
  always_allow: []                # no standing grants
  confirm: [write, delete, send, spend]
  log: [time, server, tool, inputs]
  tool_descriptions: reapprove_on_change
```

Where this list comes from

Ten-plus years taking products from zero to one across seven sectors, including defence, fintech, medical and telecom, with AI shipped into regulated and air-gapped environments. The @demotoprod channel takes one production AI failure apart each day. These checklists are the same work written down: every item is a dated failure or a documented default, with the control that stops it and the primary source.

If this list found a gap

1. ### Your demo works and production is stuck.

   Production Readiness Audit. Fixed price, two weeks. A five-dimension scorecard, cost and failure analysis, and a 90-day execution roadmap.

2. ### You need senior AI judgement without a full-time hire.

   Fractional AI Lead. Monthly: 90-day minimum, then 30 days' notice either side. Weekly 1:1 with the founder or CTO, architecture and code review, technical hiring support, investor-ready due diligence.

3. ### An AI feature has to work, by a date.

   AI Build Sprint. Fixed price, four to eight weeks. Scope signed off up front, weekly demos, CI/CD and observability included, 30-day production warranty.

4. ### A regulator, an auditor or a customer's security team is asking.

   AI Governance & Compliance. Fixed price, four weeks. ISO 27001 / SOC 2-aligned controls, a model risk framework, human-in-the-loop and audit logging, regulator-ready documentation.

BOOK A CALL

indica-tech.com/#contact

EMAIL

hello@indica-tech.com

WEBSITE

indica-tech.com

ALL CHECKLISTS

indica-tech.com/checklists/

YOUTUBE

youtube.com/@demotoprod

LINKEDIN

linkedin.com/in/startup-founderx

INSTAGRAM

instagram.com/demotoprod

TIKTOK

tiktok.com/@demotoprod

Free to share. The live list, with every source link: indica-tech.com/checklists/#mcp-permissions