SOLUTION OVERVIEW

DEVSECOPS: ENABLING SECURITY FOR CLOUD OPERATIONS

CloudPassage

Today, every business is in the software business. New business opportunities, cost reduction, increased market penetration and revenues, are all predicated on new technology—specifically software technology. Having the hot new app, the most efficient supply chain, the deepest customer analytics software solutions sit at the top of any CEO's wish list. Development organizations have embraced DevOps, Continuous Integration/ Continuous Development (CI/CD), agile, and other methodologies designed to scope, develop, test, and deploy business critical software in the least amount of time.

Toward that end, rather than developing monolithic applications with long release cycles they now focus on three primary delivery methods (often employing all three depending on the project):

- **Release often** – deliver new features and functions in days and weeks not months and years. This entails careful scoping of projects with appropriate design, development, and testing teams ready to execute sequentially.
- **Release incrementally** – focus on delivering key features and enhancements in rapid and staged intervals vs full product releases.
- **CI/CD automation vs. handoffs** – continuously build and run new releases using a fully automated process—essentially a software assembly line for building software. CI/CD represents a full commitment to a DevOps methodology and at this time the highest level of automation you can employ in a software development lifecycle. You are essentially always developing and always releasing new versions of your application.

All of these build and delivery methodologies share these common attributes:

- Small teams utilizing service-oriented architectures (or microservices) to assemble their apps rather than code each component and function from scratch
- Build, QA and Release Automation (CI)
- Deployment Automation (CD)
- Hosted in public (or occasionally private) clouds

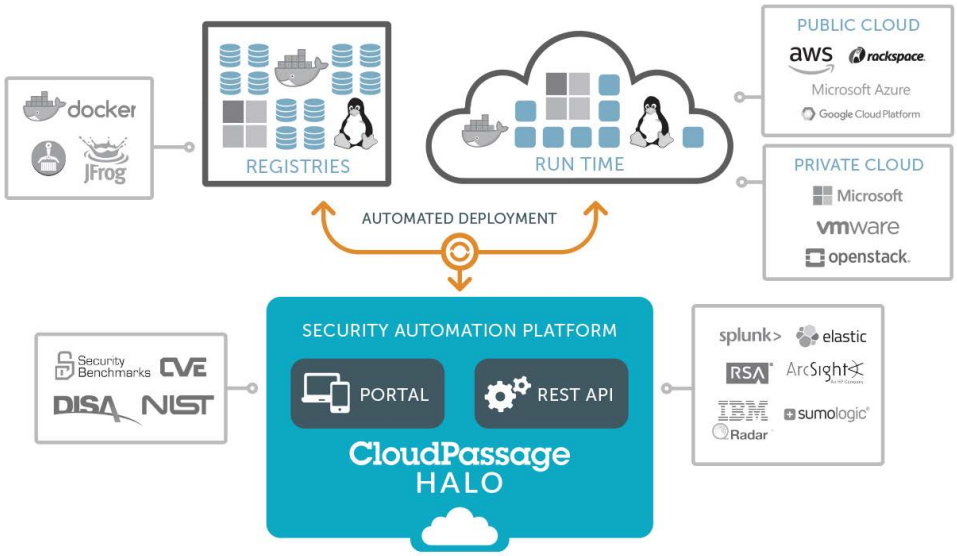
KEY SECURITY & COMPLIANCE CHALLENGES

The attack surface has expanded

Until now, the standard approach for protecting an enterprise's most prominent applications has been to host them in a siloed (virtualized) infrastructure with network perimeter security around it. This is a flawed approach even when applied to relatively static and predictable private data center environments. In the public cloud this approach is not enough. In a shared infrastructure, besides securing your perimeter, you have to secure each host and cloud asset for your application.

Compliance is even more complex

Because of the combination of a new and expanded attack surface, change rate inherent in the application development, deployment, and environment, as well as the lack of a dedicated and parameterized infrastructure, there are net-new compliance challenges that you will have to accommodate.



Security as part of the CI/CD pipeline

Unlike traditional deployment methodologies, in an automated CI/CD pipeline the only way to achieve application deployment in pre-production and push-to-production is through automation. As a result various security checks and assessments have to be completed before the application is marked as ready for production. However, most security tools, checks, and controls are simply not built to work prior to deployment. Most security checks have to be applied after an application is deployed and if it fails, sent back to development for remediation. In high velocity application delivery team this is the worst possible outcome—slowing down the entire process, adding cost and adding delays to the operation. In short, all the problems that DevOps & CI/CD methodologies are designed to overcome.

Security can't keep up

Unlike legacy applications, a cloud hosted application is designed for higher rates of change. The application scales up and down automatically based on load; new features and functionality is delivered weekly if not daily; and the application itself is typically deployed automatically. A manual security approach that is bolted on at the end of the deployment cycle, while appropriate for legacy applications in a data center, will not be able to keep up with modern applications running on modern infrastructure or the DevOps teams charged with their creation, care, and feeding.

