LEGIT

CASE STUDY

Large health insurance provider gets fast visibility into SDLC risks with Legit

The CTO and his team at a large health insurance provider were struggling to get visibility into their SDLC, its security issues, and its security controls.

The fact that the firm primarily outsources their application development added urgency to the issue. They were conducting static code analysis, but they needed a better way of understanding the full picture of their code, its security, and its components – both to better secure it and to create SBOMs for compliance purposes.

This lack of visibility was especially apparent after the Log4j vulnerability announcement. It took the team a full six months to figure out where they had Log4j in use.

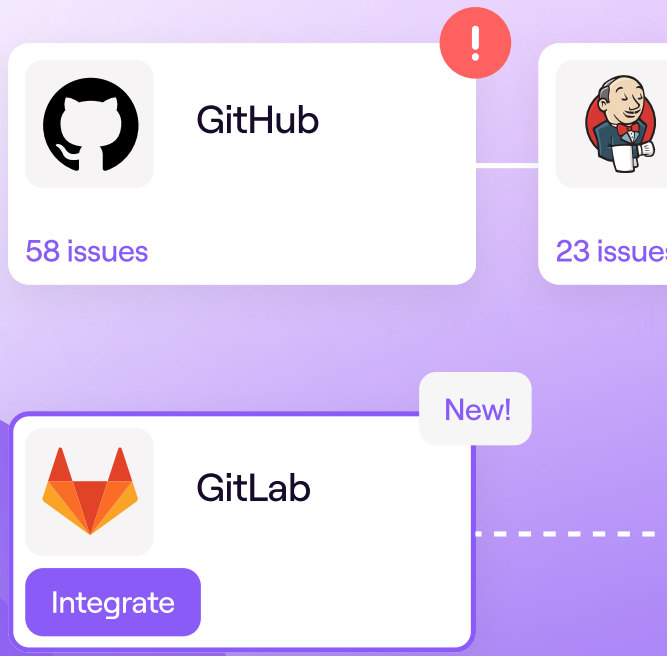
After explaining their problem to a partner, they were introduced to the team at Legit.

SDLC visibility – from AI to vulnerabilities and secrets – in minutes

The team implemented the Legit AI-native ASPM platform and after being impressed by how quickly the solution was up and running, they were especially pleased with the breadth and depth of SDLC visibility – from AI to vulnerabilities and secrets – and how fast they could clearly pinpoint where they had security issues that needed to be addressed.

"It was just easy. It was an easy integration," said the CTO. "It's a great complement to our static testing; it added a layer of security analysis to the SAST. And it solved the issue that I was trying to understand, which is where in my code that a third party writes for me do I have security issues?"

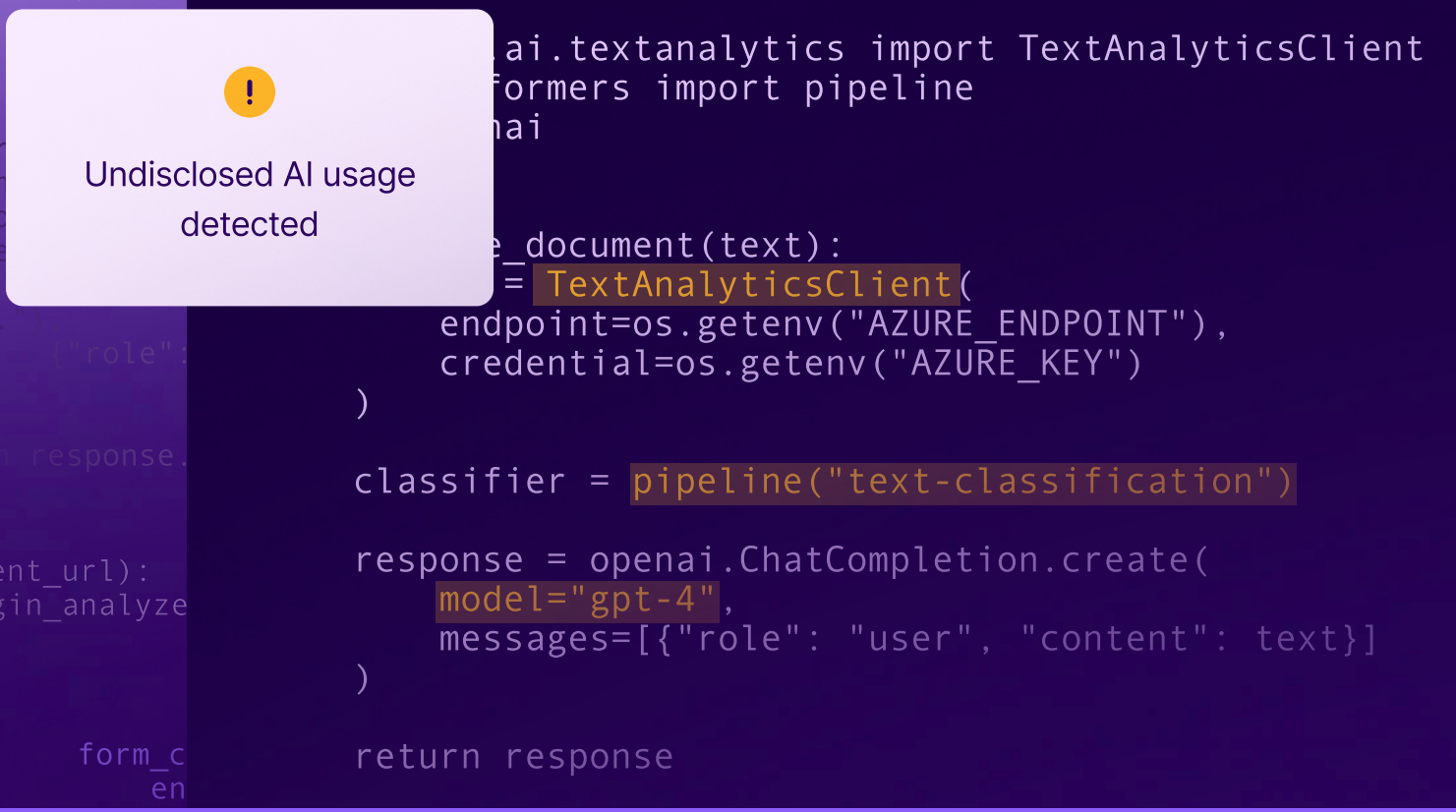
The SOC at the insurance company has seen significant value from the Legit platform as well. Previously, when the SOC got an alert about a significant vulnerability, it would trigger a complex process that involved about 20 people figuring out if there was exposure and where. With the Legit platform in place, that investigation now takes about 5 minutes.



