Alpha-Omega

QUARTERLY REPORT
Q3 2025

Where we are

TODAY

Q3 2025

SUMMARY

In Q3 2025, Alpha-Omega grantees made significant strides in security, infrastructure, and policy development. The Apache Software Foundation progressed on the Apache Trusted Release pipeline, now in internal alpha testing with six PMCs, while planning improvements to tooling and repository upgrades. The Eclipse Foundation addressed a vulnerability in Open VSX, improved Otterdog tooling, and hosted well-attended security training sessions. FreeBSD advanced preparations for its 15.0 release, focusing on migrating to OpenSSL 3.5 with key integration milestones achieved. Node.js enhanced its security and permission model, fixing critical vulnerabilities and expanding threat modeling, alongside community outreach and improved release processes. The Python Software Foundation focused on vulnerability triage, phishing response, account security, and delivered multiple community keynotes.

RubyCentral completed foundational work enabling prebuilt binary gem support in RubyGems, onboarded AWS as the first Organization on RubyGems.org, and advanced policy and security infrastructure planning, including SSO. The Rust Foundation launched Trusted Publishing in production on crates.io, enhanced its Capslock security system, progressed vulnerability surfacing plans, reached consensus on TUF implementation, transitioned key tooling to standalone libraries, and became an official CVE Numbering Authority.

Additionally, in June at Open Source Summit North America, we successfully hosted an in-person Roundtable and the first in-person Open Source Corps of Security Engineers meeting, fostering valuable collaboration across projects.

NEW ENGAGEMENTS

In Q3, we continued to invest in automated analysis tooling. This work builds on Google's Capslock project (for Golang) and the previously funded work in Rust to make automated call graph and reachability analysis available in Java. The new work will open source an existing Java callgraph tool (courtesy of OpenRefactory) and create annotations for Java system calls. We have also funded new work that uses these callgraphs to reduce toil on the creation and management of VEX statements for open source projects. The new work will use AI to do root-cause analysis of vulnerabilities and then reachability analysis to help maintainers understand if their code is affected by a particular vulnerability. This work is being done in partnership with the Apache Solr project.

We also kicked off a major audit program with the Open Source Technology Improvements Fund (OSTIF) that will cover PyTorch, LLVM, LangChain, mbedTLS, Requests, CacheControl, and urllib3.

This program will also include “rapid security reviews” for over 20 projects. A Rapid Security Review is a focused, time-boxed assessment conducted over a short amount of time: the aim is for reviewer and project to each spend less than a couple of hours on the entire process. Each review aims to quickly identify and prioritize key security improvements within the projects, providing actionable guidance without overwhelming maintainers. The rapid security review concept was developed by the Eclipse Foundation security team.

The rapid security review project scope includes the following projects: pyYAML, pycparser + python-cffi, cachetools, Soupsieve, NumPy, Monolog (PHP), PHP-Parser, LZ4, libpng, Pandas, Postgresql, Sqlite, fsnotify, snakeyaml, Grafana, Jenkins, memcached, fluent-bit, MariaDB, Struts2, and as many Apache Commons projects as the budget allows.

Continuing our engagements with the Linux Kernel we funded two initiatives with eBPF: an audit of the eBPF Verifier & JIT Compilers and Enabling Sanitizers for JIT-ed eBPF Programs.

EXISTING ENGAGEMENTS

Our existing engagements provide monthly updates to us through our public GitHub repository:

- Apache Software Foundation
- Eclipse Foundation
- Ecosyste.ms
- Linux Kernel
- OpenJS / Node.js
- Ruby Central
- Rust Foundation
- Python Software Foundation
- PyCA
- OSTIF

A FEW NOTABLE WORK UPDATES FROM GRANTEES

Apache Software Foundation

The Apache Trusted Release (ATR) platform development is focused on creating a new release pipeline that strictly adheres to the Apache Release Policy. Since February, our efforts have concentrated on stabilizing this core functionality. Once stable, we plan to implement additional features before launching a working Beta later this year.

Progress and Next Steps

Alpha Testing: ASF initiated an internal Alpha Test of the new pipeline with six Project Management Committees (PMCs): Airflow, Commons, Logging, Maven, Pekko, and Tomcat. The initial focus is on correctly implementing the ASF policy gates within the pipeline, and they've received positive initial feedback.

The Apache Software Foundation Cont.

