FOSSA

CUSTOMER SPOTLIGHT: SmartThings

How SmartThings runs IoT open source compliance across dozens of releases per day

Featuring: Dean Hemstreet, Vice President Q/A & Release Engineering

Watch the interview at http://fossa.io/

SmartThings is building an open platform for the Internet of Things that hundreds of thousands of consumers across the world use to connect and secure their homes. Their flagship product, "The Hub", serves as the brain for the connected home, allowing smart devices like the Amazon Echo, Phillips Hue, Ring Doorbell and more to interact and bring the average home into a delightful, connected experience.

To power this platform, SmartThings runs a blazingly fast release process. From the ground-up, the team automates everything that touches code as it goes out the door. Their CI/CD mentality supercharges their tools, workflow and release velocity, allowing them to ship dozens of times per day:

> "We release code **dozens of times** a day. As a shop, one of our first things we did was bring in CI/CD, part of that was to automate everything we do."

SmartThings's release velocity is especially impressive given the complexity of their development process. Among mobile and web apps, SmartThings ships firmware which implements over a dozen programming languages ranging from traditional (i.e. C/Java) to the cutting edge (i.e. Rust/Swift). Much of their code passes through a sophisticated build pipeline and code review workflow. This process brings in thousands of open source libraries at different points during the build, each with unique license obligations. A solution to track these libraries would not only have to support this complexity, but also understand how multiple build targets, configuration or release branches could affect build behavior, license obligations and what ends up in their code.

The Challenge

SmartThings needed to integrate a solution quickly and efficiently to help scan and comply with open source licenses. The right solution had to be able to manage the large complexity of SmartThings's development workflow without sacrificing speed or getting in the way.

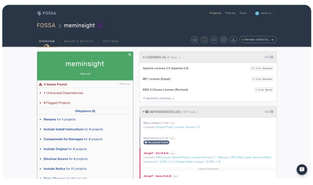
> "Slowing down was not an option. We are simply moving too fast."

Enter FOSSA

> "FOSSA was a huge relief… the big surprise was how easy it was!"

Watch the video interview (2:26)

In an environment where slowing down wasn't an option, FOSSA was brought in to get a working process within days that covered every part of development without getting in the way.



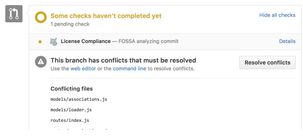
Turning compliance on autopilot with deep integrations

The key to FOSSA's success was its ability to integrate deeply into a sophisticated workflow, becoming involved at every point as code passes through the release process.

> "FOSSA is integral to our workflow… we have an integration there with the build process, CI/CD, all our languages in the stack … and tools like JIRA, Github, Slack, email… With this, FOSSA helps us find issues in realtime."

From the outset, FOSSA's GitHub integration meant that onboarding FOSSA only took a few clicks before all of SmartThings products were automatically set up with license scanning, dependency analysis and compliance alerts at each commit. To accurately understand complicated codebases, FOSSA pre-analyzed and configured itself against the code, picking up dozens of languages that were mixed together within and across codebases.

Beyond monitoring, FOSSA configured itself to embed proactive and realtime compliance in the SmartThings engineering workflow. SmartThings developers work on alternate copies of the codebase (branches) and open code reviews (via GitHub Pull Requests) to get changes into their products. Out of the box, FOSSA supports multiple branches, release channels and adding automated checks to code review, allowing SmartThings to preview how new changes will affect their code before merging in. This ensures compliance issues are prevented at the earliest stages, with no expensive rework.

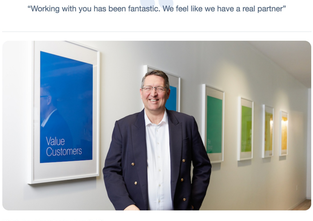


Out of the box, FOSSA set up a suite of custom alerts and policies based off baked-in legal knowledge and industry best practices for compliance in IoT, speedily kicking off an alerting infrastructure in SmartThings. If any issues land into core code without review, the FOSSA Slack and email integrations notifies the right stakeholders. Since firmware often requires merging the work of multiple repos and teams, this allows company-wide communication of compliance status and redundant checks as code gets merged. Managers can transparently see via status badges integrated into GitHub, JIRA and Confluence on-demand, with actionables efficiently escalated to the most relevant team members.

When it comes time to release, the benefits of FOSSA's integration and deep scanning allowed releases to continue without a hitch. Due to the battery of checks code have to pass through, code staged for release is already clean – allowing SmartThings to retain "anytime" flexibility in their release schedule. Instead of manually assembling reports and attribution, FOSSA automatically collects raw license headers and assembles reports per-commit, so every release is guaranteed to have the most recent documentation and disclosures. Finally, because FOSSA keeps a full history, SmartThings can compare how things changed across releases.

This whole automated process along with a full on-prem install took only a few days. Compliance is now as simple as adding a "FOSSA check" as code goes out the door.

> "Working with you has been fantastic. We feel like we have a real partner"



Watch the video interview here (2:13)

SmartThings, Inc.

SmartThings Inc. is a technology company building an open platform for smart homes and the consumer Internet of Things. SmartThings makes a hub (sometimes called "gateway" or "home controller"), cloud platform, and client applications.

Employees

Over 400

Industry

Hardware, Internet of Things (IoT)

Headquarters

Mountain View, California

Deployment

FOSSA On-prem in AWS private cloud

Integrations

GitHub (hosting/pull requests), Jenkins, Circle-CI, Slack, Email

Languages

Java (maven/gradle), Rust, iOS (cocoapods/carthage), NodeJS, C, etc...