Open Source and Unsupported Software

George Masters, Lead Product Engineer, Secure Engineering
Schweitzer Engineering Laboratories, Inc
March 23, 2020

RELIABILITY | RESILIENCE | SECURITY

NERC
NORTH AMERICAN ELECTRIC
RELIABILITY CORPORATION

Agenda

- Introduction: Open Source and Unsupported Software
  - *What is* Open Source and Types of users
  - How Open Source and *Unsupported* are related
- Assessing Operational Risk (this includes “pilot error”)
  - Threat Modeling
- Evaluating Trustworthiness
  - Authenticity and Reputation
- Case Studies
- The Problem of Unsupported Software
- Conclusion
- Q&A

NERC
NORTH AMERICAN ELECTRIC RELIABILITY CORPORATION

Introduction

“Open source software is software with source code that anyone can inspect, modify, and enhance.”

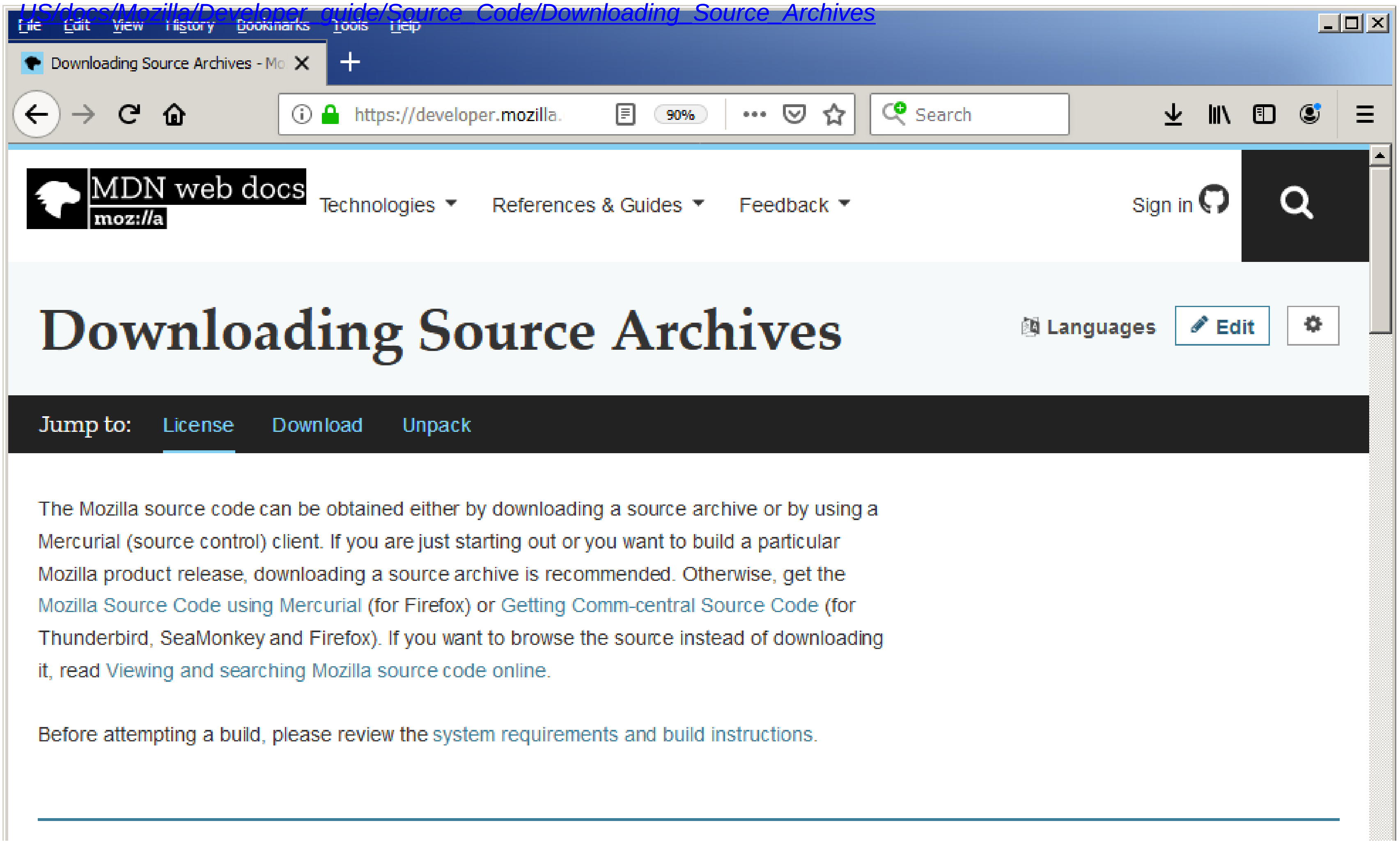
For users, no contractual relationship with a supplier –> *Self-Service Support*

