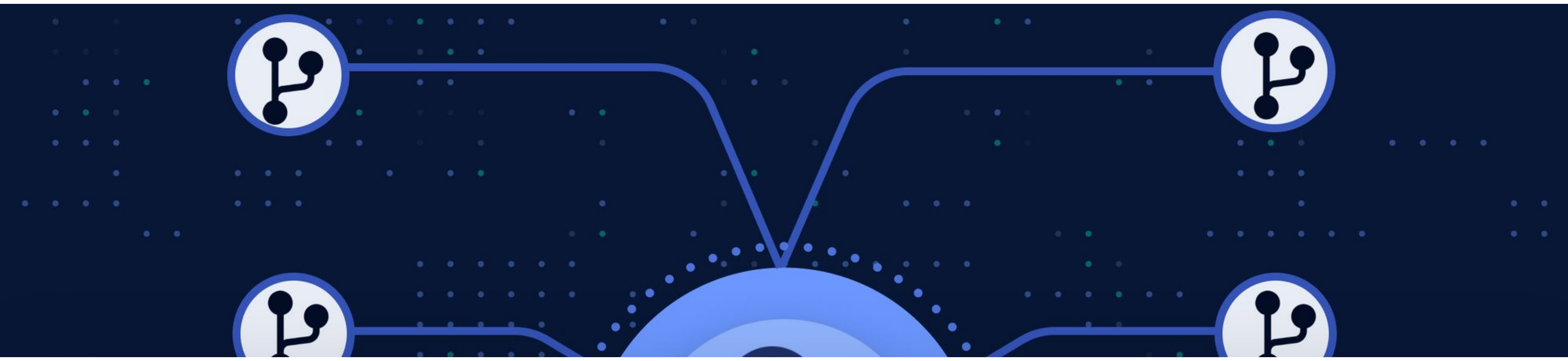
DEVSECOPS

Millions of .git folders exposed publically by mistake

MACKENZIE JACKSON
9 NOV 2022 · 6 MIN READ

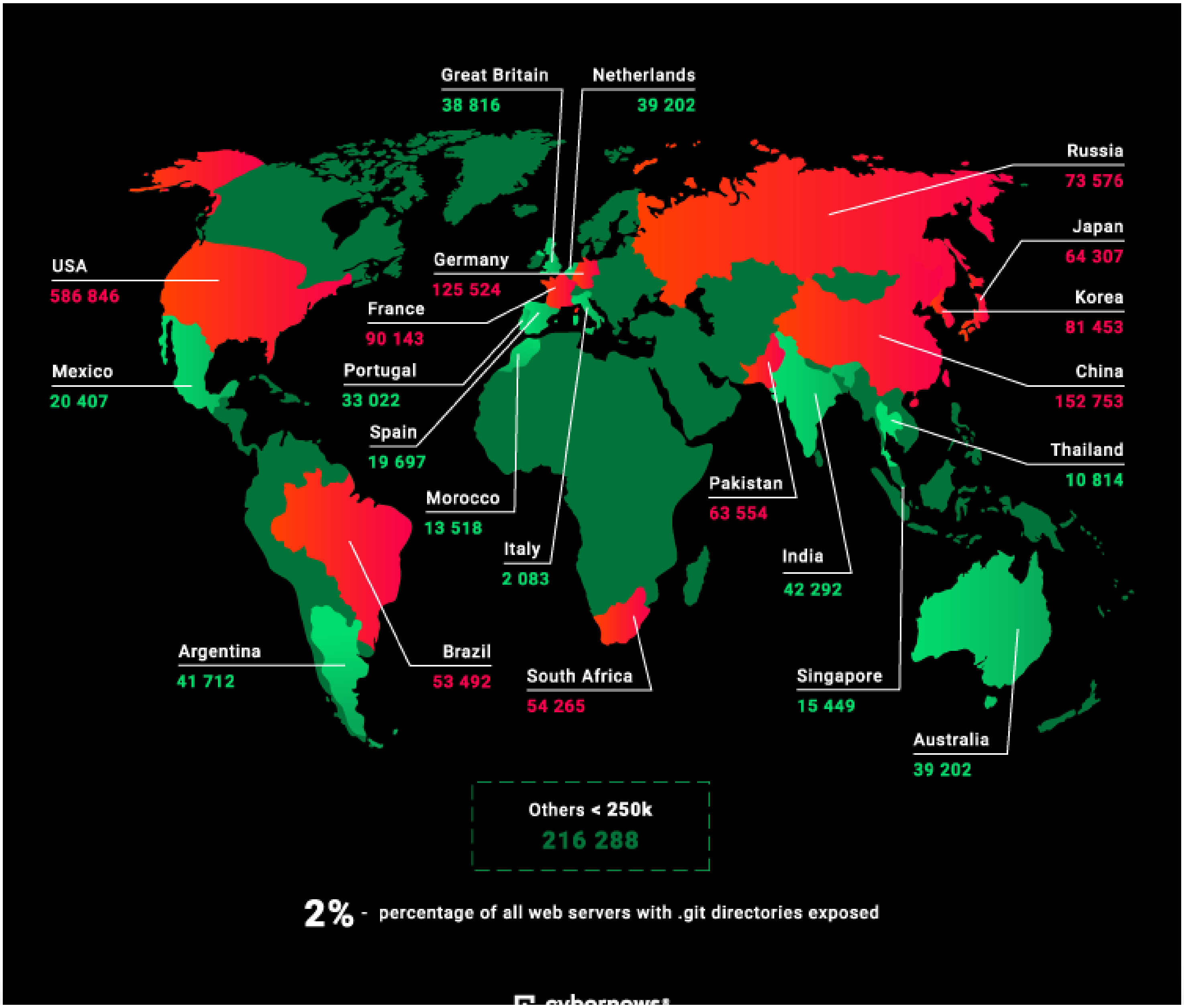
Share



2022 has been the year of source code leaks; Microsoft, Nvidia, Samsung, Rockstar, and many more companies have had their source code ***involuntarily open-sourced***. But some **new research by CyberNews has revealed that there are millions of private git repositories that are, in fact, not all that private**. In this article, we will take a look at the research on exposed git repositories, review why this can be such a problem, and suggest what you can do differently.

Nearly 2 million exposed git repositories

Git is a technology that nearly all software developers use to collaborate and version control their software. You will likely be familiar with git repository hosts like GitHub, BitBucket, or GitLab which all offer turnkey solutions to sign up and start pushing code to your own repositories and collaborating with others. Git can be a tricky technology and prone to user errors that can result in sensitive information being exposed. For example, when you create a new git repository on your machine, a `*.git*` folder is created, this is a folder that contains all the information and meta-data about your project since it was created. If you made an edit from 2012 to your application, 10 years later that edit is still hidden in that *.git* folder. If you commit an API key on a development branch 3 years ago, it's still inside this *.git* folder. **Basically, unless you are certain you and no one on your team have ever committed anything**



×

Notice

We and selected third parties use cookies or similar technologies for technical purposes and, with your consent, for other purposes as specified in the cookie policy. Denying consent may make related features unavailable.
In case of sale of your personal information, you may opt out by using the link "Do Not Sell My Personal Information".
To find out more about the categories of personal information collected and the purposes for which such information will be used, please refer to our privacy policy.
Use the “Accept” button or close this notice to consent to the use of such technologies.

Accept

Learn more and customize

repositories and discovered

- 1053 fully or partially exposed git repositories
- 12 usernames with passwords in the git config data

These research projects, plus the countless related breaches we have had in the last two years show what a huge issue this truly is.

> ***"Even after I parallelized the scanning script it took some days to scan the 2.6 million domains. I did not expect many results, but was surprised how widespread the problem is."***
> ***SDCat***

How do .git folders become exposed?

There are many ways *.git* sprawls into locations they might not be intended. It could be a misconfiguration of a backup, or it could be an attempt to host your own git server, but usually, it is a deployment issue. One example that occurred multiple times was with a static website, if someone is using an Amazon S3 bucket to host their site, instead uploading the current version they have uploaded an entire directory including the *.git* folder. For anyone who understands how sensitive these are it will seem unlikely and shocking that this would happen, but it here are the key stats

