apiiro

Secrets Insights

Across the Software Supply Chain
June 2022

v1.0

Intro

In recent years, incidents originating from secrets creeping into code repositories and finding their way into adversaries' hands are growing in number and impact. To more rapidly exploit vulnerabilities, attackers automate their toolset for better identification and discovery of exposed secrets.

To minimize the risks of secrets slipping into code repositories, organizations must understand how internal repositories function and how developers and DevOps handle them. It's also critical to have benchmark data on the methods organizations use to overcome issues with exposed secrets.

Apiiro's security research team recently bagged, tagged, analyzed, and drew numerous insights from organizational data using almost two million code commits, spanning years of data mined for the purpose of gaining an industry-first broad review of secrets in code originating from internal repositories.

The report you are about to read was polished from multiple validation cycles and peer reviewed by fifteen industry leaders and experts in the field of application security with experience spanning multiple industries and backgrounds.

We hope that readers will find the presented information and insights valuable for their own strategies and for prioritizing remediation of secrets in code.

Thank you

This endeavor would never have been possible without the support, active discourse, practical knowledge, cross-examination and validation provided by a group of volunteer experts who contributed their time and effort to perfecting Apiiro's Secrets Insights 2022 Report.

We would like to thank every contributor involved in the research stages and culminating in the creation of this report:

Alex Mor, Global Director of Application Security AbInBev

Amol Shukla, Executive Director, Morgan Stanley

David Coursey, Application Security Lead, DataRobot

Elli Shlomo, Director Cybersecurity Architect, Gopuff

Frank Catucci, Head of Application & Product, DataRobot

Hila Dagan, Vice President Application Security, Blackrock

James Chiappetta, Senior Vice President Cybersecurity, Blackstone

Julie Davila, Senior Director Application Security, Sophos

Kenni Boisjoly-Moreau, Security Engineering Lead, Bitcoin.com

Maor Saubron, Director of Cybersecurity, Amdocs

Roman Lavrik, Lead Vulnerability Analyst, MasterCard

Rotem Reiss, Application Security Engineer, Playtika

Roy Avrahamy, Application Security Engineer, Compete

Yaniv Toledano, VP Global CISO & IT, Pagaya

Anonymous, Managing Director, Financial Sector

Apiiro internal reviewers

Igal Kreichman, VP Engineering

Yonatan Eldar, CTO

Eldan Ben Haim, Chief Architecture Officer

Apiiro's Security Research

Tafat Gaspar, Senior Security Researcher

Moshe Zioni, VP Security Research

Table of contents

The data

To discover and understand real-world risks of hardcoded secrets across the software supply chain, Apiiro's Security Research team analyzed a total of:

45K+
Secrets were detected

1,967,882
Commits

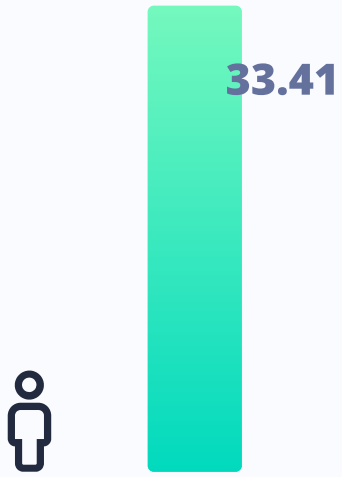
820K+
Pull Requests

25K+
Repositories

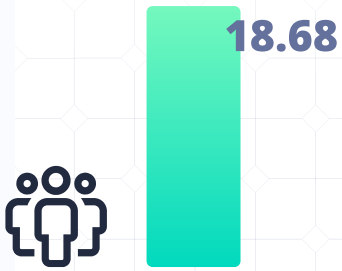
All data was collected and analyzed from internal repositories of small to large organizations, providing us with statistically significant data and actionable insights.