- Mozilla Firefox
- Google Chrome
- Linux
- Git

Reference: https://opensource.com/resources/what-open-source

Example: Mozilla Firefox

https://developer.mozilla.org/en-US/docs/Mozilla/Developer_guide/Source_Code/Downloading_Source_Archives


File Edit View History Bookmarks Tools Help
Downloading Source Archives - Mo
https://developer.mozilla.
90%
Search
MDN web docs
moz://a
Technologies
References & Guides
Feedback
Sign in
Languages
Edit

Downloading Source Archives

Jump to: License Download Unpack

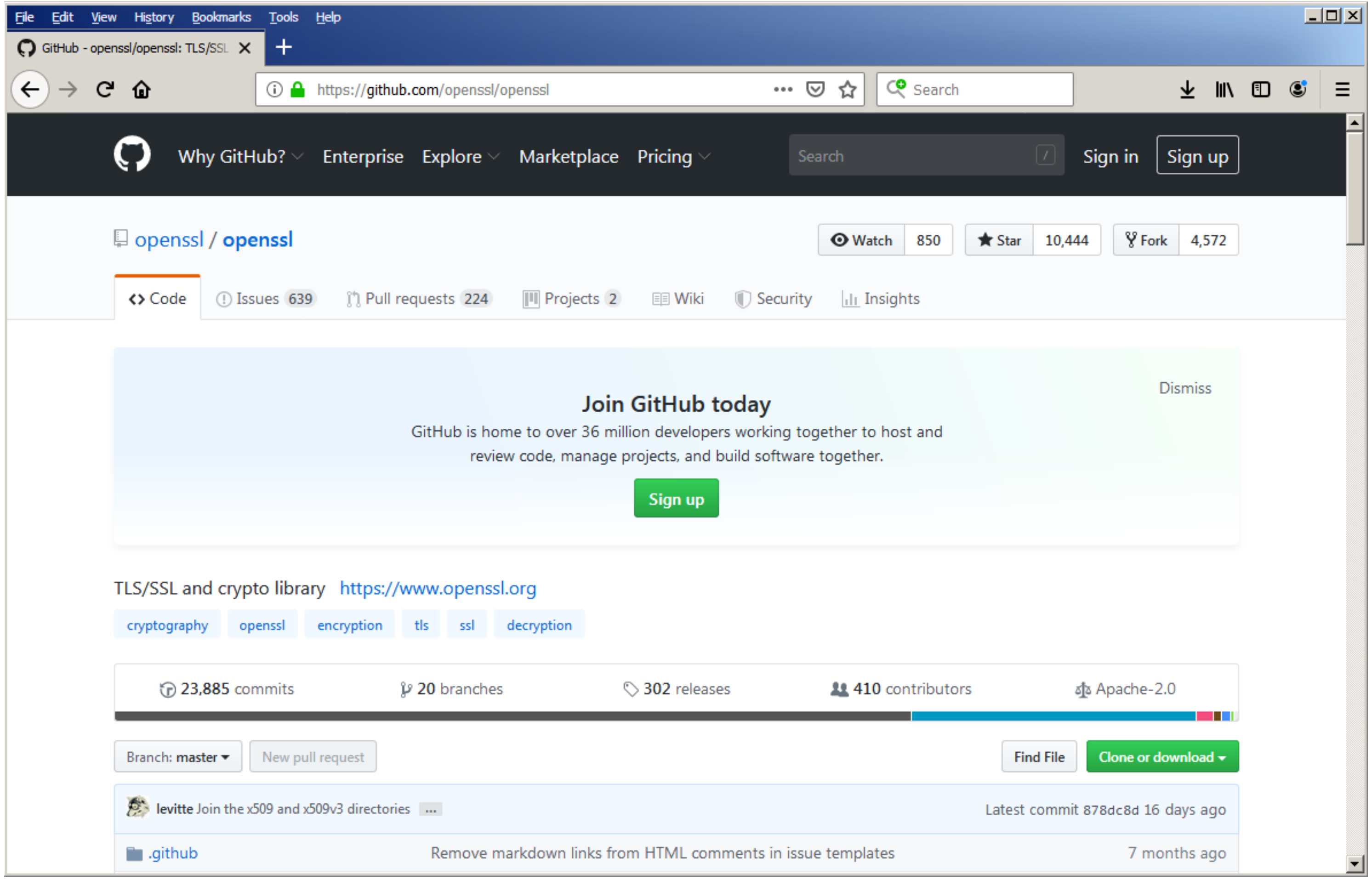
The Mozilla source code can be obtained either by downloading a source archive or by using a Mercurial (source control) client. If you are just starting out or you want to build a particular Mozilla product release, downloading a source archive is recommended. Otherwise, get the Mozilla Source Code using Mercurial (for Firefox) or Getting Comm-central Source Code (for Thunderbird, SeaMonkey and Firefox). If you want to browse the source instead of downloading it, read Viewing and searching Mozilla source code online.

