How to secure your Docker with CIS Benchmarks?

This?InSpec?compliance profile implement the?CIS Docker 1.13.0 Benchmark?in an automated way to provide security best-practice tests around Docker daemon and containers in a production environment.

InSpec is an open-source run-time framework and rule language used to specify compliance, security, and policy requirements for testing any node in your infrastructure.

Look how easy it is to use:

- inspec exec https://github.com/dev-sec/cis-docker-benchmark

Features

We use a yml attribute file to steer the configuration, the following options are available:

- trusted_user: vagrant?define trusted user to control Docker daemon.
- authorization_plugin: authz-broker?define authorization plugin to manage access to Docker daemon.
- log_driver: syslog?define preferable way to store logs.
- log_opts: /syslog-address/?define Docker daemon log-opts.
- registry_cert_path: /etc/docker/certs.d?directory contains various Docker registry directories.
- registry_name: /etc/docker/certs.d/registry_hostname:port?directory contain certificate certain Docker registry.
- registry_ca_file: /etc/docker/certs.d/registry_hostname:port/ca.crt?certificate file for a certain Docker registry certificate files.
- container_user: vagrant?define user within containers.
- app_armor_profile: docker-default?define apparmor profile for Docker containers.
- selinux_profile: /label\:level\:s0-s0\:c1023/?define SELinux profile for Docker containers.
- container_capadd: null?define needed capabilities for containers. example:?container_capadd: NET_ADMIN, SYS_ADMIN
- managable_container_number: 25?keep number of containers on a host to a manageable total.
- daemon_tlscacert : /etc/docker/ssl/ca.pem?configure the
- daemon_tlscert: /etc/docker/ssl/server_cert.pem?configure the server certificate.
- daemon_tlskey: /etc/docker/ssl/server_key.pem?configure the server key.
- swarm_mode: inactive?configure the swarm mode.
- swarm_max_manager_nodes: 3?configure the maximum number of swarm leaders.
- swarm_port: 2377?configure the swarm port.
- benchmark_version?to execute also the old controls from previous benchmarks, e.g. set it to 1.12.0 to execute also the tests from cis-benchmark-1.12.0 (which is the default).

Installation

Install CIS Docker Benchmark - InSpec Profile by running:

InSpec makes it easy to run your tests wherever you need. More options listed here:?InSpec cli

```
# run profile locally
$ git clone https://github.com/dev-sec/cis-docker-benchmark
$ inspec exec cis-docker-benchmark

# run profile locally and directly from Github
$ inspec exec https://github.com/dev-sec/cis-docker-benchmark

# run profile on remote host via SSH
inspec exec cis-docker-benchmark -t ssh://user@hostname -i /path/to/key

# run profile on remote host via SSH with sudo
inspec exec cis-docker-benchmark -t ssh://user@hostname -i /path/to/key --sudo

# run profile on remote host via SSH with sudo and define attribute value
inspec exec cis-docker-benchmark --attrs sample_attributes.yml

# run profile direct from inspec supermarket
inspec supermarket exec dev-sec/cis-docker-benchmark -t ssh://user@hostname --key-files private_key --sudo
```

Run individual controls

In order to verify individual controls, just provide the control ids to InSpec:

```
inspec exec cis-docker-benchmark --controls 'cis-docker-benchmark-1.4 cis-docker-benchmark-1.5'
```

Contribute

- Issue Tracker: https://github.com/dev-sec/cis-docker-benchmark/issues
- Source Code: https://github.com/dev-sec/cis-docker-benchmark

Support

If you are having issues, please contact luc.ibata@autodesk.com.

License
