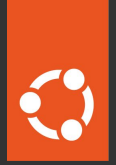


Managing Vulnerabilities through SSDLC

Luci Stanescu

9 April 2025

Can you imagine a world without cybersecurity threats?

Photo by Fausto García-Menéndez on Unsplash

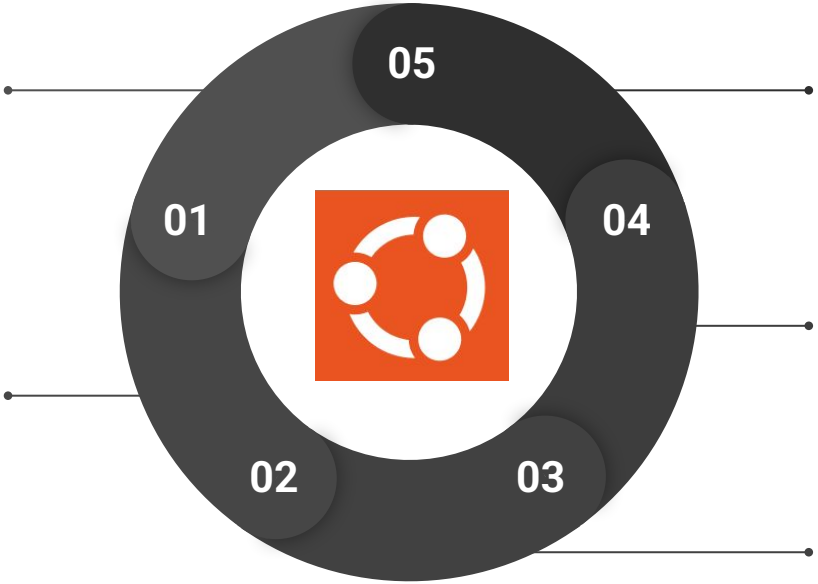
The process

Threat Modelling

Homegrown methodology based
on best practices
Focus on CIA

Static Code Analysis

Code quality, metrics and insights
Embedded in CI/CD
Improvements over time



Vulnerability Response

Analysis and prioritization
Fixing vulnerabilities prior to release

Pentests

Threat-led pentesting, using
insights from Threat Modelling

Vulnerability Scanning

Homemade scanning solution
Fixed intervals
Gather metrics for insights

Knowledge is key

> Risks can be **mitigated**, **avoided**, **transferred**, or **accepted**.
>
> It's never a good idea to **ignore** risks.

Threat modelling

Who should perform threat modelling

> Threat modelling is best done by the engineers that built the product.

An opinion by Luci Stanescu

Security Engineering Manager, Canonical

Threat modelling isn’t simple

> Canonical developed an in-house threat modelling methodology for use by projects, based on best practices and focused on CIA aspects of the product.

Threat modelling stages

1. Define the Target of Evaluation (TOE)
2. Identify assets (crown jewels and stepping stones) and data flows (entry points, exit points and trust boundaries)
3. Identify and quantify threats
4. Identify mitigating controls
5. Manage residual risk
6. Rinse and repeat

Pain points

- Out of context product
- Limiting the scope
- Assets without data flows
- Missed assets
- Critical confidentiality asset without confidentiality risks
- Missed inherent risks – existing controls need to be explicit, not implied

Static Code Analysis

Static Code Analysis

- Canonical using TIOBE
- Integrating in CI pipelines is best
- Code quality improvements happen
- Consider the risk of gaining a false sense of security

Vulnerability scanning

Vulnerability scanning: what & why

> Using an off-the-shelf vulnerability scanner, primarily for distributed artifacts.
>
> Useful for finding known vulnerabilities in vendored dependencies or complex artifacts (e.g. OCI images).
>
> Not useful for finding new vulnerabilities (duh!).

Vulnerability scanning lessons

- Not all scanners are created equal – try multiple options on known (possibly synthetic) vulnerable targets.
- Figuring out the frequency is tricky – release time is best.
- Rescan the same artifact – daily is possible with automation!
- Define a process for handling positives (whether true or false).
- Need to manage false positives – automation is necessary for frequent scans.

Penetration tests

Penetration tests

- Threat-Led Penetration Tests (TLPT) leverages insights from threat modelling to help define the scope of the activity.
- Performed after vulnerability scanning to ensure focus on architectural / logic issues.
- TLPT mandated by DORA (EU financial sector).

Vulnerability response

Panic doesn’t help.

A framework helps ensure consistent handling of security issues.

Things that don’t come naturally

Defining product / release lifetime

Scoring and prioritization

Considering mitigations

Announcements

SECURITY.md ❤️

Tell people how to disclose security issues.

Document your vulnerability response process.

If on GitHub, use the repository security advisories feature.

Security documentation

Everybody advocates for documentation.

Until they have to write it themselves.

Security documentation framework

- Define your audience
- Discuss risks
- Discuss information security (loss, incorrect retention, unlawful disclosure, etc.)
- Cover security-sensitive functions (e.g. authentication)
- Cover use of cryptography
- Include how-to guides

Cryptography topics

How it's used

Overview of what crypto tech the project uses for what

Why it's used

The protected data (to understand implications)

What is used

What libraries, low-level or high level primitives

What is exposed

How can users influence crypto functions

Security how-to guides

- Pre-deployment / deployment security checklist
- Managing authentication and authorization
- Backups
- Hardening guide

- https://maas.io/docs/how-to-enhance-maas-security
- https://documentation.ubuntu.com/juju/latest/user/howto/manage-your-deployment/harden-your-deployment/

Thank you! Questions?