AI discovery

The CTO also found value in Legit's ability to identify AI in use. For example, one of the firm's vendors told them they solely use Azure AI. But then Legit revealed HuggingFace and OpenAI components in their code.

> "Legit puts intelligence into our conversations," says the CTO. "It gives you the ability to understand a new piece of software a vendor is introducing, to very quickly analyze it, to see what kind of exposures you have from a vulnerability perspective, or an AI risk perspective. It's just absolutely fantastic."

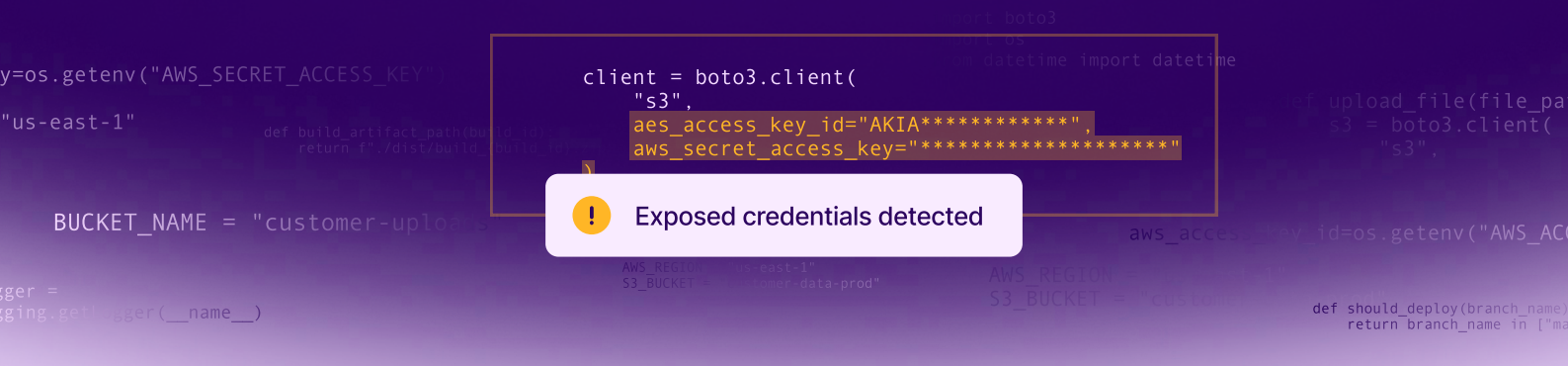


Highlighting exposed credentials

“Legit identifies the security-related technical debt that results from poor coding practices,” says the CTO. “And it’s debt that will get you into trouble if you allow the code to go into production.”

An early and clear indication of Legit’s value was that the platform immediately highlighted that they were passing clear-text password credentials to AWS.

“First, I didn’t even know we were using AWS,” said the CTO. “I certainly didn’t know that there were IDs and passwords in our code, and definitely didn’t understand that we were passing them, in the clear, to a cloud provider. Because Legit integrated so easily into Azure DevOps, it revealed this issue almost immediately.”



Better visibility = better collaboration

Ultimately, the partnership between the insurance firm and Legit has yielded better collaboration between security, engineering, and executives. With one source of truth about the SDLC and all its components and controls, the teams have a “common language” to discuss risk and how to address it.

Build fast with AI. Secure with AI-powered ASPM.

LEGIT

The Legit Security ASPM platform is a new way to manage application security in a world of AI-first development, providing a cleaner way to manage and scale AppSec and address risks. Fast to implement, easy to use, and AI-native, Legit has an unmatched ability to discover and visualize the entire software factory attack surface, including a prioritized view of AppSec data from siloed scanning tools. As a result, organizations have the visibility, context, and automation they need to quickly find, fix, and prevent the application risk that matters most. Spend less time chasing low-risk findings, more time innovating.