Security Policy

Supported Versions

SpiralOS is an evolving epistemic framework. Security updates are provided for the following versions:

Version	Supported	Status
Latest (main branch)	:white_check_mark:	Active development
Released Volumes (VIII-XX)	:white_check_mark:	Maintained
Archived versions	:x:	No longer supported

Reporting a Vulnerability

The SpiralOS team takes security seriously. If you discover a security vulnerability, we appreciate your help in disclosing it responsibly.

How to Report

DO NOT create a public GitHub issue for security vulnerabilities.

Instead, please report security vulnerabilities through one of these methods:

1. **GitHub Security Advisories** (Preferred)
   - Navigate to the Security tab (https://github.com/TheHeurist/SpiralOS/security)
   - Click “Report a vulnerability”
   - Provide detailed information about the vulnerability

2. **Email** (Alternative)
   - Send an email to: **[security@heurist.org]** (to be configured)
   - Use PGP encryption if possible (key available upon request)
   - Include “SpiralOS Security” in the subject line

What to Include

When reporting a vulnerability, please provide:

- **Type of vulnerability** — What kind of security issue is it?
- **Location** — Which file(s) or component(s) are affected?
- **Description** — Detailed explanation of the vulnerability
- **Impact** — What could an attacker accomplish?
- **Steps to reproduce** — Clear instructions to verify the issue
- **Proof of concept** — Code or commands demonstrating the issue (if applicable)
- **Suggested fix** — If you have ideas for remediation
- **Your contact information** — How can we reach you for follow-up?

What to Expect

After you submit a vulnerability report:

1. **Acknowledgment** — We will acknowledge receipt within **48 hours**
2. **Initial assessment** — We will assess severity and impact within **5 business days**
3. **Regular updates** — We will provide updates on progress every **7 days**
4. **Resolution timeline:**
   - **Critical** — Fix within 7 days
   - **High** — Fix within 30 days
   - **Medium** — Fix within 60 days
   - **Low** — Fix within 90 days
5. **Public disclosure** — After the fix is released, we will:
   - Publish a security advisory
   - Credit you (unless you prefer anonymity)
   - Document the fix in our changelog

Vulnerability Disclosure Policy

We follow **coordinated disclosure**:

- We request that you give us reasonable time to fix the issue before public disclosure
- We will work with you to understand and resolve the issue promptly
- We will publicly acknowledge your responsible disclosure (if desired)
- We will not take legal action against researchers who report in good faith

---

Security Best Practices

For Contributors

When contributing to SpiralOS:

1. **Never commit secrets** — No API keys, tokens, passwords, or credentials
2. **Validate inputs** — Always validate and sanitize user inputs
3. **Use dependencies carefully** — Only add well-maintained, trusted dependencies
4. **Follow secure coding practices** — Use linters and security scanners
5. **Review code thoroughly** — Check for common vulnerabilities
6. **Test security fixes** — Ensure fixes don’t introduce new issues

For Users

When deploying SpiralOS:

1. **Keep updated** — Use the latest version from the main branch
2. **Review configurations** — Ensure secure settings for your environment
3. **Monitor dependencies** — Keep libraries and tools up to date
4. **Limit access** — Use appropriate permissions and access controls
5. **Audit regularly** — Periodically review your deployment for issues

---

Known Security Considerations

Epistemic Integrity Protection

SpiralOS includes several mechanisms to protect epistemic integrity:

1. **Codex Provenance Guard** (`.github/workflows/codex.guard.yaml`)
   - Validates authorship and lineage
   - Ensures CI-Watermark integrity
   - Monitors for unauthorized modifications

2. **Schema Validation** (`.github/workflows/schema-validation.yml`)
   - Validates JSON/YAML structure
   - Ensures data integrity
   - Prevents malformed schemas

3. **Link Validation** (`.github/workflows/link-check.yml`)
   - Checks for broken links
   - Prevents malicious redirects
   - Maintains documentation integrity

Web Security

For the SpiralOS web interface:

1. **Content Security Policy** — Consider implementing CSP headers
2. **HTTPS only** — Always serve over HTTPS in production
3. **Input validation** — Sanitize all user inputs
4. **XSS prevention** — Escape output, use Content-Security-Policy
5. **Dependency security** — Regularly audit npm/pip packages