Before attempting a build, please review the system requirements and build instructions.

NERC NORTH AMERICAN ELECTRIC RELIABILITY CORPORATION

Example: OpenSSL

https://github.com/openssl/openssl


File Edit View History Bookmarks Tools Help

GitHub - openssl/openssl: TLS/SSL

https://github.com/openssl/openssl

Why GitHub? Enterprise Explore Marketplace Pricing Search Sign in Sign up

openssl / openssl

Watch 850 Star 10,444 Fork 4,572

Code Issues 639 Pull requests 224 Projects 2 Wiki Security Insights

Join GitHub today

GitHub is home to over 36 million developers working together to host and review code, manage projects, and build software together.

Sign up Dismiss

TLS/SSL and crypto library https://www.openssl.org

cryptography openssl encryption tls ssl decryption

23,885 commits 20 branches 302 releases 410 contributors Apache-2.0

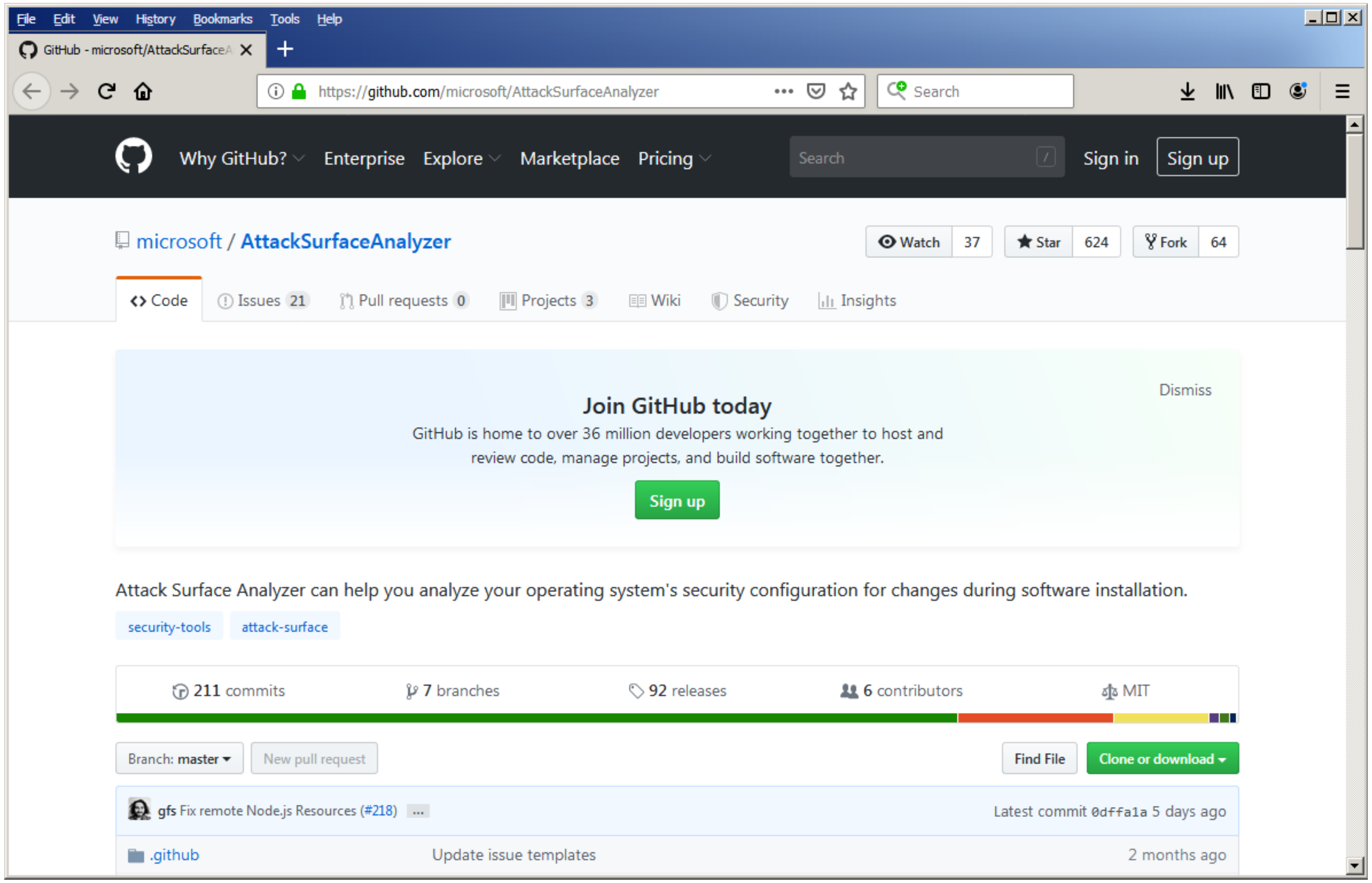
Branch: master New pull request Find File Clone or download

levitte Join the x509 and x509v3 directories … Latest commit 878dc8d 16 days ago

.github Remove markdown links from HTML comments in issue templates 7 months ago

Some Open-Source Has a Sponsor

https://github.com/microsoft/AttackSurfaceAnalyzer (Sponsored Open Source)


File Edit View History Bookmarks Tools Help
GitHub - microsoft/AttackSurfaceA
https://github.com/microsoft/AttackSurfaceAnalyzer
Search
Why GitHub? Enterprise Explore Marketplace Pricing
Sign in Sign up
Watch 37 Star 624 Fork 64
Code Issues 21 Pull requests 0 Projects 3 Wiki Security Insights
Dismiss

microsoft / AttackSurfaceAnalyzer

Join GitHub today

GitHub is home to over 36 million developers working together to host and review code, manage projects, and build software together.

