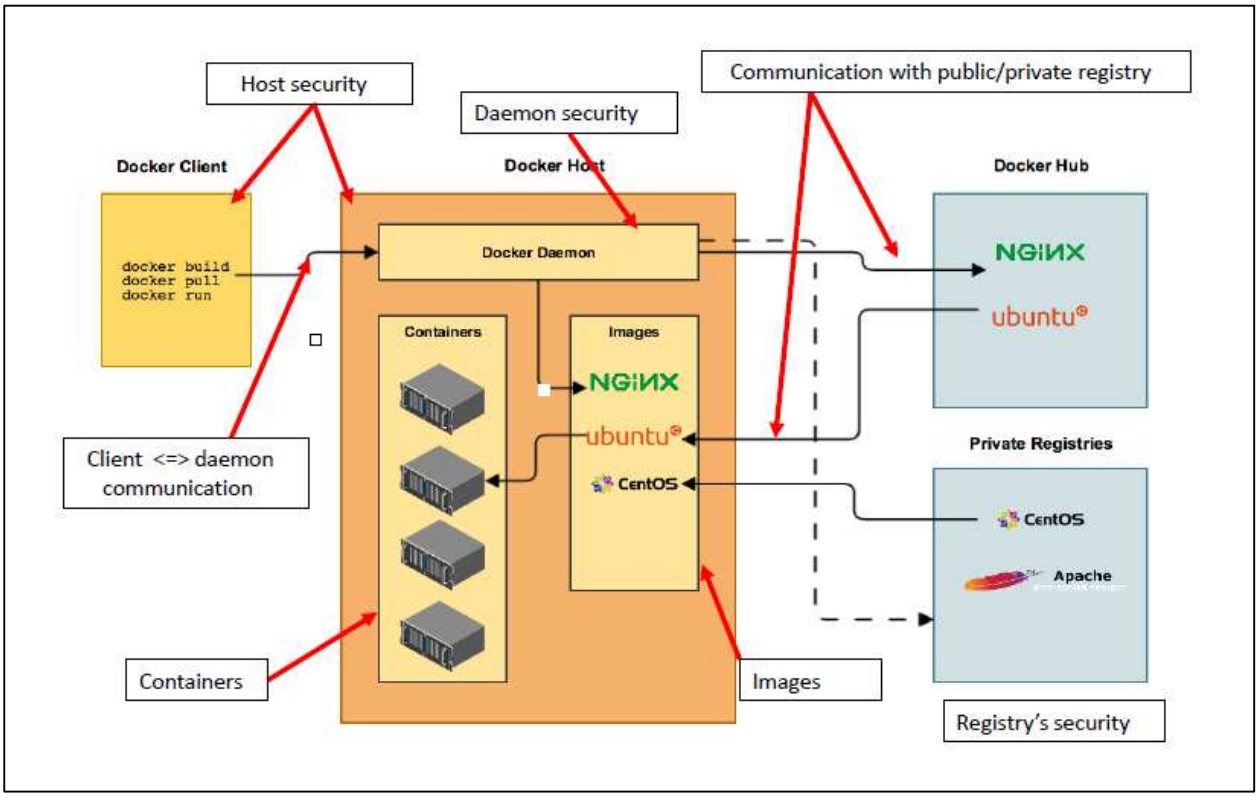
Security Analysis of Docker

Abstract: The concept of containerization was in Linux from ages in the form of jails, zones, LXC etc. but it is since 2 years it gained tremendous recognition. The credit goes to "Docker" which made the concept of containerization very useful and handy by adding many benefits to existing container technologies. Tech giants like Redhat, Google, IBM, VMware etc. are not only the biggest contributors to this most active open source project but also major users of it. Only Google spins up more than 2 billion containers per week, more than 3,300 containers per second. The effect of containers already impacted the virtual machine market and this impact is going to increase significantly in near future.

Security is always an important issue for any upcoming technology and Docker is no exception to it. This research aims at analyzing the vulnerabilities within Docker containers and Docker container environments basing on the pipeline approach. First individual components in the pipeline have been analyzed (Images, Container Runtime, Daemon, Hosts, Registry's) and then ecosystem/pipeline level security analysis has been done. Research also provides recommendations on how containers and container environments can be made secure.



Pipeline approach to Security Analysis of Docker

References:

