Gitops คืออะไร

GitOps is a way to do Kubernetes cluster management and application delivery. It works by using Git as a single source of truth for declarative infrastructure and applications. With Git at the center of your delivery pipelines, developers can make pull requests to accelerate and simplify application deployments and operations tasks to Kubernetes.

GitOps เป็นวิธีการจัดการ Kubernetes คลัสเตอร์และการส่งมอบแอปพลิเคชัน มันทำงานได้โดยใช้ Git เป็นแหล่งความจริงเดียวสำหรับโครงสร้างพื้นฐานและแอปพลิเคชั่นที่เปิดเผย ด้วย Git ที่เป็นศูนย์กลางของท่อส่งของคุณผู้พัฒนาสามารถส่งคำขอเพื่อเร่งและลดความซับซ้อนของการปรับใช้แอปพลิเคชันและงานการดำเนินการให้กับ Kubernetes

An operating model for building cloud native applications

รูปแบบการดำเนินงานสำหรับการสร้างแอปพลิเคชันเนทีฟแบบคลาวด์

GitOps can be summarized as these two things:

1. An operating model for Kubernetes and other cloud native technologies, providing a set of best practices that unify deployment, management and monitoring for containerized clusters and applications.
2. A path towards a developer experience for managing applications; where end-to-end CICD pipelines and git workflows are applied to both operations, and development.

1. รูปแบบการดำเนินงานสำหรับ Kubernetes และเทคโนโลยีเนทีฟคลาวด์อื่น ๆ ให้ชุดของแนวปฏิบัติที่ดีที่สุดที่รวมการปรับใช้การจัดการและการตรวจสอบสำหรับคลัสเตอร์และแอพพลิเคชั่นที่รวมเข้าด้วยกัน

2. เส้นทางสู่ประสบการณ์นักพัฒนาสำหรับการจัดการแอปพลิเคชัน โดยที่ end-to-end CICD ไปป์ไลน์และ git เวิร์กโฟลว์ถูกนำไปใช้กับทั้งการดำเนินงานและการพัฒนา

Key benefits of GitOps

ประโยชน์ที่สำคัญของ GitOps

Automated delivery pipelines roll out changes to your infrastructure when changes are made to Git. But the idea of GitOps goes further than that – it uses tools to compare the actual production state of your whole application with what’s under source control and then it tells you when your cluster doesn’t match the real world.

By applying GitOps best practices, there is a ‘source of truth’ for both your infrastructure and application code, allowing development teams to increase velocity and improve system reliability.

ท่อส่งอัตโนมัติจะแผ่การเปลี่ยนแปลงโครงสร้างพื้นฐานของคุณเมื่อทำการเปลี่ยนแปลงกับ Git แต่แนวคิดของ GitOps ยิ่งไปกว่านั้น - มันใช้เครื่องมือในการเปรียบเทียบสถานะการผลิตจริงของแอปพลิเคชันทั้งหมดของคุณกับสิ่งที่อยู่ภายใต้การควบคุมของแหล่งที่มาแล้วมันจะบอกคุณเมื่อคลัสเตอร์ของคุณไม่ตรงกับโลกแห่งความจริง

ด้วยการใช้แนวทางปฏิบัติที่ดีที่สุดของ GitOps จะมี 'แหล่งที่มาของความจริง' ทั้งโครงสร้างพื้นฐานและรหัสแอปพลิเคชันของคุณช่วยให้ทีมพัฒนาสามารถเพิ่มความเร็วและปรับปรุงความน่าเชื่อถือของระบบ

The benefits of applying GitOps best practices are far reaching and provide:ประโยชน์ของการใช้แนวทางปฏิบัติที่ดีที่สุดของ GitOps นั้นอยู่ไกลและให้:

1. **Increased Productivity**   
   Continuous deployment automation with an integrated feedback control loop speeds up Mean Time to Deployment. Your team can ship 30-100 times more changes per day, increasing overall development output 2-3 times.

1.ผลผลิตเพิ่มขึ้น

การปรับใช้อัตโนมัติอย่างต่อเนื่องพร้อมด้วยลูปควบคุมตอบกลับแบบรวมจะช่วยเพิ่มความเร็วในการปรับใช้เวลาเฉลี่ย ทีมของคุณสามารถจัดส่งการเปลี่ยนแปลงได้มากขึ้น 30-100 เท่าต่อวันเพิ่มผลการพัฒนาโดยรวม 2-3 ครั้ง

1. **Enhanced Developer Experience**  
   Push code and not containers.  Developers can use familiar tools like Git to manage updates and features to Kubernetes more rapidly without having to know the internal of Kubernetes. Newly on-boarded developers can get quickly up to speed and be productive within days instead of months.