Sign up

Attack Surface Analyzer can help you analyze your operating system's security configuration for changes during software installation.

security-tools attack-surface

211 commits | 7 branches | 92 releases | 6 contributors | MIT

Branch: master | New pull request | Find File | Clone or download

gfs Fix remote Node.js Resources (#218) … Latest commit 0dffa1a 5 days ago

.github | Update issue templates | 2 months ago

The Skills Required and Availability of Documentation and Support Varies Widely

- User types
  - Downloader
  - Builder
  - Developer
- Things Users are responsible for
  - Installation
  - Configuration
  - Security maintenance
  - Watching out for dependencies and install-alongs

Assess Operational Risk

"… identify and assess cyber security risk(s) to the Bulk Electric System from vendor products or services resulting from: (i) procuring and installing vendor equipment and software; and (ii) transitions from one vendor(s) to another vendor(s)"

- R1.1 risk assessment revolves around procuring and installing
- Procurement is a pretty broad term that includes selection
- A *threat model* is a good way to describe operational risk for a process, including how it might go wrong, the likelihood and impact of those possibilities, and what you plan to do about it.

NERC
NORTH AMERICAN ELECTRIC
RELIABILITY CORPORATION

Threat Modeling

Expanding the threat model idea a bit...

It describes the **things that affect the system**, the **things the system can affect**, and the **criticality** of those things. It also describes **what can go wrong**, either accidentally or deliberately induced by an attacker, how likely those are, and ways threats can be reduced.