- Project Identification: ASF is refining tooling to encourage communities to consistently self-identify all their projects and products.
- Release Information: ASF is reviewing preferred locations and file formats for publicly providing project release information. This historically involved optional DOAP files, often in varied URL patterns, which ASF aims to standardize.
- Repository Upgrade: ASF is planning an upgrade of the current Nexus 2 repository (https://repository.apache.org/) to Nexus 3 to support modern workflows and expand distribution channel options.
- Community Lifecycle Tooling: ASF is collaborating with the ASF Board to develop tooling that will efficiently guide communities, their projects, and releases through their lifecycle management.

Eclipse Foundation

The OpenVSX package registry had a vulnerability in the extension publishing process reported on May 4. After confirmation, a fix was deployed on June 24, followed by a security advisory. No compromise was found, but 81 extensions were deactivated as a precaution. Recommendations have been issued to mitigate future risks.

Otterdog Improvements include better string handling, support for archived organizations, a new check-status CLI command, and a fix for code scanning settings. A new release is coming soon. The contributor guide has been simplified with updated tooling using Skaffold, Minikube, and optional Tailscale.

Security Training was held on June 3rd & 10th with 137 attendees. Topics included vulnerability management and SBOM. Recordings are available online and shared with the Alpha-Omega community, though funded by the Sovereign Tech Agency.

FreeBSD

Preparations for FreeBSD 15.0, scheduled for release in December 2025, are in full swing with the feature freeze set for August 8. A major blocker is the migration to OpenSSL 3.5, a critical priority due to security and support timelines—OpenSSL 3.0 reaches end-of-life in 2026, while 3.5 is supported through 2030, aligning with FreeBSD 15's lifecycle. Pierre Pronchery is leading the integration effort and made significant progress in July, including upgrading to OpenSSL 3.5.1, enabling builds across all supported architectures, generating manual pages, and preparing the branch for testing and review (Phabricator D51613). The merge is targeted for completion before the feature cut-off. Meanwhile, the FreeBSD Foundation is planning next steps beyond the OpenSSL integration.

OpenJS / Node.js

In Q3, the Node.js project focused on security and permission model improvements. Key enhancements included better debugging tools, updates to the process permission model, and fixes for edge cases. Security releases addressed two high-severity vulnerabilities: a path traversal issue on Windows (CVE-2025-27210) and a HashDoS vulnerability in V8 (CVE-2025-27209), with coordinated fixes across all active release lines. Security release automation was also improved, streamlining CVE requests and communications. The threat model was expanded with better documentation and clarification on path-related risks. Seven reports were evaluated through HackerOne. Broader JavaScript ecosystem efforts included launching a security-help channel, project evaluations by Alpha Omega, and ongoing work on a public Express.js bug bounty, improved release scheduling, and updated documentation, including a new Secure Releases Guide and EOL page.

Python Software Foundation

In Q3, Seth Larson focused on triaging and patching Python vulnerabilities, publishing CVE-2025-8194 for the tarfile module and updating PyPI and Truststore. He contributed to discussions on sustainable security, delivered multiple keynotes (including Open Source Security work isn't "Special"), and submitted revisions for an NSF grant. He also onboarded the new Python Release Manager and finalized a case study on SBOMs.

Mike Fiedler handled a surge in malware reports on PyPI, responded to a phishing attack, and improved account safety by blocking takeovers via expired domains. He resolved infrastructure issues, improved developer tooling, and reduced API errors. Mike also spoke at AWS Summit NY and several open source events, highlighting PyPI security challenges and best practices.

Ruby Central

Samuel Giddins made major progress enabling support for prebuilt binary gems ("wheels") in RubyGems, completing a complex integration that required overhauling how platforms are represented and matched. This foundational work will later extend to Bundler. He also co-presented a keynote at RailsConf on the history and evolution of rubygems.org and supported efforts to add SHA-based pinning in Gemstash.

Marty Haught oversaw the production onboarding of AWS as the first RubyGems.org Organization, addressed early feedback, and began planning the transfer of hundreds of gems into the new structure. The team also held a successful offsite during RailsConf to align on new policies for gem deletion, ownership, and inactive maintainers. Progress continued on infrastructure security, with planning for Single Sign-On (SSO) implementation led by Colby, and first-phase development underway.

The Security Working Group began organizing its first meeting, though engagement remains low. Marty also participated in Open Source Summit North America and RailsConf, gaining valuable insights around open source sustainability and financial support models for RubyGems. Feedback from international events sparked ideas for increasing Ruby Central's visibility and launching a business membership model.

