

CASE STUDY

Trend Micro Uses vFunction to Refactor Critical Monolith for AWS

4X

Faster modernization with automation over manual efforts

90%

Decrease in deployment time for critical service

Dev Experience Boost

Greatly improved dev morale and training

Executive Summary

With over 500,000 global customers and \$1.7 billion in revenues, Trend Micro is the global leader in Cloud Workload Security, [according to industry analysts](#) Forrester and IDC. The Trend Micro Cloud One™ platform provides services for hybrid cloud security, network security, user protection, and detection and response for cybersecurity threats.

Trend Micro's products were not born in the cloud—until a few years ago, many solutions were delivered on-premise for the vast majority of customers. While early re-hosting efforts to AWS were successful as far as optimizing compute resources, the monolithic nature of Trend Micro's Workload Security Runtime product suite—with a combined 2 million lines of code (LOC) and 10,000 highly-interdependent Java classes—made it difficult to achieve developer productivity, increased deployment velocity and speed, and other benefits of the cloud.

By using the vFunction Platform, Trend Micro was able to begin iteratively refactoring their Workload Security product suite, using AI and automation to identify and decompose complex circular interdependencies in their critical “Heartbeat” service in just 3 months, compared to 1 year for a previous project. This enabled them to increase deployment speed by at least 12X, improve the boundaries and domains of their codebase and engineering team structure to boost morale, and provided them a head start on decomposing and extracting remaining monolithic applications in their premier product suite.

The Challenges

A Legacy Monolith Among Cloud Native Microservices

Workload Security is part of Trend Micro's larger Cloud One™ Platform, which comprises multiple product lines and separate code bases—the majority of the Cloud One Platform was designed somewhat recently using a microservices architecture on AWS, but Workload Security was brought into the platform as a large monolith.

Internal engineers were able to encapsulate and re-host some minor services to AWS, sharpening the company's best practices around domain-driven design (DDD), DevOps, and CI/CD; yet despite best efforts, the majority of the product suite was still overwhelmingly monolithic in nature, with deep interdependencies across multiple modules.

Negative Impact On Engineer Morale

With the majority of Trend Micro's platform already in the cloud, the teams working on the Workload Security monolith were dealing with an aging set of technologies and practices that were increasingly causing problems for Trend Micro.

Faced with a large shared codebase that wasn't divided into logical domains and teams with clear areas of concern, engineer morale began to suffer due to the lack of ability to make any sort of meaningful impact—especially during a system error or failure, during which making urgent fixes was a slow, manual process.

Ultimately, the engineering team desired more velocity to deliver faster value to customers and to modernize their code base in a way that it could integrate with the rest of their CI/CD pipeline for faster, cloud-based deployment processes

“Lift and Shift” Wasn't Delivering Enough Value

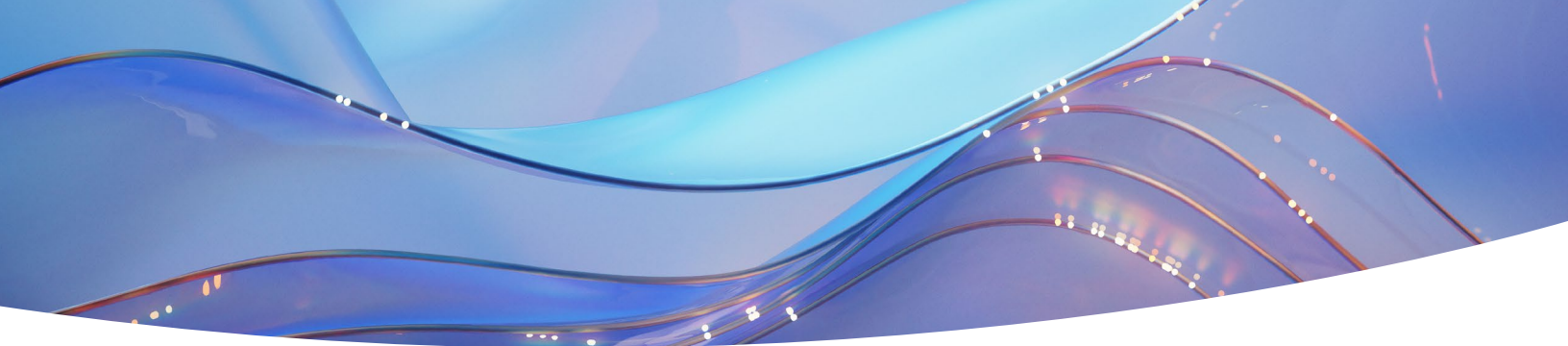
Once the team was able to lift-and-shift part of Workload Security to AWS, they began to notice challenges with performance and scaling—while compute efficiency was optimized in the cloud, without deeper refactoring it was necessary to keep these services always on and over-provisioned even when not being accessed.

