Verified Git commit signing with GitLab and Beyond Identity

Beyond Identity cuts through the anonymity of Git to provide a secure, scalable way for development and GitOps teams to immutably sign and verify the author of every commit. Our author verification API in Git proves that what you've shipped is what your developers actually built—and that nothing else got added.

Key Benefits:

By integrating Beyond Identity's Secure DevOps solution with your GitLab environment, organizations will achieve enterprise-grade security for their world-class public, community-developed code. This will enable organizations to:

- ✓ Verify authorship of every Git commit, eliminating any ambiguity regarding authorship, in order to create a significantly more secure and trustworthy development process.
- ✓ Eliminate reliance on the author field to prevent users from spoofing developers and admins on GitLab.
- ✓ Verify the identity of developers continuously and specifically at every code check in, especially those that do not login via an SSO or Git web console, and ensure the security posture of their devices.
- ✓ Generate an immutable and search-ready log of all developer authorizations and transactions for audit and compliance needs.

Verified Git commit signing secures code

GitLab's development platform has dramatically accelerated the ability to plan, manage and deploy software, while also allowing teams of developers to collaborate on a single release with tight version control. Use of Git repos is the primary way software is produced today.

The popularity of the platform has made it a target of attackers who exploit vulnerabilities in distributed, cloud-based Git environments. Recent attacks such as Solarwinds, Kaseya, and NotPetya have revealed that even mature, security-focused companies have enormous supply chain blindspots. They have shown that it's not only costly to remedy a breach of assets and third party tooling, credential theft, and key sprawl - it also erodes fundamental trust with the company and their intellectual property. Many times, that trust is irrecoverable.

Beyond Identity cuts through the anonymity of Git to provide a secure, scalable way for development and GitOps teams to immutably sign and verify the author of every commit. Beyond Identity's solution prevents malicious code commits by cryptographically binding access and signing keys to a validated corporate identity and a secure authorized device. This allows DevOps teams and development organizations to systematically inspect every commit to ensure that only source code that is signed by a corporate or authorized identity, is built into the product. Organizations can now secure their Git commits against unauthorized external and malicious insider threats.