- An average-sized company with 400 developers will have 13,000 secrets (1,000 unique) inside their private repositories
- GitGuardian scanned all public GitHub repositories and found over 6,000,000 secrets in 2021
- 3 out of every 1000 commits GitGuardian scanned contained at least one secret

There is a perfect storm resulting from the fact that git allows such easy collaboration of developers, secrets are meant to be programmatic, and that a git history never dies. While the research project by CyberNews didn't scan each repository for secrets in depth. They did find that 6% of the git repositories had their deployment credentials in the git configuration file….. I'm going to say that again, slowly.

> 6% of the exposed git repositories, had the credentials to deploy their applications, publicly accessible to the world, in the configuration file!

Screenshot of a configuration file with deployment credentials repositories weren’t actually private.

Source code, nearly always, contains more than just source code. In the history of a project, on often forgotten development branches sensitive information is hidden. This is why even though source code might not be considered a security-critical asset, it needs to be protected and this is why private code repositories that are public are such a big concern.

what can we do

The answer is obviously to make sure our git repositories are private right?

Well not quite. This research adds to the compelling pile of evidence that git repositories are not appropriate places to contain sensitive information. If your git repository is protected it becomes harder, but not impossible, for a bad actor to gain access to them. In 2021 the supply chain attack of CodeCov meant bad actors got access to up to 20,000 CodeCov users' private git repositories including HashiCorp, Twilio, and Rapid7 even though these were never exposed publicly. We have also seen companies like Uber have their repositories breached due to a compromised developer account. The point is that repositories have been exposed to bad actors as a weak point in our infrastructure and we need to secure them in more than one way

Summary

While there is huge evidence to show that git repositories are high-value targets for adversaries, we can add to this evidence the fact these repositories are easily accessible to attackers via domain and IP scanning searching for *.git* folders. Yes, we must better protect these repositories and scan our own infrastructure for exposed weaknesses but we also must ensure sensitive data like secrets are not our repositories as a minimum effort for security.

Share this article on **Twitter**, **HackerNews**, **LinkedIn**, or **Reddit**.

Key Highlights From the New NIST SSDF

1 Jun 2022 – 8 min read

See all 25 posts →

security incidents with GitGuardian's playbooks

Learn more about GitGuardian's no-code workflows and how they can help you enjoy some respite from the manual and grunt work no security engineer ever enjoys.

ZIAD GHALLEB
8 NOV 2022 – 4 MIN READ