Rust Foundation

In Q3, the Rust project made significant progress in strengthening its security and infrastructure. Trusted Publishing is now fully live on crates.io, providing secure crate publishing with integrated CI/CD support, secret scanning, and incident response capabilities. The Capslock capability-based security system was enhanced to support robust runtime scanning and efficient monitoring of crate behavior using sandboxing tools. Planning for vulnerability surfacing is underway, focusing on integrating RustSec advisories into crates.io, with UI development and community feedback processes expected to begin soon. The project reached consensus on implementing the TAP-16 Merkle Tree approach for TUF, simplifying quorum management and targeting a minimum viable product by early August. Additionally, the Painter analysis engine is being transitioned into a standalone library to support further tooling development. The Ferrocene Language Specification has been fully incorporated into the Rust Project, and Rust officially became a CVE Numbering Authority hosted by Red Hat. The forthcoming Rust Foundation Technology Report will highlight these advancements and ongoing efforts to enhance the security and resilience of the Rust package ecosystem.

Q3 AO BLOGS

Trusted Publishing: Secure Rust Package Deployment Without Secrets
Alpha-Omega Endorses the Joint Statement on Sustainable Stewardship
OSTIF 2025 Alpha-Omega Partnership Updates and Roadmap
Member Spotlight: AWS – Funding Open Source Security Empowerment
Unmasking Phantom Dependencies with Software Bill-of-Materials as Ecosystem-Neutral Metadata | White Paper by: Seth Larson – Python Software Foundation

OKRs

O1: Catalyze trustworthy and secure software, runtimes, and infrastructure for all the major open source ecosystems through staffing
KR 1.1: Fund security improvements and initiatives for at least ten critical open source organizations by the end of 2025.
KR 1.2: For each engagement, confirm progress toward improved security outcomes, evidenced through initial and/or follow-on assessments, monthly reporting, and periodic check-ins.
KR 1.3: Drive the organizations we work with to obtain security funding from at least one organization other than Alpha-Omega, targeting 33% by the end of 2024.
KR 1.4: Organize quarterly roundtables for at least 5 major ecosystems to share information, build connections, and collaborate, resulting in at least one new project or joint publication started in 2024.
KR 1.5: Scaling adoption, consumption, value of OSS Security projects, Getting to sustainability tipping points.
O2: The top 10,000 open source projects are free of critical security vulnerabilities
KR 2.1: Drive adoption of key security processes, including static analysis, credential scanning, the use of private vulnerability disclosures, structured metadata (Security Insights) and the use of multi-factor authentication by maintainers of 500 critical projects from the top 10,000 by the end of 2024.
KR 2.2: Independently scan, triage, and notify maintainers when critical vulnerabilities are found in 2,000 projects, chosen from the top 10,000 by the end of June 2024, with emphasis on clearing a "section of the beach" by focusing on the top PyPI packages.
KR 2.3: Publish in a machine readable format the attestations for all packages from 2.2 that returned no vulnerabilities and those that found vulnerabilities which were subsequently fixed and verified.
KR 2.4: Open Source Data
KR 2.5: Towards beach cleaning tooling and playbooks
KR 2.6: OS Corps of Engineers

OKRs

O3: Enhance Alpha-Omega's effectiveness in innovation experimentation and marketing
KR 3.1: By the end of 2024, run three experiments to explore new strategies for reducing security risk within the open source ecosystems, share the results/learnings, using them to refine our overall strategy and objectives for 2025.
KR 3.2: More active internal marketing to stakeholders targeted at specific teams through infographics and marketing assets.
O4: Run an operationally efficient, growing, and effective program
KR 4.1: Allocate at least 85% of our yearly spend to activities directly in support of our mission.
KR 4.2: Receive at least $5 million in renewed funding in 2024.
KR 4.3: For each partner engagement, at least 70% of the objectives defined within the respective agreement are met within the defined time period.
KR 4.4: Develop and deliver quarterly reports. Increase engagement/interest across stakeholders, grant recipients, and other target orgs.
KR 4.5: Funding Partnerships 3-5 Partnerships (STF or OTF) and Corporations

COMING UP

Our next Quarterly update will be published in January 2026. Our next Alpha-Omega public meeting will take place on October 1, 2025 (all are welcome), and our next quarterly meeting will take place on Thursday November 6th. If you have not received an invite or would like to attend please reach out to info@alpha-omega.dev,

If you have any questions about this update or any of our work, please contact the Alpha-Omega team at info@alpha-omega.dev, reach out to one of us directly, or come say 'hi' on the #alpha_omega channel in the OpenSSF slack.

Bob Callaway, Google
Henri Yandell, AWS
Michael Scovetta, Microsoft
Michael Winser, Technical Strategist, Alpha-Omega