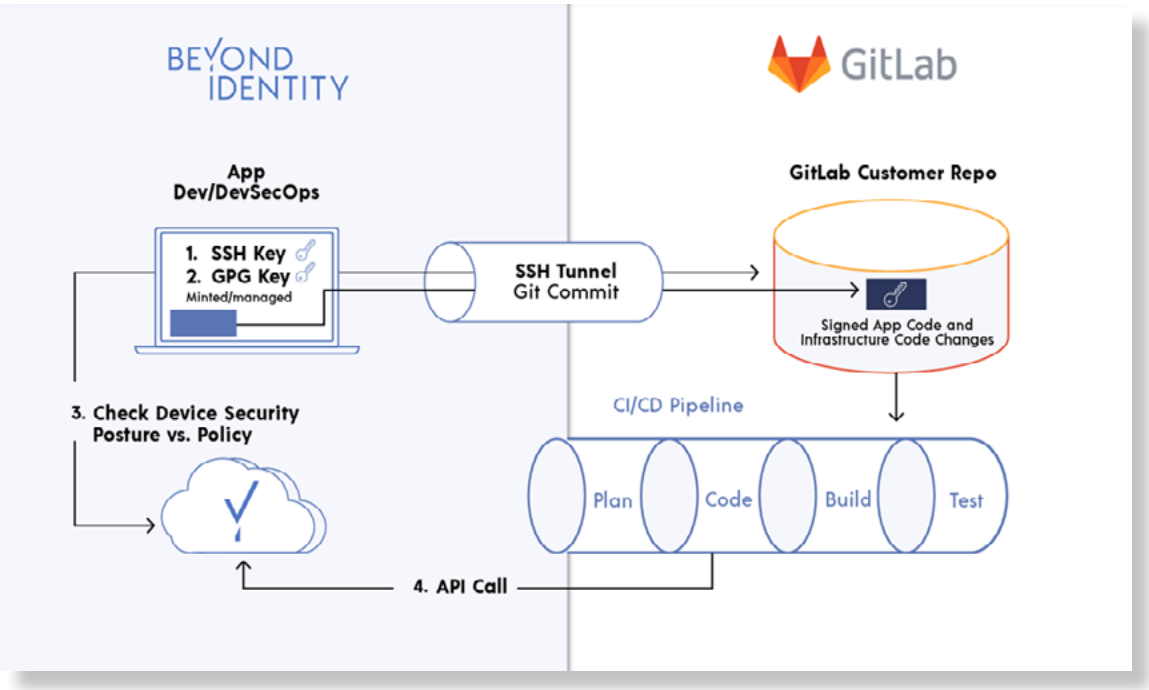
Solution overview

As software development moved to the cloud, the build environment became an attractive target for malicious actors looking to establish deep and broad compromise within organizations. From SolarWinds to Kaseya, the vulnerability of the software supply chain and the potential for damage has never been more clear or urgent. However, the speed and highly distributed nature of agile software development processes resists tighter security controls. GitLab and Beyond Identity have teamed up to track source code provenance, requiring developers to sign source code committed to corporate repositories using a validated corporate identity and device. The Beyond Identity Secure DevOps solution mints GPG keys, and stores them in the secure TPM of the device, to control access and bind the keys to a known user identity, all while creating no additional work or friction for developers. Leveraging what's already available in GitLab, the combined solution establishes a simple, secure, and automated way to confirm that all source code entering a corporate repository and processed by the continuous integration/continuous deployment (CI/CD) pipeline is signed by a key that is cryptographically bound to a corporate identity and device. This ensures trust, integrity, and auditability for every piece of source code that is built into the end software product.

Solution diagram

Beyond Identity creates/manages keys. Keys are cryptographically connected to a valid corporate identity. For developers it's "set it and forget it".

1. Project repo access with a Beyond Identity minted/managed SSH key.
2. Sign every code commit to protect code integrity with a Beyond Identity managed GPG key. (One time setup, set and forget for the developer)
3. Beyond Identity Policy engine restricts repo access and git commits to authorized and secure workstations to prevent unknown or insecure endpoints from accessing the repo or committing code or changing infrastructure settings for infrastructure as code (IAC).
4. Verify that every commit is signed by a GPG key connected to a validated identity at front end of CI/CD pipeline.



"Knowing that only users with confirmed corporate identities have access to the repo and that only source code signed by a GPG key that is cryptographically connected to a corporate identity protects the code base and ensures that any issues can be traced back to a specific person." — **Johnathan Hunt, Vice President of Security at GitLab**

Solution use cases

Audit Standards for Code Reviews: It's easy to spoof users in Git, so it's difficult to trace where a vulnerability came from. The only way to achieve code integrity and authenticity is to trust the signature on every commit.

Infrastructure As Code (IAC): If your infrastructure is compromised, attackers can open ports and change firewalls, leaving your network wide open. Preventing unauthorized commits is a crucial step in securing your IAC.

Third Party Development: Third party contributors are checking in code on non company-issued machines. Verifying author commit signing is the only way to ensure that a malicious actor didn't check in code.

About Beyond Identity

Beyond Identity is fundamentally changing how the world logs in with a groundbreaking invisible, unphishable MFA platform that provides the most secure and frictionless authentication on the planet. We stop ransomware and account takeover attacks in their tracks and dramatically improve the user experience. Beyond Identity's state-of-the-art platform eliminates passwords and other phishable factors, enabling organizations to confidently validate users' identities. The solution ensures users log in from authorized devices, and that every device meets the security policy requirements during login and continuously after that. Our revolutionary approach empowers zero trust by cryptographically binding the user's identity to their devices and analyzing hundreds of risk signals on an ongoing basis. The company's advanced risk policy engine enables organizations to implement foundationally secure authentication and utilize risk signals for protection, rather than just for detection and response.

For more information on why Unqork, Snowflake, and Roblox use Beyond Identity, please visit us here.

About GitLab

GitLab is The DevOps platform that empowers organizations to maximize the overall return on software development by delivering software faster and efficiently, while strengthening security and compliance. With GitLab, every team in your organization can collaboratively plan, build, secure, and deploy software to drive business outcomes faster with complete transparency, consistency, and traceability.

For more information, visit us here.