|  |
| --- |
| #!/bin/sh |
|  | # ------------------------------------------------------------------------------ |
|  | # Docker Bench for Security v1.0.0 |
|  | # |
|  | # Docker, Inc. (c) 2015 |
|  | # |
|  | # Checks for dozens of common best-practices around deploying Docker containers in production. |
|  | # Inspired by the CIS Docker 1.6 Benchmark: |
|  | # https://benchmarks.cisecurity.org/tools2/docker/CIS\_Docker\_1.6\_Benchmark\_v1.0.0.pdf |
|  | # |
|  | # ------------------------------------------------------------------------------ |
|  |  |
|  | # Load dependencies |
|  | . ./output\_lib.sh |
|  | . ./helper\_lib.sh |
|  |  |
|  | # Setup the paths |
|  | this\_path=$(abspath "$0") ## Path of this file including filenamel |
|  | myname=$(basename "${this\_path}") ## file name of this script. |
|  |  |
|  | export PATH=/bin:/sbin:/usr/bin:/usr/local/bin:/usr/sbin/ |
|  |  |
|  | # Check for required program(s) |
|  | req\_progs='awk docker grep netstat stat' |
|  | for p in $req\_progs; do |
|  | command -v "$p" >/dev/null 2>&1 || { printf "%s command not found.\n" "$p"; exit 1; } |
|  | done |
|  |  |
|  | # Ensure we can connect to docker daemon |
|  | docker ps -q >/dev/null 2>&1 |
|  | if [ $? -ne 0 ]; then |
|  | printf "Error connecting to docker daemon (does docker ps work?)\n" |
|  | exit 1 |
|  | fi |
|  |  |
|  | usage () { |
|  | cat <<EOF |
|  | usage: ${myname} [options] |
|  |  |
|  | -h optional Print this help message |
|  | -l PATH optional Log output in PATH |
|  | EOF |
|  | } |
|  |  |
|  | # Get the flags |
|  | # If you add an option here, please |
|  | # remember to update usage() above. |
|  | while getopts hl: args |
|  | do |
|  | case $args in |
|  | h) usage; exit 0 ;; |
|  | l) logger="$OPTARG" ;; |
|  | \*) usage; exit 1 ;; |
|  | esac |
|  | done |
|  |  |
|  | if [ -z "$logger" ]; then |
|  | logger="${myname}.log" |
|  | fi |
|  |  |
|  | yell "# ------------------------------------------------------------------------------ |
|  | # Docker Bench for Security v1.0.0 |
|  | # |
|  | # Docker, Inc. (c) 2015 |
|  | # |
|  | # Checks for dozens of common best-practices around deploying Docker containers in production. |
|  | # Inspired by the CIS Docker 1.6 Benchmark: |
|  | # https://benchmarks.cisecurity.org/tools2/docker/CIS\_Docker\_1.6\_Benchmark\_v1.0.0.pdf |
|  | # ------------------------------------------------------------------------------" |
|  |  |
|  | # Warn if not root |
|  | ID=$(id -u) |
|  | if [ "x$ID" != "x0" ]; then |
|  | warn "Some tests might require root to run" |
|  | sleep 3 |
|  | fi |
|  |  |
|  | logit "Initializing $(date)\n" |
|  |  |
|  | # Load all the tests from tests/ and run them |
|  | main () { |
|  | # List all running containers |
|  | containers=$(docker ps | sed '1d' | awk '{print $NF}') |
|  | # If there is a container with label docker\_bench\_security, memorize it: |
|  | benchcont="nil" |
|  | for c in $containers; do |
|  | labels=$(docker inspect --format '{{ .Config.Labels }}' "$c") |
|  | contains "$labels" "docker\_bench\_security" && benchcont="$c" |
|  | done |
|  | # List all running containers except docker-bench (use names to improve readability in logs) |
|  | containers=$(docker ps | sed '1d' | awk '{print $NF}' | grep -v "$benchcont") |
|  |  |
|  | for test in tests/\*.sh |
|  | do |
|  | . ./"$test" |
|  | done |
|  | } |
|  |  |
|  | main "$@" |