The average number of secrets detected during our investigation can be classified by an organization's developers count

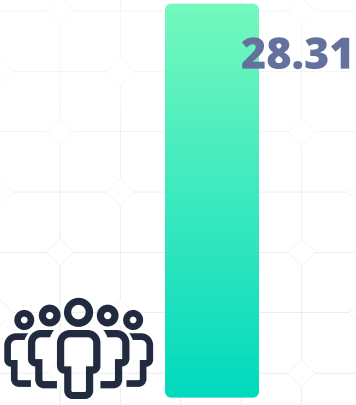
The average number of detected secrets per 1K commits:



110 developers on average
(**5K - 40K** commits)



439 developers on average
(**40K - 110K** commits)



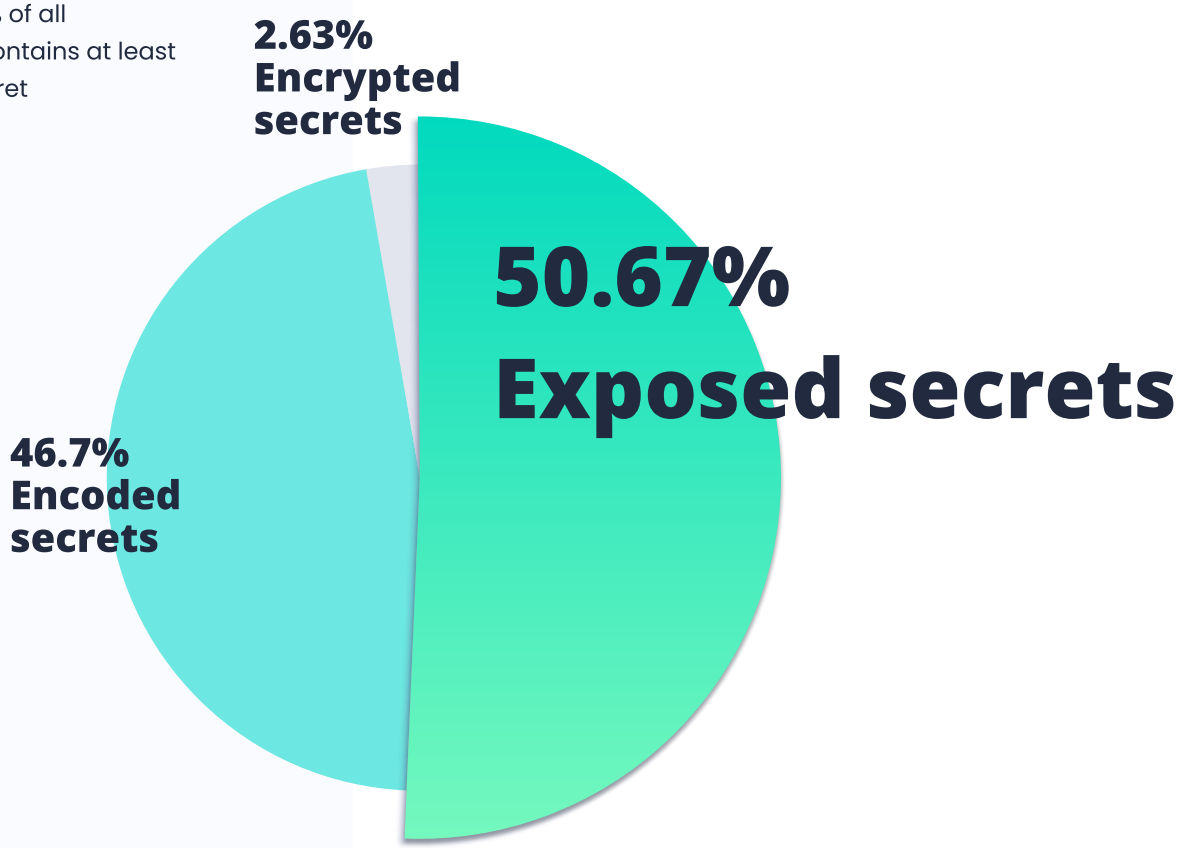
1056 developers on average
(**110K - 650K** commits)

25.7 commits out of 1,000 expose a secret.

This number is more than 8 times greater than the number we see from reports about public repositories.

Types of secrets

When we focus on secrets distribution per repository, we find that almost **every 5th repository** (17% of all repositories) contains at least one stored secret



This means that **every other secret** found can immediately be used by a malicious person who gets access to the relevant files.

For every 1,000 repositories with secrets, about 7 repositories have secrets that are exposed to anyone on the Internet.

apiiro

Secrets exposure

While secrets come in all shapes and sizes, the ones that pose the greatest risk are those which can be used as-is. Others may need to be decrypted, which slightly reduces the likelihood they will be exploited, thus causing less of an impact in terms of abuse potential.