A threat model evaluates each threat for importance and prioritizes mitigation. It concludes with residual risks that are accepted.

https://en.wikipedia.org/wiki/Threat_model

NERC
NORTH AMERICAN ELECTRIC
RELIABILITY CORPORATION

Risk Factors

Some things to look for:

- *Is this software authentic and unmodified?*
- *What are the risks?*
  - Internet connection (how can an attacker get access?)
  - Operating as privileged user (administrator, Local System, etc.)
  - What are the inputs and how trustworthy are they?
    - Inauthentic? Malicious?
  - What are the output and what happens if inappropriate?
  - What can this system reach via the network?
  - **How trustworthy is the software?**

NERC
NORTH AMERICAN ELECTRIC
RELIABILITY CORPORATION

Trustworthiness of Software

Evaluating trustworthiness of software is not unique to Open Source, but the problem is especially acute, because usually no contractual relationship exists with a supplier - you’re on your own, so you look for evidence.

Things to look for:

- Visible community activity
- Visible defect and commit (and revision) history
- Security defects being identified
- Security defects being handled quickly
- Possibility of a sponsoring organization
- Reputation and Internet chatter

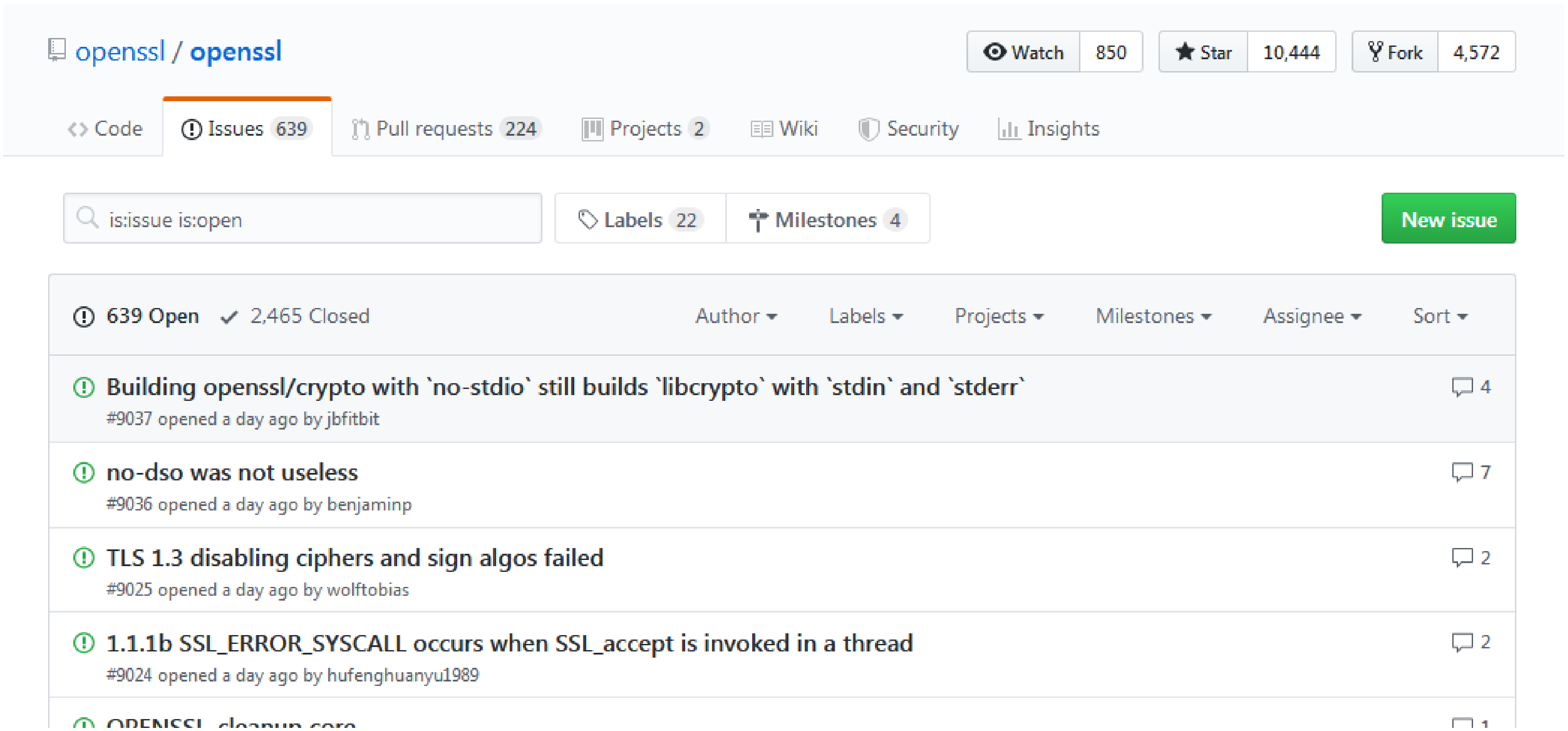
NERC
NORTH AMERICAN ELECTRIC RELIABILITY CORPORATION