THE 4 KEY STEPS FOR MASTERING THESE CHALLENGES

1. Shift left

Move your control implementation and security audits to be as early as possible in your development/deployment cycles. In a public cloud infrastructure it takes minutes, if not seconds for a poorly secured server to become compromised. Therefore, it is definitely best practice for your application nodes to be secured prior to deployment in production environments. You wouldn't deploy your application without completing your functional, integration, and user acceptance testing (UAT). Add security and compliance assessment to you test matrix early and prevent huge problems later.

2. Automate everything – especially security

Business and development teams are moving fast. They have automated deployments for the underlying infrastructure as well as the full application stack.

Building and testing these applications have also been automated. In this automated world, security testing, implementation, and monitoring, have to be automated as well. In the absence of security automation, businesses have to face a trade off between security vs time to market—which is not a trade off at all when you consider the risk and damage to a business's reputation and customer trust that a security breach can cause.

3. APIs and native integrations are not optional

Developers do not work with native user interfaces outside of their tool sets. If they can't automate and integrate a product into their stack, it's virtually invisible to them and it won't get used. Period. Your security strategy should work with a multitude of platforms, natively integrate with CI/CD tools, and secure application nodes packaged as both VMs and containers.

4. Continuously monitor for IoCs and policy violations

It's no longer appropriate to set policies, controls, and checks and then move on. In the cloud, you must continuously monitor your systems as vulnerabilities can be introduced and exploited in seconds with potentially disastrous results.

In the public cloud you must continuously scan for:

- New vulnerabilities announced since your servers were initially deployed
- Critical vulnerabilities that are still not patched
- Configuration policy violations
- File system integrity violations
- Monitor load balancer logs, applications logs, database transaction logs, network flow logs for any unwanted or non-compliant activity
- Unauthorized access to root level accounts

HALO: DEVSECOPS DONE THE RIGHT WAY

Build security

As builds are completed and go through their functional test cycle, Halo incorporates a comprehensive set of security checks to make sure that build artifacts meet the security and compliance policies of the organization. These CI/CD integrated checks include:

- Testing for any known vulnerable packages
- Secure configuration monitoring

QA / Stage (Pre-production)

Halo will continue to monitor your servers and containers for any new vulnerabilities. Additionally, Halo will also monitor your systems for the following policy violations:

- Configuration drift or deviations
- PII or sensitive personal information
- SPI data in any of the log files

Production

In addition to the checks implemented in the previous stages of the pipeline, Halo also incorporates checks for various IoCs in your production systems. These include:

- File system integrity
- Investigating and auditing various log files for events of interest.
- Privileged access auditing and monitoring
- Network traffic monitoring

WHAT'S UNIQUE ABOUT HALO'S SOLUTION?

- **Halo was built for DevSecOps.** Halo offers one of the industry's most complete REST API enabling enterprise security, IT, and DevOps teams to seamlessly integrate security into their DevOps processes, CI/CD toolchains, and infrastructure

automation solutions. Halo also features a Python SDK that acts as a wrapper to the REST API. It handles authentication, pagination and decreased snowflakes by promoting reusable code. The API itself is fully bi-directional. It can use data from Halo to integrate with third-party products (e.g. open a ticket in Jira or ServiceNow, export data to common SIEMs or create an Ansible playbook to remediate vulnerable packages). Halo is also used by DevOps teams to automate configuration security monitoring scans in their build pipeline. Finally, the API can pull data into Halo such as security policies, e.g., a file integrity or configuration security policy.

- Infrastructureagnostic – Halo supports the full cloud application stack. This is critical as today's modern cloud applications are assembled from multiple services—orchestrated and managed by multiple services running in a multiplicity of environments. Halo delivers visibility across all levels of the stack. More importantly Halo is the only unified cloud security platform on the market that can be used for both virtualized and container/microservices environments. As containers become an increasingly popular way to develop, package, and deploy applications, proper security hygiene is as critical for those environments as it is for virtualized ones. While many vendors can claim to support different public cloud vendors such as AWS or Azure, only Halo can deliver visibility and control across the entire cloud application stack—including containers and microservices.
- Comprehensive security capabilities – while many vendors claim to provide a full stack of security monitoring and alerting tools, only Halo delivers fully integrated functionality across the development and deployment stack. Rather than purchasing and integrating multiple modules Halo delivers single agent/single console visibility to the critical security functionality that enables you detect, protect, and remediate threats and vulnerabilities to your cloud infrastructure at speed and at scale. More importantly Halo enables you to incorporate these capabilities as a native part of your CI/CD tool chains.

ABOUT CLOUDPASSAGE

Founded in 2010, CloudPassage® was the first company to obtain a U.S. patent for universal cloud infrastructure security and has continued to innovate cloud security automation and compliance monitoring for application development and deployment. CloudPassage Halo® is an award-winning cloud security and compliance automation platform, delivered as a service, that provides universal visibility and continuous protection for multi-cloud environments. Halo deploys in minutes and scales effortlessly. The platform integrates with automation and orchestration tools such as Puppet and Chef, as well as CI/CD tools such as Jenkins. Today, CloudPassage Halo secures the infrastructure of leading global finance, insurance, media, ecommerce, high-tech service providers, transportation and hospitality companies.

www.cloudpassage.com | 800.838.4098

CloudPassage