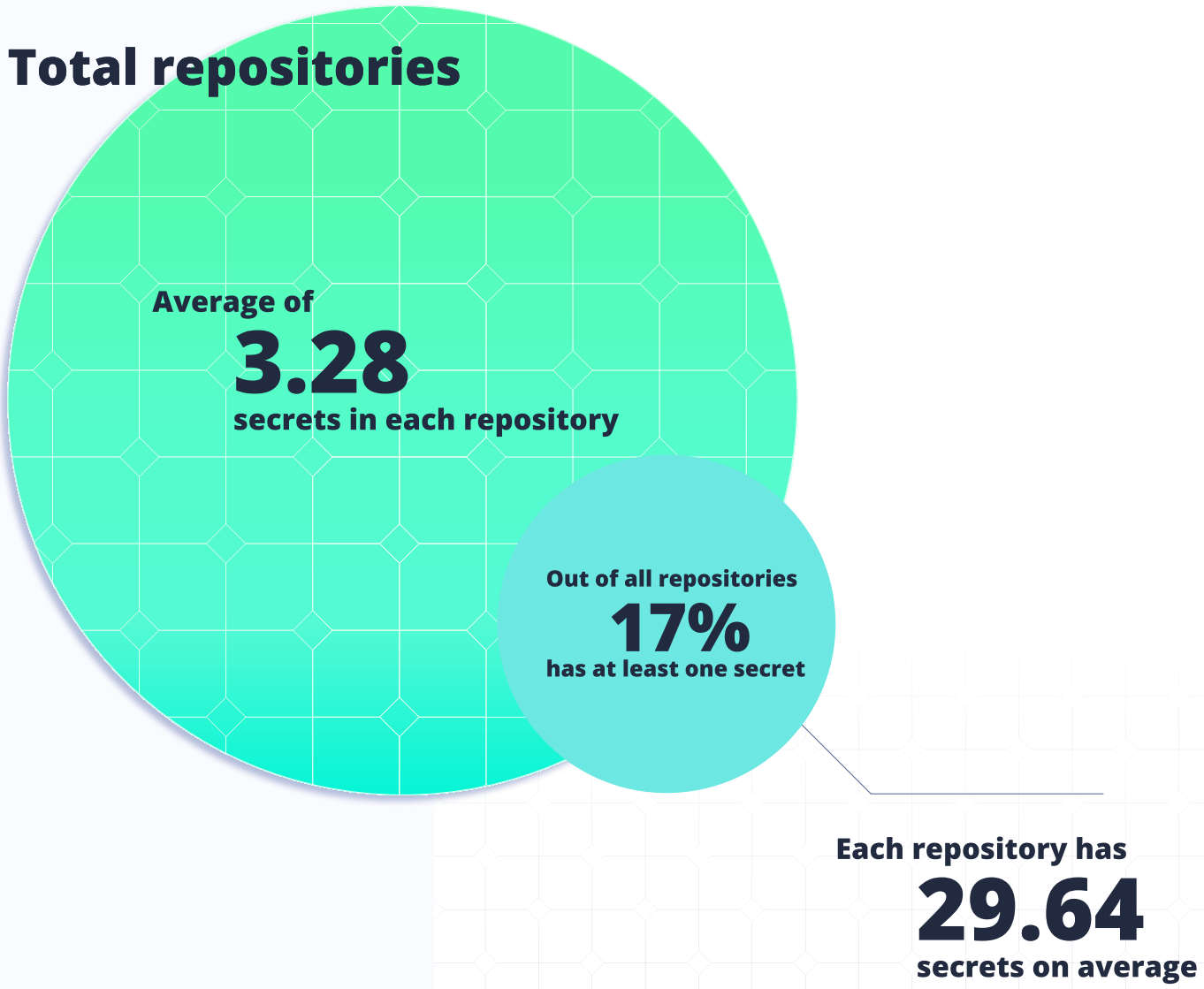
An example of an exposed secret can be an API key. When used, it may give an entity the ability to collect data, modify records or spin up new instances, just to name a few common operational permissions possible with an API key.

A real-life example is the Twitch Code Leak, which was a massive data breach that included source code and creator earnings, leading to severe brand damage.

Encrypted or hashed secrets might need to be “brute-forced” to decrypt them, meaning their use is constrained in a way (for example, Slack webhooks that can only be used for spam) that limits abuse potential and leads to a substantially lower risk than exposed ones.

