MICRO FOCUS

Secure Artifact Management: Avoiding Corporate Ransom

Source Code in the Age of Ransomware

Traditionally, business IP was secured on internal servers, with no access from the outside world and no ability for users to make unauthorized changes. However, the increase in adoption and utilization of cloud-based solutions means that critical IP is frequently and inadvertently stored in the public cloud.

While most enterprises worry about cloud security, their concerns often focus on common cloud applications that store financial information, HR data or corporate documents. In the age of digital transformation, source code becomes an even greater target and key source of differentiation. Protecting against malicious insiders, direct competitors or bad actors must extend to version management tools.

What's the potential impact to your organization if a rogue actor were to gain access to your source code, or if they made unauthorized changes, deletions or held your source code ransom? As the recent hack of several git-based repositories has shown, this scenario is not as far-fetched as it may seem. Before we take a look at potential solutions to address this problem, let's understand the changes in

The top three vectors for exfiltrating data are database leaks, cloud applications, and removable USB drives.

Source: McAfee

process and tooling that have brought us to this junction.

The Rise of Developer Empowerment

The adoption of agile practices in development communities significantly increased over the past few years. The empowerment of development organizations is one of the biggest benefits that agile transformation has brought to the table, frequently simplifying the day-to-day activities of the entire development organization and increasing the pace of innovation. A function of this empowerment is that developers select the tooling they wish to use, and are no longer constrained by the dictates of corporate IT.

But how does this tooling discussion relate to the hacking of corporate git repositories and the associated risks that introduces to enterprise organizations?

Flash Point Paper

Application Delivery Management"

Securing intellectual property (IP) is a key and fundamental concern for all enterprise organizations.

The heavy and bureaucratic process models historically associated with software development were certainly shackles that needed to be removed. Yet strong processes are key to any software development undertaking.

Core constituent components of any Enterprise Grade Application lifecycle process:

- *Integrate security*
- *Maintain audit requirements*
- *Ensure compliance*

Learn how you can provide the necessary security, governance and compliance to implement Git at enterprise scale: www.microfocus.com/dimensions-cm/

Contact us at:
www.microfocus.com

Like what you read? Share it.

The Role of Developer Choice

In many instances, agile transformation is intimately associated with tooling transformation. Within many enterprises, development empowered teams make decisions on how, when and where intellectual property should be stored, the processes that should be used to support application development, and whether critical software assets are stored on-premises or in the cloud.

Distributed version management using Git-based technology has become the standard approach for version management in a large number of organizations. Ease of use, simplicity of user interface coupled with the perception of low-cost of ownership and minimal administrative overhead has driven significant adoption. Restricting freedom of choice for developers and introducing constraints that slow development teams down are not choices that any IT professional would advocate.

So again, what's the relevance to the hacking of Git-based repositories and the security of corporate IP?

Developers Leave the Door Wide Open

In many instances, corporate IT and security departments are not fully aware that mission-critical IP is being stored off-premises. The fact that the administrative burden of maintaining and administering servers and systems that they do not generally interact with has been seen as a benefit by corporate IT. No longer needing to patch, upgrade and support complex version control of software change and configuration management (SCCM) systems, coupled with more freedom for development teams to choose tooling has often reduced insight into the day to day practices of development teams.

Where SCCM solutions or internal git repositories were once centrally managed we now see many departments posting content to cloud-hosted repositories autonomously. This isn't necessarily a bad practice, but many enterprises are quickly learning that private repositories were inadvertently made public or were left unsecured. The fact that security teams are often unaware that core IT assets are being shared or left unsecured in the public domain is enough to give even the most relaxed security officer sleepless nights.

Ransomware and Other Threats

What's the worst that can happen if these publicly hosted repositories area accessed by a malicious actor? As reported in ZDNET, a hacker is asking for a ransom to release source code that's been downloaded and stored on their servers, else the code will be made public. While the exact nature of the hack remains unclear, it appears that in this particular instance hackers gained access to unsecured repositories and scanned repositories for configuration details in order to gain access to user credentials.

Code becoming available in the public domain is not that important for a college project, but what if it happened to valuable source code stored in a compromised project? This scenarios raises potential concerns: from audit failures, to code tampering, loss of customer confidence and revenue decline.

In Search of Secure Solution

So, what are the alternatives to publicly hosted repositories? Well naturally you can implement locally hosted git repos, but the challenges with repository sprawl and management will likely introduce an unexpected burden on your development or IT teams. Two-factor authentication is another option, but if it isn't implemented in a seamless manner it may introduce an unexpected overhead (and additional process burden) to the development organization.

Micro Focus® offers fully secured, enterprise grade source and artifact repositories that enable developers to use their preferred git client and supports multiple disparate git repositories to be brought under centralized and highly secured back end repository. User credentials are authenticated against your domain with full support for smart cards and fully immutable version history, ensuring that your IP and software assets remain tamper proof.