EVOLVEN

EVOLVEN FOR KUBERNETES

Overview

Evolven Change Analytics for Kubernetes allows Kubernetes users to automatically detect, correlate and prioritize actual changes and differences in master components, node components, add-ons, and configuration of containers orchestrated by Kubernetes. With Evolven, application teams, DevOps and Kubernetes operations teams can accelerate troubleshooting and problem investigation, while avoiding performance and availability issues caused by application and platform changes, ensuring consistency of configuration across Kubernetes clusters and automatically maintaining a granular, prioritized audit trail of any updates and deployments in their environment.

How It Helps

Identify changes that are the root cause of an investigated issue.

Evolven lists granular changes in applications running in pods, containers, Kubernetes master components, node components, and add-ons that happened prior to an incident prioritized by their probability to cause an issue. The range of changes can be selected manually (e.g. show me changes that happened in a specific environment for the last 24 hours), or changes can be automatically correlated with an incident ticket in a service desk, or with an alert or event detected by a monitoring tool.

Rapidly provide visibility into the key configurations.

Evolven visualizes key configuration parameters, while the list of such parameters is defined out of the box. The list can be customized - additional parameters can be added, or parameters perceived as less critical can be removed. Examples of such key parameters include:

- Current version of the docker image used to instantiate running
- Restart count
- Config maps parameters
- Related ServiceAccounts
- Secrets
- Memory/CPUs values of replica set

Proactively detect high risk changes.

Evolven automatically assesses the risk of a change by its probability to cause a future issue- as soon as the change is detected. Evolven continuously adjusts this risk based on the new data collected, time passed, observed behavior of applications, and underlying platform where the change was detected.

Verify Kubernetes cluster consistency.

Evolven compares configuration, content, and full inventory of Kubernetes clusters within and across environments ensuring alignment and consistency of the clusters.

Verify configuration baselines.

Evolven provides a set of out-of-the-box configuration policies based on common frameworks, for example, CIS Benchmark. Collected Kubernetes configuration and detected changes are continuously verified against the rules defined in the policies to identify and report any deviations. Evolven also provides a powerful and flexible engine allowing users to define their custom policies.

Record historical state of an environment for problem investigation.

Tracking actual changes in dynamic Kubernetes environments, Evolven can present a detailed historical state of these environments at any point in time. Visibility into historical configuration of the environment at the point of an incident and audit trail of changes leading to this state allow an incident to be easily reproduced and efficiently search for problem root causes that triggered the incident.

Checks and balances for automated deployments.

Automated deployments eliminate the risk of “fat finger” type mistakes. However, deployment packaging, logic, and scope are still implemented and managed by humans. Even deployment execution can be affected by environment conditions and active workload. Detecting and validating actual changes created by an automated deployment, Evolven provides checks and balances, closing the automation loop.

How It Works

Collection

Evolven collector is deployed outside of Kubernetes clusters on a host that should have access to container image repository. Another requirement for this host is to have a container platform used by Kubernetes deployed (e.g. Docker). Evolven connects to Kubernetes using its API to extract configuration of the entire environment from application configuration to Kubernetes infrastructure configuration. The collector periodically scans Kubernetes configuration looking for changes. Application changes are detected periodically inspecting retrieved container images used to instantiate applications. All the detected changes are reported to a central Evolven Change Analytics server that either can be deployed in the customer's network or Evolven software-as-a-service (SaaS) server instances. In addition, when relevant, Evolven agents can be locally deployed on each of the hosts running Kubernetes cluster to collect configuration and track changes in the underlying infrastructure including operating system, hypervisor any virtualization or cloud infrastructure used, network infrastructure, storage etc.

The information collected in each of the environment layers is at its most granular form including: individual configuration parameters, lowest database schema elements, application master data values, application files checksums and versions.

Data Blending

Evolven extracts valuable insights directly from the collected data (what changed, what's different, anomalous and irregular changes, undesired inconsistencies etc.) Further, Evolven can provide additional types of insights using existing operational data collected by other IT solutions (e.g. are there unauthorized changes, which changes were done manually, which changes caused a monitoring alert etc.)

Evolven imports two key types of external data:

- Symptoms that indicate undesirable conditions in system behavior and health. Such symptoms could be Application Performance Management (APM) alerts, system alerts, network alerts, log-based alerts/KPIs etc.
- IT Context that describe the activities carried out or planned by IT. This includes, for example, infra and application changes coming from Deployment Automation and change requests from Service Desk

Analytics

Evolven turns data collected by Evolven and imported data into actionable insights. First, Evolven consolidates, cleans and correlates all the data. Then it assesses risk of changes over time and differences across environments. Evolven calculates risk and probability for each change/difference dimension and blended data sources, by applying:

- Machine Learning based algorithms (time anomaly detection, loneliness assessment, change frequency etc.)
- Out-of-the-box and custom knowledge base and SME input to fine tune analysis
- Heuristic algorithms (e.g. change consistency assessment)

Policies

Evolven provides a powerful policies engine allowing users to define desired configuration states. Each policy includes a rule or a set of rules evaluating a configuration parameter or a configuration element. For example, is memory allocated to a container within a target range, or does a docker image contain a specifically requested software component? The rules can be easily defined and managed in Evolven UI or using a command line interface (CLI). The CLI allows embedding policy-based verification into CI/CD pipelines to ensure that updated and deployed configurations match the desired configuration state. The policies will enable the organization to encode the knowledge of their environment for risk assessment in addition to automated Evolven analytics.

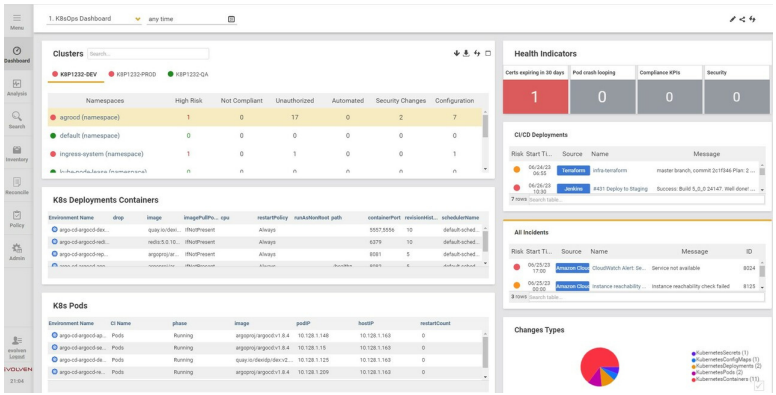

Figure 1: Evolven for Kubernetes

To find out more visit www.evolven.com and follow updates on LinkedIn and Twitter.