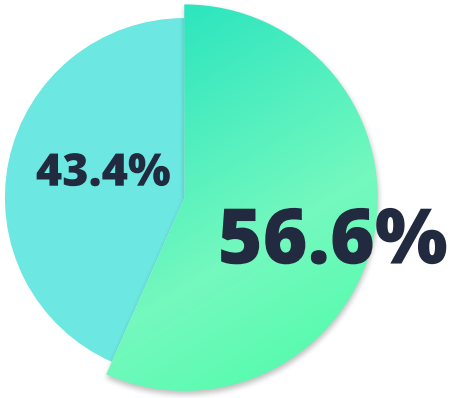
Types of secrets

On average, there are 3.28 secrets in each repository. When we focus on repositories that have secrets, each one has 29.64 secrets on average. Also, 0.72% of repositories with secrets are public.



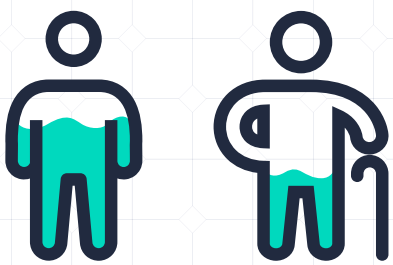
The people behind secrets

6% of the developers account for all of the secrets in an organization, while a mere 0.57% of the developers account for 56.61% of them!



When we normalize the top 10 secrets contributors with the number of the commits they've inserted, we find that the number of days that these developers are active in the organization is 28.65% less than the average number of days of all active employees.

Newer employees are more prone to insert secrets.



In depth analysis

Out of all secrets
38.15%
are in repositories with personal information (PII)

17.31%
of secrets are in repositories that are user-facing and have sensitive data

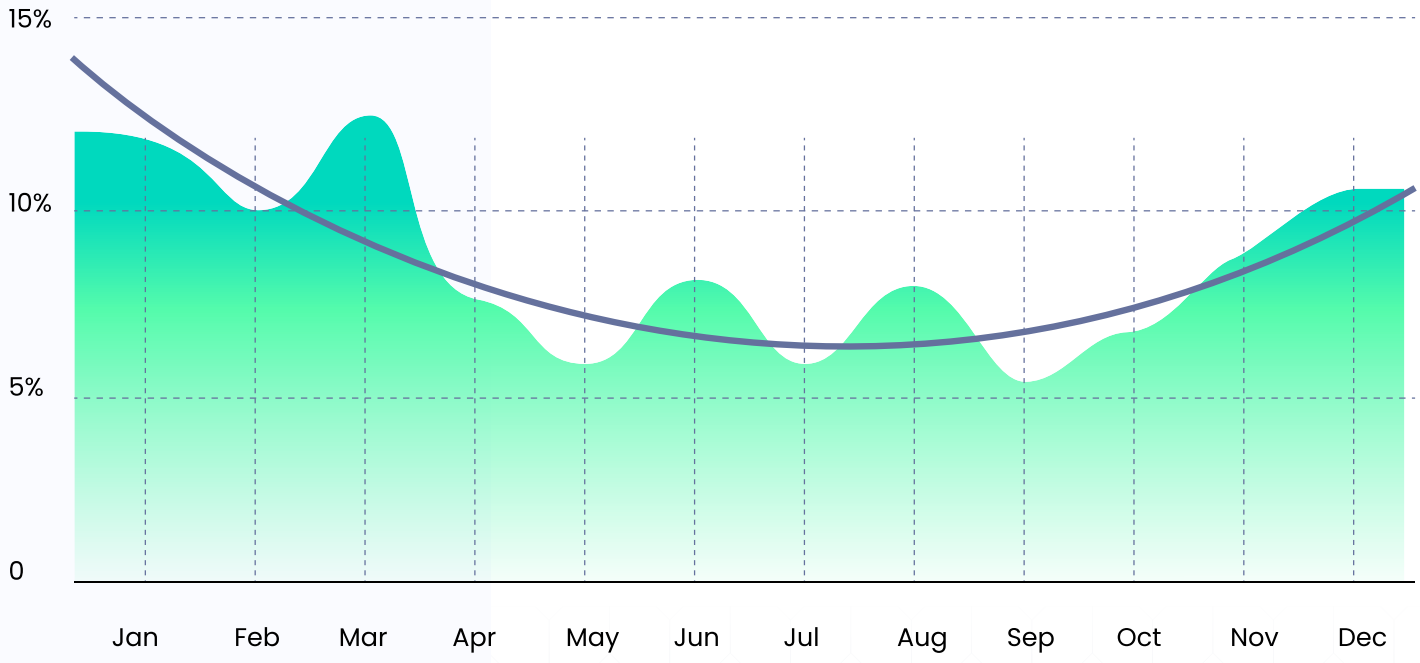
84.5%
of the secrets in these repositories are exposed secrets

50.67%
out of all secrets are exposed secrets

Combining all these factors can lead to a severe breach that can cause serious damage to an organization.

