Secure Development with Claude Code

CLI Security Checklist · 20 items across 5 categories

20-item pre-session checklist

resource.7312.us/guide/secure-claude-code.html

FILE ACCESS CONTROL

- [ ] Create a .claudeignore file at the repo root before first use — CRITICAL
- [ ] Block .env, *.pem, *.key, .ssh/, .aws/credentials, and all credential paths — CRITICAL
- [ ] Block kubeconfig, *.tfvars, and infrastructure secrets — CRITICAL
- [ ] Review .claudeignore whenever adding new secret file types to the project — IMPORTANT

PERMISSION CONFIGURATION

- [ ] Define an explicit allow list of safe commands in .claude/settings.json — CRITICAL
- [ ] Add git push, git commit, npm publish, curl, and wget to the deny list — CRITICAL
- [ ] Never use --dangerously-skip-permissions on production or credential-bearing repos — CRITICAL
- [ ] Review your allowlist periodically to remove stale entries — IMPORTANT

MCP SERVER SECURITY

- [ ] Enable only MCP servers you have reviewed and explicitly trust — CRITICAL
- [ ] Scope MCP permissions to specific tools — not wildcards (mcp__server__*) — CRITICAL
- [ ] Disable the filesystem MCP server unless strictly required — IMPORTANT
- [ ] Store MCP server credentials in environment variables, never inline in config — CRITICAL

INJECTION DEFENSE & REVIEW

- [ ] Review every git diff before approving commits from Claude Code — CRITICAL
- [ ] Run Claude Code in a VM or container when exploring untrusted codebases — IMPORTANT
- [ ] Check for unintended file creation with git status after each session — IMPORTANT
- [ ] Scan CLAUDE.md and .claude/ for injection patterns before each session — IMPORTANT

CLAUDE.MD & OPERATIONAL HYGIENE

- [ ] Add a security rules section to CLAUDE.md explicitly forbidding credential access — CRITICAL
- [ ] Version control CLAUDE.md and review changes like you would code — IMPORTANT
- [ ] Keep Claude Code updated to the latest version for security patches — IMPORTANT
- [ ] Always start work on a new branch — never directly on main — GOOD PRACTICE