Sources of Evidence

- Internet Searches
  - “chatter”
  - User groups
  - Project Web page
    - Release distribution
    - Bugs and Issues
    - Information mailing lists
- https://nvd.nist.gov/vuln/search?execution=e2s1
- https://scan.coverity.com/projects
- https://www.openhub.net

GitHub again

GitHub provides a convenient view into issues and activity



openssl / openssl

Watch 850 | Star 10,444 | Fork 4,572

Code | Issues 639 | Pull requests 224 | Projects 2 | Wiki | Security | Insights

is:issue is:open | Labels 22 | Milestones 4 | New issue

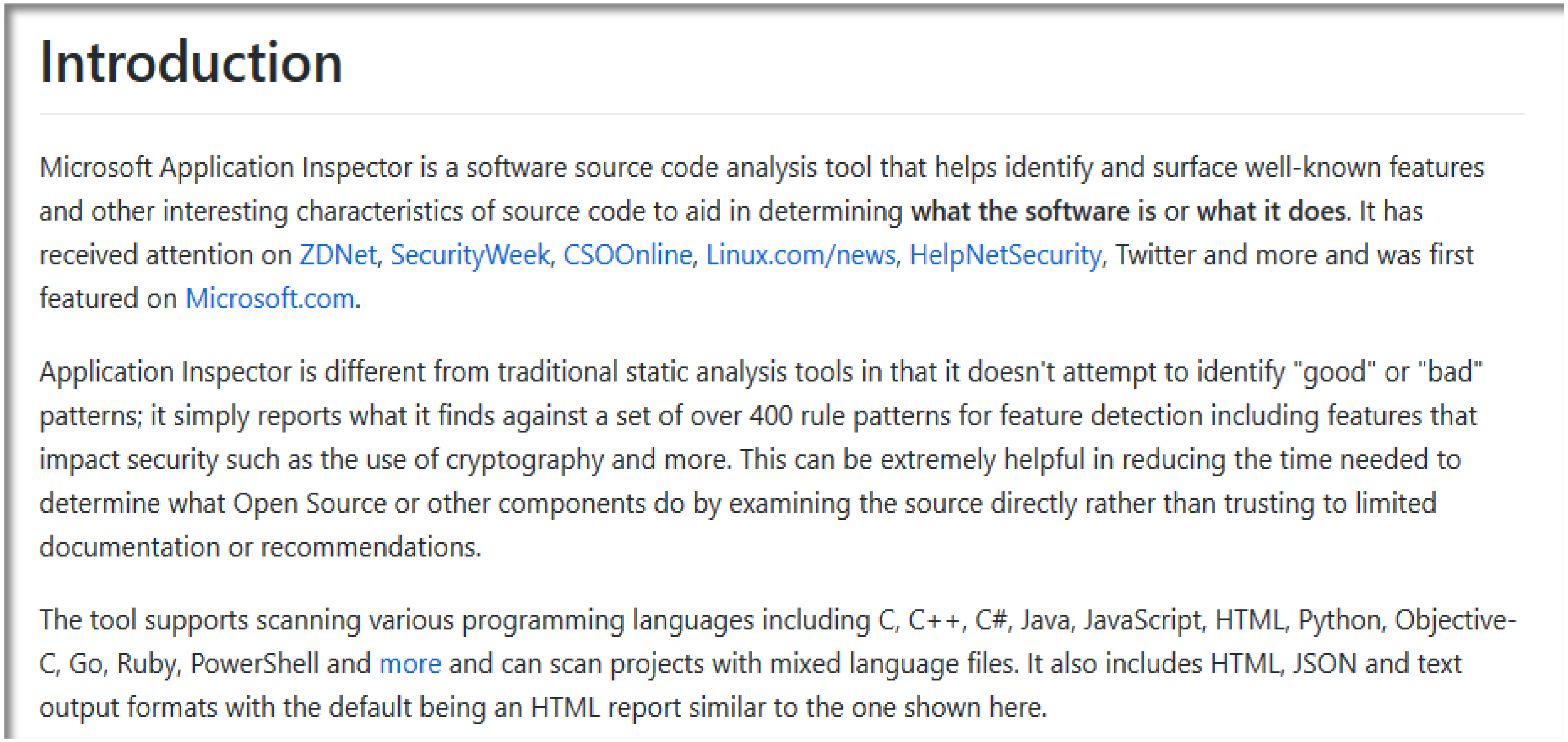
639 Open ✓ 2,465 Closed — Author, Labels, Projects, Milestones, Assignee, Sort