Although some services were broken into modules (aka mini-services), the core functionality, data, and provisioning layers were still monolithic.

The lack of ability to scale, speed of deployment, and lack of product agility forced teams to turn down feature requests, affecting customer satisfaction and the potential for lost renewals/contracts.



“Without vFunction, we never would have been able to manually tackle the issue of circular dependencies in our monolithic system. The key service for our most important product suite is now untangled from the rest of the monolith, and deploying it to AWS now takes just 1 hour compared to nearly a full day in the past.”



The Solution with vFunction

Critical “Heartbeat” Service Targeted for Modernization

One of Trend Micro’s key services running on AWS is known as their “Heartbeat” service, which integrates data from various agents, sensors, and services across the product suite and originally had over 4000 Java classes.

The vFunction agent was installed to observe and learn the business domain-driven application flows, in the pre-production environment, including deep tracking of call stacks, memory, and object behaviors from actual user activity, events and tests.

This analysis, which synthesized dynamic learning with directed static code inspection of the binaries, ensured complete coverage of all app flows and enabled Trend Micro to begin carving out functional domains both in the code and amongst the engineering team itself, all of whom had previously been trying to work on the same big monolith together.

Revelations of AI-based Analysis

vFunction uses patented methods of static analysis, dynamic analysis, and dead code detection, plus data science and AI that applies graph theory and clustering algorithms to automatically identify optimal business-domain microservices and untangle dependencies across database calls, classes, and resources.

vFunction revealed a host of never-before-seen dynamic call flows across classes in the Heartbeat service–this indicated that these interconnections among classes were only occasionally called by services elsewhere.

These insights also revealed that these highly complex, inter-class dependencies were pulling business logic from unnecessary sources, and removing these with iterative refactoring also exposed dependencies in additional product modules.

Iterative Refactoring of Circular Dependencies

Complex, circular dependencies like those existing in the target Heartbeat service meant setting the right scope–this problem occurs when class dependencies that exist in the common library also bring classes that should belong to the Heartbeat service, and vice versa.

To solve this, Trend Micro iteratively refactored sets of classes, trusting vFunction’s patented static analysis methods to identify any emergent interdependencies.

This enabled them to identify the right set of classes to be scoped to the Heartbeat service and what should be scoped into the common library, iteratively removing the dependencies.



The Results

From 1 Year to Less Than 3 Months (For Refactoring)

In the past, Trend Micro spent over 1 year manually refactoring and extracting a similarly sized monolithic application—using AI and automation with vFunction allowed them to accelerate time-to-completion by 3X, completing the Heartbeat project in less than 3 months.

From 1 Day to 1 Hour (For Deployments)

Iterative refactoring revealed the challenging, interdependent nature of the Workload Security products—by fully extracting the key Heartbeat service from the rest of the monolith, Trend Micro was able to take the full deployment process down from 24 hours to just 1 hour for updates to the Heartbeat service.

Improved Team Morale

Powered by vFunction's AI-driven insights and automation, Trend Micro was able to refactor not only their Heartbeat service to be significantly easier to work on and faster to deploy and fix, but created a more positive, purposeful engineering ecosystem with clear domains and areas of responsibility.

Rapid Validation of Hypotheses

With the insights provided by vFunction, Trend Micro was able to quickly and effectively validate hypotheses about ideal code areas for extraction into services—vFunction provided enough visibility for the team to confirm their hypotheses in a matter of weeks.



Ready To Scale To More Monoliths

After a month of iterative refactoring, Trend Micro was able to decompose most of the circular dependencies detected in the analysis phase for the Heartbeat service—as an additional benefit, extracting this service from the monolith enabled them to more easily identify and tackle other dependencies in the system for two other services in the product suite as well: Security Control and Inventory Resources.



[Request a Demo](#)

[About vFunction](#)

vFunction is the first and only AI-driven platform for developers and architects that intelligently and automatically transforms complex monolithic Java applications into microservices, restoring engineering velocity and optimizing the benefits of the cloud. Designed to eliminate the time, risk and cost constraints of manually modernizing business applications, vFunction delivers a scalable, repeatable factory model purpose-built for cloud native modernization. With vFunction, leading companies around the world are accelerating the journey to cloud-native architecture and gaining a competitive edge. vFunction is headquartered in Palo Alto, CA, with offices in Israel. To learn more, visit vFunction.com.