DETECTIFY CASE STUDY: GRAMMARLY

How Grammarly combines Detectify with bug bounty to optimize their security posture

detectify

AN INTERVIEW WITH:

Vladimir Suslenko
APPLICATION SECURITY ENGINEER

grammarly

About Grammarly

Grammarly provides software that helps 30 million people and 30,000 teams write more clearly and effectively every day.

Industry:	Internet
Company size:	500+ employees
Location:	Remote; hubs in Kyiv, San Francisco, Vancouver, New York
Detectify products:	Surface monitoring, Application scanning
Detectify integrations:	Slack, Jira, Detectify API Integration

Problems & challenges

Vladimir Suslenko is an Application Security Lead at Grammarly, specializing in infrastructure and product security with strong knowledge of security operations.

Grammarly seeks to strengthen its security position by utilizing the power of automation and expertise gathered from the hacking community.

They chose Detectify to deliver unobtainable results that could only come from ethical hackers in Detectify's Crowdsource team.

Grammarly had the difficult task of creating an inventory of all of its product offerings and applications. As they moved rapidly, with thousands of users added daily, development worked parallel with 50+ publicly facing applications.

Keeping track of these continually moving applications and their domains was where Detectify came in, providing an effortless discovering and alerting system for vulnerabilities in Grammarly's dynamic approach to shift-left security.

Working with hacker knowledge

Grammarly highly values the hacker community due to its innovative ways to produce findings and chain several instances together to create a more impactful bug.

The broad scope of Grammarly's applications meant that they needed a combination of automation and bug bounty to scan all of their external applications. The only thing that most hackers see in a bug bounty program is the external perimeter of Grammarly.

There are many more complicated services and integrations behind the scenes. These need to be categorized and assessed to determine an appropriate level of risk a vulnerability can pose.

Detectify Surface Monitoring was a huge help due to Grammarly's large perimeter containing 100's of public subdomains and even more internally, which Grammarly had to manage. The moment an internal subdomain was exposed with a takeover possibility, **Detectify could find it and trigger an alert via their Slack integration.**

From their bug bounty programs, Grammarly saw that Surface Monitoring could pick up subdomain takeovers and report them as quickly if not before hackers reported it through their bug bounty program.

"Surface Monitoring is an impressive product as it allows us to manage all of our subdomains and quickly search for new vulnerabilities."

"Surface Monitoring is an impressive product as it allows us to manage all of our subdomains and quickly search for new vulnerabilities."

Vladimir Suslenko
APPLICATION SECURITY ENGINEER AT GRAMMARLY

Fast remediation times through automation

Grammarly's security team knew that they needed to both find and remediate vulnerabilities as quickly as possible. A subdomain takeover can be fast to find but a complicated task to remediate.

To counteract this, Grammarly built a working Proof of Concept that triggered when a subdomain takeover was detected through Detectify. Grammarly could automatically claim that subdomain and remove the threat.

The remediation process

Grammarly uses the Detectify API to receive alerts about any new discovered vulnerabilities. Depending on the vulnerability type this may trigger an additional alert to the application security team.

The application security team then confirms its validity and executes a pre-configured command from the alert to remediate the weakness. For all other vulnerabilities, a pre-configured Jira ticket is exported and added to their Jira instance. *"The integration of alerts for Slack is really good as it works just like an RSS feed."*

Subdomain takeovers were just one of the many vulnerability types that Detectify found. Grammarly frequently benefits from their fast detection and praises

Detectify for its growth trajectory of continually adding new vulnerability research. *"The best thing that our team knows is that as Detectify moves as a company, it moves as a product."*

Reporting

Clear information without the noise

Grammarly's security team is appreciative of the reporting structure of Detectify as it makes it easier for their engineers to understand why something is a vulnerability.

In addition, it makes their security team's job considerably easier with straightforward remediation tips. Most importantly, it helps make their engineers write secure code.

How automation and reliability is key

Grammarly's Application security team's core work is rooted in Risk management. They have a strong understanding of the priorities of the company and try to match that with the speed that the engineering teams wish to deliver products.

Any results that are generated from external tools not only have to be accurate, but work seamlessly with their existing processes. Vladimir believes that you should empower all engineers so they can deliver secure code.

"Detectify makes security user-friendly by integrating so smoothly into our existing tools."

What would happen if you weren't using Detectify?

Detectify's level of coverage and speed to deliver new findings to Grammarly would be difficult to replicate. Grammarly would have been forced to find a collection of alternative tools or build one themselves.

This would result in spending time and resources to either find new services or create a more friendly infrastructure that limits the possibility of vulnerabilities occurring.

"Most solutions on the market are not friendly to developers. Detectify makes security user-friendly by integrating so smoothly into our existing tools," says Vladimir.

Vladimir's take on the future of security

Vladimir thinks that cybercrime will keep increasing and that companies should invest in more access controls to diversify their security options.

For Grammarly, this means searching for tools that provide the most significant leverage and contain operational intelligence to solve most issues themselves. Finding more methods to detect and remediate vulnerabilities is also key.

The process of getting a vulnerability fixed will also need to be overhauled, as this is not sustainable for many companies.

"Getting a vulnerability fixed is difficult within the perimeters of a company. You must first set up a scanner to detect the vulnerabilities and then have a triaging effort to assign it to the responsible team while determining the risk level that a vulnerability holds. It's not uncommon to wait a few weeks to have a medium-risk bug fixed even when prioritized."

For most bugs, it takes too much time to get fixed. Vladimir sees the future in the automation of detection and remediation.

Start your 2-week free trial on detectify.com

go hack yourself.