2.ประสบการณ์นักพัฒนาที่เพิ่มขึ้น

กดรหัสและไม่ใช่คอนเทนเนอร์ นักพัฒนาสามารถใช้เครื่องมือที่คุ้นเคยเช่น Git เพื่อจัดการการอัปเดตและฟีเจอร์ต่าง ๆ ให้กับ Kubernetes ได้รวดเร็วยิ่งขึ้นโดยไม่ต้องทราบภายในของ Kubernetes นักพัฒนาซอฟต์แวร์ใหม่ที่มาพร้อมบอร์ดสามารถเพิ่มความเร็วได้อย่างรวดเร็วและมีประสิทธิผลภายในไม่กี่วันแทนที่จะเป็นเดือน

1. **Improved Stability**  
   When you use Git workflows to manage your cluster, you automatically gain a convenient audit log of all cluster changes outside of Kubernetes. An audit trail of who did what, and when to your cluster can be used to meet SOC 2 compliance and ensure stability.

3.ปรับปรุงเสถียรภาพ

เมื่อคุณใช้เวิร์กโฟลว์ Git เพื่อจัดการคลัสเตอร์ของคุณคุณจะได้รับบันทึกการตรวจสอบที่สะดวกสบายของการเปลี่ยนแปลงคลัสเตอร์ทั้งหมดภายนอก Kubernetes โดยอัตโนมัติ หลักฐานการตรวจสอบของผู้ที่ทำสิ่งใดและเมื่อใดในคลัสเตอร์ของคุณสามารถใช้เพื่อให้เป็นไปตามข้อกำหนด SOC 2 และสร้างความมั่นคง

1. **Higher Reliability**  
   With Git’s capability to revert/rollback and fork, you gain stable and reproducible rollbacks.  Because your entire system is described in Git, you also have a single source of truth from which to recover after a meltdown, reducing your meantime to recovery (MTTR) from hours to minutes.

4.ความน่าเชื่อถือที่สูงขึ้น

ด้วยความสามารถของ Git ในการย้อนกลับ / ย้อนกลับและการแยกคุณจะได้รับความเสถียรและการทำซ้ำได้ เนื่องจากระบบทั้งหมดของคุณมีการอธิบายไว้ใน Git คุณยังมีแหล่งความจริงเดียวที่จะกู้คืนหลังจากการล่มสลายช่วยลดเวลาในการกู้คืน (MTTR) จากชั่วโมงเป็นนาที

1. **Consistency and Standardization**  
   Because GitOps provides one model for making infrastructure, apps and Kubernetes add-on changes, you have consistent end-to-end workflows across your entire organization. Not only are your continuous integration and continuous deployment pipelines all driven by pull request, but your operations tasks are also fully reproducible through Git.

5.ความสอดคล้องและมาตรฐาน

เนื่องจาก GitOps มีรุ่นเดียวสำหรับการสร้างโครงสร้างพื้นฐานแอปและ Kubernetes การเปลี่ยนแปลงเพิ่มเติมคุณจึงมีเวิร์กโฟลว์แบบครบวงจรที่สอดคล้องกันทั่วทั้งองค์กรของคุณ ไม่เพียง แต่การรวมอย่างต่อเนื่องและการวางท่ออย่างต่อเนื่องของคุณทั้งหมดขับเคลื่อนด้วยคำขอแบบดึง แต่งานการดำเนินการของคุณนั้นสามารถทำซ้ำได้อย่างสมบูรณ์ผ่าน Git

1. **Stronger Security Guarantees**  
   Git’s strong correctness and security guarantees, backed by the strong cryptography used to track and manage changes, as well as the ability to sign changes to prove authorship and origin is key to a secure definition of the desired state of the cluster.

6.รับประกันความปลอดภัยที่แข็งแกร่ง

ความถูกต้องและการรับประกันความปลอดภัยที่แข็งแกร่งของ Git ซึ่งได้รับการสนับสนุนจากวิทยาการเข้ารหัสลับที่แข็งแกร่งที่ใช้ในการติดตามและจัดการการเปลี่ยนแปลงตลอดจนความสามารถในการเซ็นสัญญาการเปลี่ยนแปลงเพื่อพิสูจน์การประพันธ์และต้นกำเนิดนั้นเป็นกุญแจสำคัญ

https://www.weave.works/technologies/gitops/