- Building openssl/crypto with `no-stdio` still builds `libcrypto` with `stdin` and `stderr` — 4
  #9037 opened a day ago by jbfitbit
- no-dso was not useless — 7
  #9036 opened a day ago by benjaminp
- TLS 1.3 disabling ciphers and sign algos failed — 2
  #9025 opened a day ago by wolftobias
- 1.1.1b SSL_ERROR_SYSCALL occurs when SSL_accept is invoked in a thread — 2
  #9024 opened a day ago by hufenghuanyu1989
- OPENSSL cleanup core — 1

NERC
NORTH AMERICAN ELECTRIC
RELIABILITY CORPORATION

NEW: Microsoft Application Inspector

- https://github.com/Microsoft/ApplicationInspector



Introduction

Microsoft Application Inspector is a software source code analysis tool that helps identify and surface well-known features and other interesting characteristics of source code to aid in determining **what the software is** or **what it does**. It has received attention on ZDNet, SecurityWeek, CSOOnline, Linux.com/news, HelpNetSecurity, Twitter and more and was first featured on Microsoft.com.

Application Inspector is different from traditional static analysis tools in that it doesn't attempt to identify "good" or "bad" patterns; it simply reports what it finds against a set of over 400 rule patterns for feature detection including features that impact security such as the use of cryptography and more. This can be extremely helpful in reducing the time needed to determine what Open Source or other components do by examining the source directly rather than trusting to limited documentation or recommendations.

The tool supports scanning various programming languages including C, C++, C#, Java, JavaScript, HTML, Python, Objective-C, Go, Ruby, PowerShell and more and can scan projects with mixed language files. It also includes HTML, JSON and text output formats with the default being an HTML report similar to the one shown here.

Case Studies

Havex

- Compromised point of distribution
- Countermeasure: Digital Signatures or Hash Values

Heartbleed

- Severe defect compromising confidential data
- In the wild from March 2012 to April 2014
- Fixed about 4 days after disclosure to the OpenSSL maintainers
- Countermeasure: Monitor for security announcements

Unsupported Software

There is no support available

- No developer community, commercial or otherwise
- If security vulnerabilities are found, they will not be patched
- … and you may never know, *but attackers might*

The security risk is UNKNOWN

If you must use Unsupported Software, mitigate risks

- Reduce Attack Surface
- Control Access
- Isolate
- Monitor

The Takeaways

- Make sure the distribution point is *authentic*
- Make sure the software is *currently supported*
- Know *where security notices and updates will appear* and monitor for them
- Always apply what *mitigations* you can
  - Reduce attack surface (turn off unused stuff, disable phone-home, etc.)
  - Control access (limit use to qualified users with need)
  - Isolate (isolated network, bastion host, air-gap)
  - Monitor (host computer and network logs)
- *An unsupported product is unknown risk*

Additional resources related to Supply Chain Security can be found at https://www.nerc.com/pa/comp/Pages/Supply-Chain-Risk-Mitigation-Program.aspx.

Questions and Answers