In depth analysis

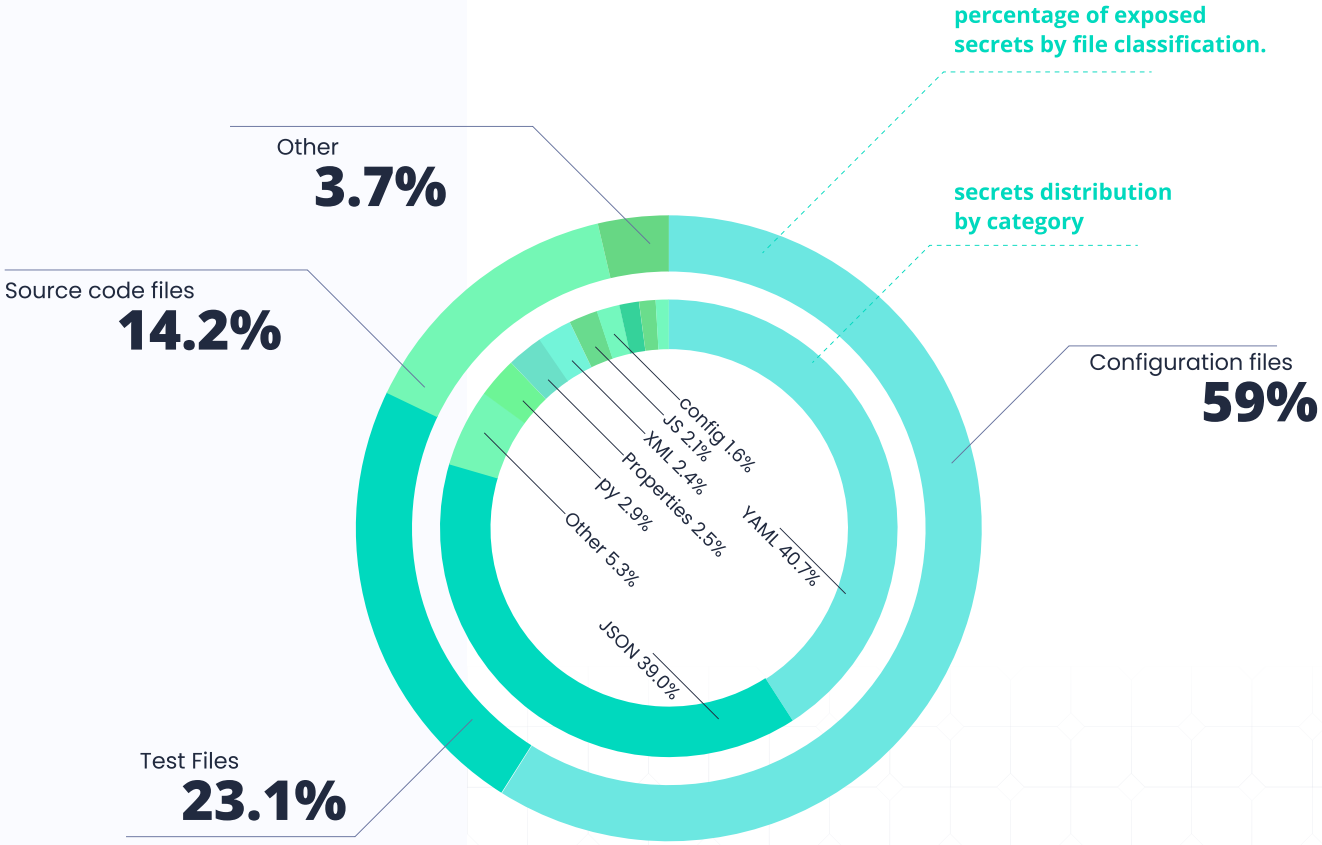
The first quarter of the year accounts for 34.34% of the detected secrets.



