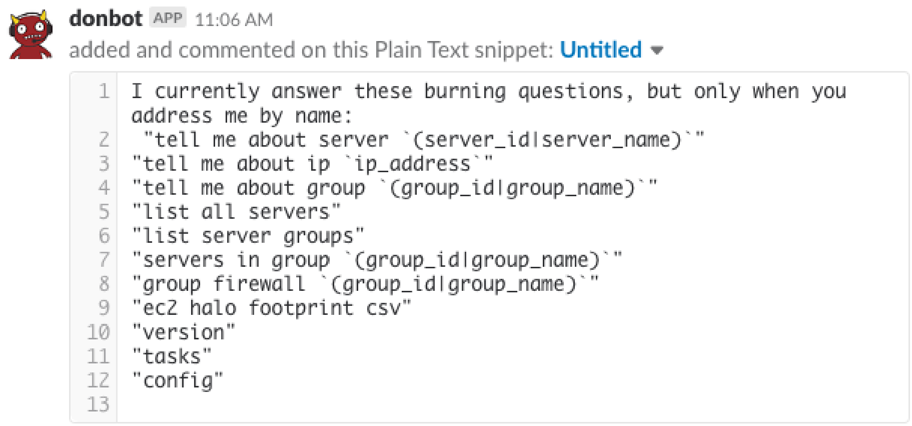
CloudPassage® Halo® Cortex: Integrations made easy

CloudPassage Halo Cortex dramatically reduces the time-to-value for your CloudPassage Halo adoption and DevSecOps transformation. Halo Cortex combines some of the most common integrations for the Halo platform in one easy-to-implement package, and gives you the ability to interact with Halo using Slack.

Halo Cortex (https://github.com/cloudpassage/don-bot) is designed for deep integration into your DevSecOps toolchain. Many of the included tools are delivered as Docker containers and can be transitioned, piece by piece, into an existing DevSecOps toolset. Halo Cortex accelerates adoption for Halo, makes Halo easier to interact with, and facilitates true security automation.

Halo Cortex can help you to rapidly realize your goals of security automation, and you can use it for an unlimited duration. Our recommendation is to use Halo Cortex as a set of training wheels to get you started with security



donbot APP 11:06 AM

added and commented on this Plain Text snippet: Untitled ▾

```
I currently answer these burning questions, but only when you address me by name:
 "tell me about server `(server_id|server_name)`"
"tell me about ip `ip_address`"
"tell me about group `(group_id|group_name)`"
"list all servers"
"list server groups"
"servers in group `(group_id|group_name)`"
"group firewall `(group_id|group_name)`"
"ec2 halo footprint csv"
"version"
"tasks"
"config"

```

automation. Then your team will likely absorb components from Halo Cortex into your existing automation processes. You can also use Halo Cortex as it is and eventually extend its functionality, and your team can customize it to meet specific needs within your environment. No matter your intended use of Halo Cortex, there is documentation to guide you through the process of adding functionality.

Components:

- Donbot (https://github.com/cloudpassage/don-bot): Allows users to interact with Halo within the Slack application. Donbot supports a number of different interactions, including searching for specific servers, describing server attributes (issues, configuration, EC2 metadata, and events associated with the server), as well as listing and describing server group configuration. Donbot will also automatically alert Slack users of critical Halo events. Donbot brings high-quality information to DevSecOps personnel in the most low-friction way possible. There's no need to buy another monitor to put on the wall or open yet another browser window, just switch to the #halo channel and ask Donbot your burning questions.
- Halo-Celery (https://github.com/cloudpassage/halocelery): Asynchronous task manager. Allows multiple users (via Donbot) to interact simultaneously with Halo and prevents long-running queries and reports from interrupting or delaying user interaction. This can be re-implemented (and even extended) independently from the rest of Halo Cortex.
- Firewall-Graph (https://github.com/cloudpassage-community/firewall-graph): Runs on-demand, and generates a graphical representation of a specific Halo group's firewall policy. This tool can be used independently from the rest of Halo Cortex.
- Scans-to-S3 (https://github.com/cloudpassage/halo-scans-archiver): Runs daily and ships all scan data, from all Halo modules, to S3 for long-term storage or further analysis. Use this to meet long-term compliance needs or to feed a data lake for deep analysis of historical security information.
- Events-to-S3 (https://github.com/cloudpassage/halo-events-archiver): Runs daily and ships all events, from all modules, from your Halo account to AWS S3 for long-term storage or further analysis. Like Scans-to-S3, this tool can be useful for facilitating deep analysis of historical security data.