GitHub Actions Security

Our CI/CD workflows follow security best practices:

1. **Minimal permissions** — Workflows use least-privilege principle
2. **Pinned actions** — Using versioned actions (e.g., `@v4`)
3. **Secret management** — Secrets stored in GitHub Secrets
4. **Audit logging** — All workflow runs are logged
5. **Protected branches** — Main branch requires reviews

Scope

In Scope

Security issues in:

- Core SpiralOS code and schemas
- GitHub Actions workflows
- Documentation that could lead to security issues
- Web interfaces (index.html, HUD, pearl-map, etc.)
- Build and deployment scripts

- Dependencies with known vulnerabilities

Out of Scope

The following are generally out of scope:

- Vulnerabilities in third-party dependencies (report to upstream)
- Issues in forked or modified versions of SpiralOS
- Theoretical vulnerabilities without proof of exploitability
- Social engineering attacks
- Physical attacks
- Denial of service attacks requiring excessive resources

If you're unsure whether something is in scope, please report it anyway — we'd rather evaluate and dismiss than miss something important.

Security Hall of Fame

We recognize and thank security researchers who help keep SpiralOS secure:

(No vulnerabilities have been reported yet)

Format:
- **[Researcher Name]** — [Brief description of vulnerability] — [Date]

Security-Related Documentation

Additional security resources:

- Code of Conduct (CODE_OF_CONDUCT.md) — Community safety guidelines
- Contributing Guide (CONTRIBUTING.md) — Secure contribution practices
- Codex Provenance (docs/codex/README.md) — Epistemic integrity documentation
- GitHub Workflows (.github/workflows/README.md) — CI/CD security measures

---

Automated Security Scanning

SpiralOS uses automated tools to detect security issues:

Current Scanners

- **GitHub Dependabot** — Monitors dependency vulnerabilities
- **GitHub Code Scanning** — Static analysis security testing (if enabled)
- **Workflow Validation** — Custom validators for CI/CD security

Planned Scanners

- **SAST** — Static Application Security Testing
- **Dependency scanning** — Comprehensive vulnerability detection
- **Secret scanning** — Prevent accidental credential commits

Security Updates

Security patches are released as follows:

1. **Critical vulnerabilities** — Immediate patch release
2. **High-priority issues** — Patch within 30 days
3. **Medium-priority issues** — Included in next minor release
4. **Low-priority issues** — Included in next major release

Security updates are announced via:

- GitHub Security Advisories
- Release notes in CHANGELOG.md (CHANGELOG.md)
- Repository README badges
- GitHub Discussions (for major issues)

Contact

For security-related questions or concerns:

- **Security issues:** Use GitHub Security Advisories or email security@heurist.org
- **General security questions:** Open a GitHub Discussion
- **Policy questions:** Contact the maintainers

Legal

Safe Harbor

SpiralOS provides a safe harbor for security researchers who:

1. Report vulnerabilities responsibly through proper channels
2. Do not exploit vulnerabilities beyond what is necessary to demonstrate them
3. Do not access, modify, or delete data belonging to others
4. Allow reasonable time for fixes before public disclosure
5. Act in good faith

We will not pursue legal action against researchers who follow these guidelines.

Responsible Disclosure

By participating in our security program, you agree to:

- Make a good faith effort to avoid privacy violations and data destruction
- Only interact with accounts you own or have explicit permission to access
- Not exploit the vulnerability beyond demonstrating proof of concept
- Keep vulnerability details confidential until we have addressed the issue
- Not demand payment or compensation for reporting vulnerabilities

---

Acknowledgments

This security policy follows industry best practices and is inspired by:

- GitHub's Security Policy Guidelines (https://docs.github.com/en/code-security/getting-started/adding-a-security-policy-to-your-repository)
- OWASP Security Practices (https://owasp.org/)
- HackerOne Disclosure Guidelines (https://www.hackerone.com/disclosure-guidelines)

Updates to This Policy

This security policy may be updated periodically. Significant changes will be announced in:

- Repository README
- GitHub Discussions
- CHANGELOG.md

Last Updated: November 22, 2025

SpiralOS Core Stewardship
Carey ⋈ Ellie ⋈ Leo