This observation is consistent with the notion that an organization's activity at the beginning of the year affects the amount of secrets inserted. For example, organizations have more code commit activity following new contracts, new features implementation, license renewal, etc.

In depth analysis

The majority of secrets are found in formats typically used for configuration files. More than 40% of all the secrets are detected in YAML files, and more than 39% are in JSON files.



Next in popularity, but with a very large difference, are .py, .properties, .xml and .js files. The top five file extensions account for 87% of all the secrets. Unsurprisingly, the majority of these secrets are found in formats typically used for configuration files (.yaml, .json, .properties, .xml).

In depth analysis

Plain-text passwords are

42.55%

of all exposed secrets

Base64 encoded strings are

24.83%

of all exposed secrets

hex-encoded strings are

11.16%

of all exposed secrets

these three categories represent 78.54% of all secrets

When we combine the secrets category and the file classifications results, we find that

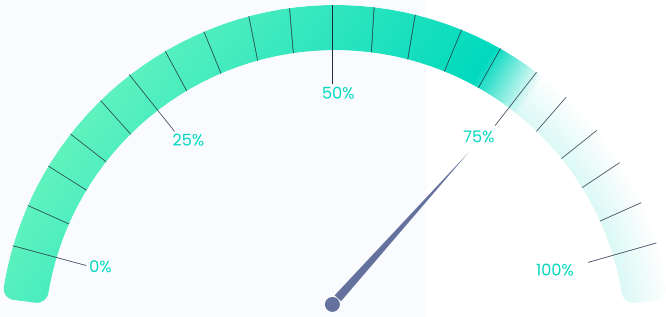
42.92% of plain-text passwords are in configuration files, so by paying extra attention to these secrets in the relevant files, we can remediate about 20% (18.26) of all secrets.

In depth analysis

1.91

Average duplication per secret

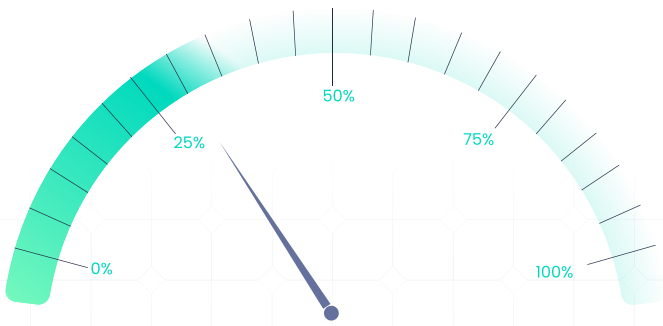
The **top 10** most frequent secrets **account for 25.49%** of all the detected secrets



By removing the top **10 persistent secrets,** you can get rid of a

quarter

of your stored secrets



and you can remediate

~50%-70% of your risky commits

much faster with the resources you have!

The Mean-Time-to-Removal (MTTR), is found to be 90 days, which means that those secrets are stored in source code repositories for about a quarter, on average.

apiiro

Conclusion

Secrets-in-code are here to stay, and more than that, they will multiply. Software development has changed, and code is no longer stored locally, but in the cloud. Cloud-based development has different security models, developers often have expanded access to the entire application including its production environments. A single compromised identity can have a catastrophic impact on the security of the entire application and infrastructure, and it all can start from an innocent mistake called secrets-in-code.

As we can observe from the report, there are several key actions you can take to quickly identify or even avoid the majority of your organization's hard-coded secrets. The first and one of the most significant ones is to identify the most involved developers (tested case is top 10), since they are responsible for more than 56% of the detected secrets on average.

Furthermore, if you want a clue as where to find them, we can observe that new-to-company employees are more prone to insert secrets, so this group can be a good one to start with. Another factor to pay attention to are calendar seasons. We have realized that the first quarter of the year accounts for the largest number of detected secrets out of the year, so by focusing and paying extra attention to this issue in this period, you can prevent secrets' insertions or remediate the inserted secrets fast and avoid potential risk to your organization. Pay extra attention to your public repositories, the most risky secrets are lying there!

Education, prevention & detection of secrets can thwart your next security incident!

apiiro

About Apiiro

A holistic Cloud-Native Application Security solution that helps security teams & developers proactively fix critical risks before releasing to the cloud

Discover

Every API, service, dependency & sensitive data to map the application attack surface across the software supply chain

Remediate

By running deep risk assessment, tying critical risks to code owners and proactively triggering contextual workflows

Free Risk Assessment