- **EC2-Halo-Delta (https://github.com/cloudpassage/ec2-halo-delta):** Runs on-demand, creates a CSV file containing a list of all EC2 instances across all of your AWS accounts that are not protected by CloudPassage Halo. This is delivered as a container image that's invoked via interaction with Donbot, and easily can be repurposed and run as a daily task from your existing CI tool. The container has additional built-in functionality that allows CSVs to be delivered to specific Slack groups based on the unprotected instances' attributes, like AWS account membership, VPC location, regional location, and SSH key used in provisioning.

Halo Cortex is delivered as a docker-compose application, which facilitates simple and rapid deployment. Components in Halo Cortex are designed to be lightweight, and they are not stateful. Inbound open ports are not required for normal operation, only for troubleshooting and CLI access to Donbot.

Halo Cortex is available as community-supported, open-source software under the BSD license. CloudPassage offers support for Halo Cortex alongside CloudPassage Halo. Contact your CloudPassage sales representative for more information.

Use case scenarios:

You want to be alerted if a root login is ever attempted on any instance within a business-critical application.

- Assign a Log-Based Intrusion Detection (LIDS) policy to the group of business-critical servers in Halo, to identify and surface root login events.
- Whenever a root login occurs within the group of business-critical servers, Donbot will report the event in the #halo slack channel.

A root login has been reported by Donbot in the #halo channel. You need to know the AWS account that owns the instance, the AWS instance ID, the instance's current security issues, and the most recent events associated with the instance.

- Get the host name from the event reported in the #halo channel. For the sake of argument, let's say the host name is ip-172.16.22.1
- In slack, type: donbot tell me about server 'ip-172.16.22.1'
- Donbot reports back in-channel with the server's Halo and EC2 instance metadata, issues, and associated events.

It’s Friday afternoon and the application development team just performed a deploy to production. You notice that your active host count in Halo just dropped substantially, and this release was a feature release, not a performance refactor. How do you find out what production instances aren’t protected?

- Go to the #halo channel in Slack and type: donbot ec2 halo footprint csv
- A moment later, Donbot uploads a CSV file containing a list of AWS EC2 instances that are not protected by Halo, including the account ID, AWS region, VPC ID, instance ID, launch time, and the SSH key used for accessing the instance.
- You create a ticket for DevOps to install the Halo agent on all instances in the attached CSV file, and this deployment problem didn’t make you late for happy hour.

ABOUT CLOUDPASSAGE

Founded in 2010, CloudPassage® was the first company to obtain a U.S. patent for universal cloud infrastructure security and has been a leading innovator in cloud security automation and compliance monitoring for high-performance application development and deployment environments. CloudPassage Halo® is an award-winning workload security automation platform that provides universal visibility and continuous protection for servers in any combination of data centers, private/public clouds and containers. The Halo platform is delivered as a service, so it deploys in minutes and scales effortlessly. Fully integrated with popular infrastructure automation and orchestration tools such as Puppet and Chef, as well as leading CI/CD tools such as Jenkins, Halo secures the enterprise where it’s most vulnerable—application development and workload deployment. Today, CloudPassage Halo secures the critical infrastructure of many of the leading global finance, insurance, media, ecommerce, high-tech service providers, transportation and hospitality companies.

CloudPassage

www.cloudpassage.com | 800.215.7404