SOLUTION OVERVIEW

CLOUD WORKLOAD SECURITY (CWS): PROTECT SERVERS AND CLOUD-BASED WORKLOADS FROM ATTACK

Automated workload visibility & security – at DevOps speed & scale.
THE PROBLEM

Organizations of all stripes are moving applications into public and private cloud environments. In some cases, these applications are brand new "cloud native" applications. In other cases, the applications have migrated from a traditional on-premises data center. Cloud environments are quite different from traditional data centers, and each difference has an implication for how you approach securing your cloud workloads. Here are some of the most important differences:

- **Architecture.** Cloud applications are designed around a microservices service-oriented architecture, which means each workload is just one configuration mistake away from open Internet attack. The traditional security practice of hardening the perimeter and leaving the inside "soft and chewy" is an invitation to disaster.

- **Speed.** Cloud workloads spin up and down very quickly. The average lifespan of a container is just 2.5 days, and of a traditional cloud instance 23 days (source: Datadog). A traditional security system that is based on periodic (e.g. monthly) scans and/or manual deployment processes is no longer sufficient.

- **Change.** Cloud hosted applications are likely to have a high rate of change. Practices such as DevOps, immutable infrastructure, and automation tools are all predicated on this. Security controls that are manually installed at the end of the deployment cycle might have worked fine for the legacy apps in your data center, but this is not sustainable in modern infrastructure.

- **Automation.** In modern IT environments, manual processes are too slow, costly, and error prone. Automating everything is the order of the day. Enterprises that have embraced cloud environments are also likely using automation tools, DevOps, and containers. A security system that cannot be automated or does not integrate with these systems is holding you back and leaving you exposed.

- **VMs and containers.** If your DevOps teams are not already using containers, they soon will be. Containers require a different set of security controls than servers, but purchasing a point-product that only secures containers adds yet another tool to manage and maintain.

THE SOLUTION: CLOUDPASSAGE HALO

CloudPassage® Halo® solves all of these challenges. Halo is an automated security platform that has been purpose-built for modern cloud environments and DevOps methodologies. Halo provides continuous visibility and protection for servers, VMs, cloud operating systems, and containers. Because Halo is delivered as a service, it’s fast to deploy, fully automated, and works at any scale.

Halo’s built-in security functions let you:

**Prevent attacks** by hardening your workloads. Halo includes a broad range of security functions including:

- Software vulnerability assessment for Linux and Windows hosts and containers. Halo detects known software vulnerabilities based on the National Institute of Standards and Technology (NIST) database of Common Vulnerabilities and Exposures (CVE).
- Secure configuration management for Linux and Windows hosts and containers. Halo assesses the configuration of your workloads by comparing them to standard benchmarks from the Internet Security (CIS) Benchmarks and Defense Information Systems Agency (DISA). You can also develop your own custom configuration checks or modify the templates using Halo’s built-in policy editor.
- Secure configuration management for Linux and Windows hosts and containers. Halo assesses the configuration of your workloads by comparing them to standard benchmarks from the Internet Security (CIS) Benchmarks and Defense Information Systems Agency (DISA). You can also develop your own custom configuration checks or modify the templates using Halo’s built-in policy editor.
- Host access monitoring for Linux and Windows provides a complete inventory of local user accounts and groups on each workload.
- Detection of secrets embedded in images and containers.
Detect policy violations and suspicious activities. Key violations that Halo detects include:

- Newly announced vulnerabilities or indications of compromise
- Configuration drift
- Unauthorized processes or ports
- Changes to server accounts or escalation of privileges
- Changes to files, directories, and registry keys.
- Suspicious activity in important server log files
- Suspicious network connections and traffic flow

Respond to incidents

- Halo’s bi-directional API lets you automate security workflows and implement a closed-loop security model including remediation via systems such as Puppet.
- Send security events and alerts into your existing SIEM or GRC system
- Quarantine workloads that have indications of compromise.

HOW HALO IS DIFFERENT

Comprehensive. Halo provides security and compliance for any mix of servers, virtual machines, cloud instances, Docker containers and images—all in one system. This directly leads to lower management overhead and lower costs.

Easy to deploy. Halo is a SaaS-based system with no hardware or software to buy or maintain. Halo micro-agents can be deployed automatically via scripts or popular orchestration tools such as Puppet, Chef, Ansible, Jenkins, Kubernetes, Docker Swarm, Mesos.

Works anywhere. Halo works in any data center, public cloud, hybrid environment, or multi-cloud environment.

High-frequency data collection. Halo monitors workloads every 60 seconds, letting you quickly identify new vulnerabilities and giving you visibility to even short-lived (ephemeral) workloads.

On-demand scalability. Grow from 100 to 10,000 workloads. Expand to new VPC environments. Halo’s SaaS architecture is available and scales when you need it.
Shift security left. Halo lets you assess workloads and images early in your deployment cycle and across your registries, providing fast feedback to developers and letting you be sure that hosts, images and containers are secure before you deploy them.

Easy integration. Halo contains a robust RESTful API and a well documented SDK which let you easily integrate Halo with your CI/CD toolchains, infrastructure automation technologies, GRC, and analytics platforms.