

# Under the microscope: Linux security tools

Lessons learned from 500+ projects

**Michael Boelen**

michael.boelen@cisofy.com

NLLGG, September 2018



# Michael Boelen

- **Open Source**
  - Lynis, Rootkit Hunter
- **Business**
  - Founder of [CISOfy](#)
- **Other**
  - Blogger at [Linux-Audit.com](#)
  - Board member NLUUG



# The LSE project

# Project: LSE

## LinuxSecurity.Expert

- Library
- People
- Toolkit

# Library

- Checklists →
- Guides
- Configuration
  - sysctl
  - systemd
  - SSH



# People

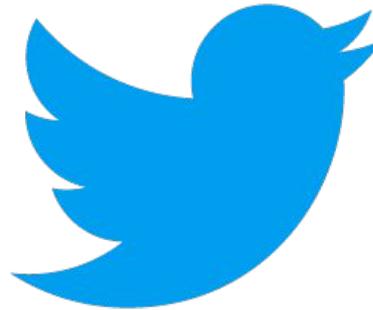
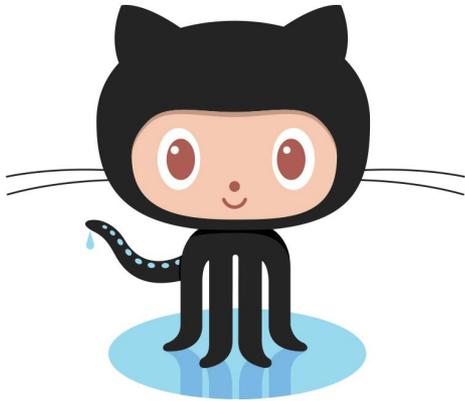
## Profiles

- Specialists in our field
  - Person behind a tool
  - Interviews

# Toolkit

- Tools
- Categories
- Snippets

# Tools - Discovery



# Tools - Discovery

## Criteria

- Open source
- Security
- Runs on Linux, macOS, BSD

# Tool analysis



# Tool analysis

## Basics

Project description

Tool category

Typical user

License

Author

Language

Keywords

Latest release

## Quality

Changelog

Popularity

Documentation

Code

Releases

## Usage

Installation

Ease of use

# Tool analysis

CISOfy / lynis

Unwatch 295 Unstar 4,997 Fork 586

Code Issues 15 Pull requests 2 Projects 7 Wiki Insights Settings

Lynis - Security auditing tool for Linux, macOS, and UNIX-based systems. Assists with compliance testing (HIPAA/ISO27001/PCI DSS) and system hardening. Agentless, and installation optional. <https://cisofy.com/lynis/> Edit

shell linux pci-dss compliance security-audit security-hardening security-scanner security-vulnerability hipaa unix vulnerability-detection vulnerability-scanners vulnerability-assessment devops devops-tools system-hardening hardening auditing gdpr security-tools Manage topics

2,158 commits 1 branch 41 releases 100 contributors GPL-3.0

Branch: master New pull request Create new file Upload files Find file Clone or download

superpoussin22 and mboelen detect if latest TAG is used (#575) Latest commit 9fe6dcd 6 days ago

**Output**

# Tool review

- Introduction
- Typical tool usage
- How it works
- Background details
- Strengths and weaknesses
- Example output
- Author information
- Tool alternatives
- Categories
- Tags
- And more...

# Tool review

LSE top 10 **Lynis (2)**

## Tool and Usage

Project details	
Inception	2007
License	<a href="#">GPLv3</a>
Programming language	shell script
Author	Michael Boelen
Latest release	<a href="#">2.6.8</a> [2018-08-23]

## Project health



This score is calculated by different factors, like project age, last release date, etc.

# Top 100: security tools

Secure | <https://linuxsecurity.expert/security-tools/top-100/>

## Tools by ranking

### 1. Frida (reverse engineering tool) 15 ▲

black-box testing, reverse engineering

Frida allows developers and researchers to inject custom scripts into black box processes. This way it can provide a hook into any function, allowing to trace executed instructions. The source code is not needed. Frida even allows direct manipulation and see the results. The tool comes with bindings for different programming languages, allowing to interact with processes. Example of the bindings that Frida provides include Python, Swift, .NET, Qt/Qml, and C API.

Black box   Dynamic analysis   Reverse engineering

### 2. Bro (network security monitoring tool) 27 ▲

security monitoring

Bro helps to perform security monitoring by looking into the network's activity. It can find suspicious data streams. Based on the data, it alert, react, and integrate with other tools.

IDS   Intrusion Detection   Network security monitoring   NIDS   NSM

### 3. Faraday (collaboration tool for penetration testing) 15 ▲

collaboration, penetration testing, security assessment, vulnerability scanning

Faraday helps teams to collaborate when working on penetration tests or vulnerability management. It stores related security information in one place, which can be easily tracked and tested by other colleagues.

Collaboration   Pentesting   Security audit

# Tools by category



# Lessons learned

# Lessons learned - Basics

- Not really open source!
- Unclear goal
- Authorship
- Versioning
- Changelog missing

# Lessons learned - Documentation

- Missing a basic description
- No 'get started' guide
- Lack of good examples

# Lessons learned - Ease of use

- Complicated installation
- No sane defaults (e.g. --help missing)
- Parameters make no sense

# What questions do you have?

## Get connected

- Twitter ([@mboelen](#) and [@LSELabs](#))
- LinkedIn ([Michael Boelen](#))

# More?

## Related articles at **linux-audit.com**

- [Why we use your open source project \(or not\)](#)
- [How to Promote your Open Source Project](#)





# Best Practices

--full-throttle-engine, -f  
--help, -h, or help  
--version, -V

Learn more: [docopt.org](https://www.docopt.org)

```
$ ./lynis show help
Lynis 2.4.1 - Help
=====

Commands:
audit
show
update
upload-only

Use 'lynis show help <command>' to see details

Options:
--auditor
--check-all (-c)
--config
--cronjob (--cron)
--debug
--developer
--help (-h)
--license-key
--log-file
--manpage (--man)
--no-colors --no-log
--pentest
--profile
--plugins-dir
--quiet (-q)
--quick (-Q)
```

# Best Practices

## Keep a changelog

- History
- Trust
- Troubleshooting

# Best Practices

## Semantic versioning!

Major.Minor.Patch

Learn more: [semver.org](https://semver.org)

# Credits

## Images

Where possible the origin of the used images are included in the slides. Some came without an origin from social media and therefore have no source. If you are the owner, let us know and we add the source.