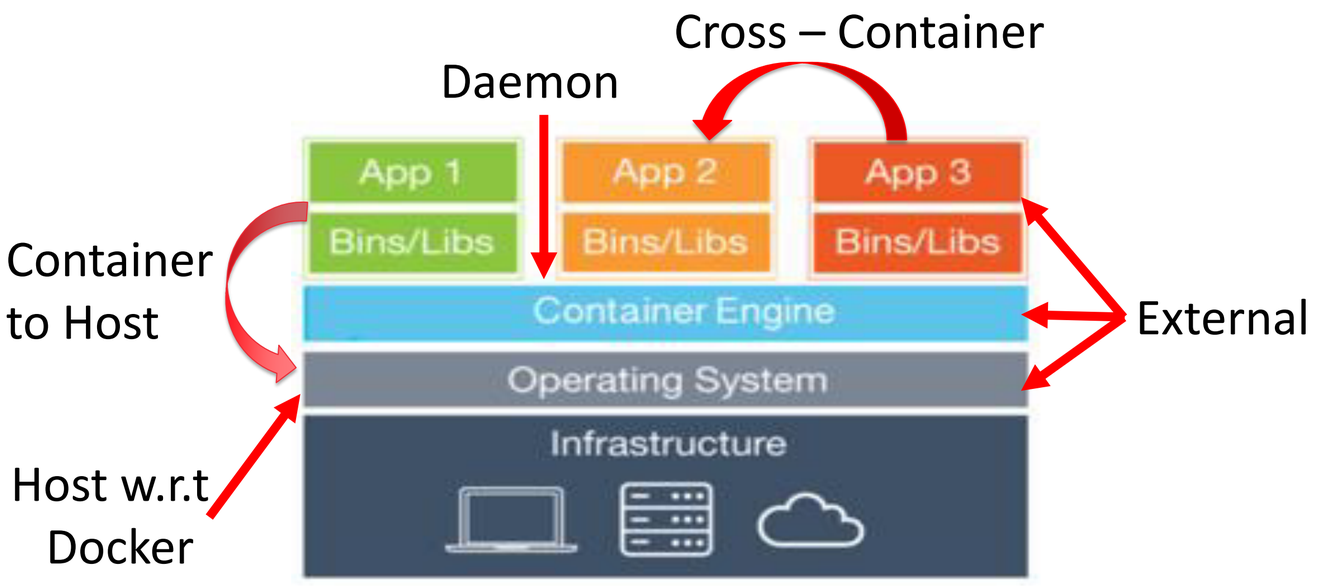
1. Breaking && Securing your 'Docker'ized environments by Manideep Konakandla @OWASP AppsecUSA'16
2. CIS Docker 1.12 benchmark by Manideep Konakandla et al.

RSAConference Security Scholar

Security analysis of Docker

Manideep Konakandla
Carnegie Mellon University

Problem Statement and Goals



Container specific issues

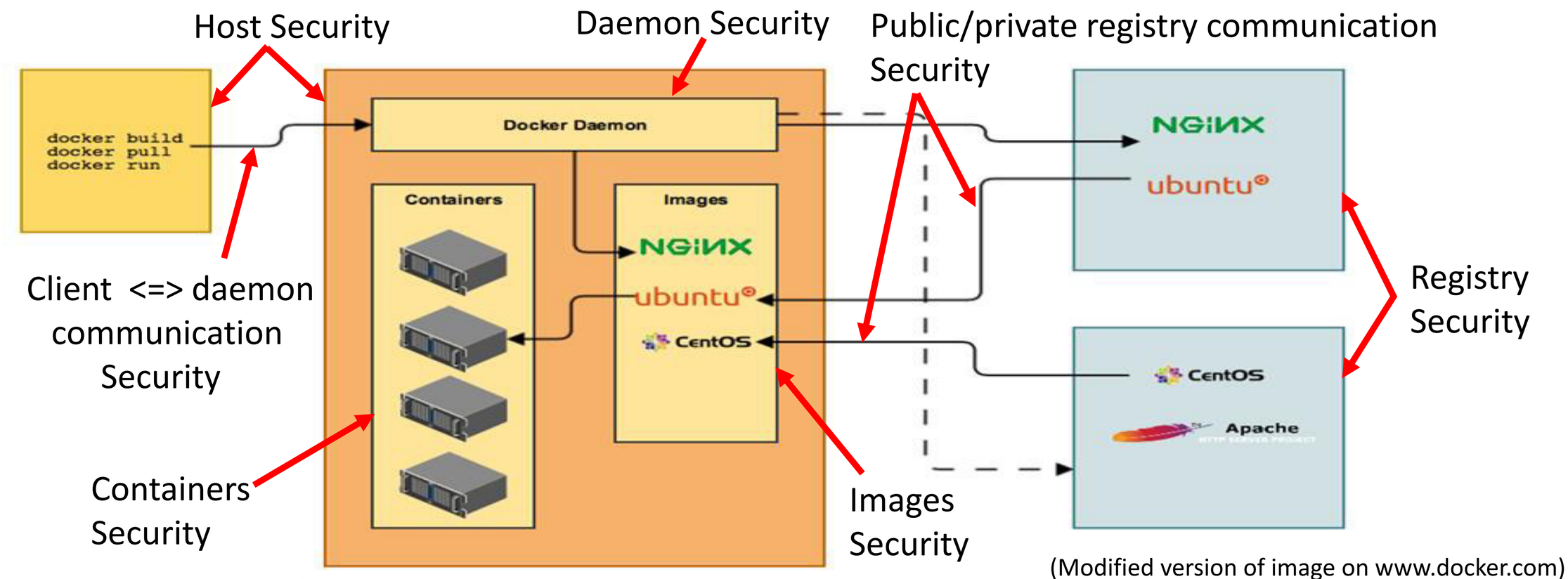
(This image excludes general security issues, issues specific to Docker Images)

Problem: Docker containers are light-weight alternative to virtual machines. Enterprises, home users are adopting it rapidly because of the advantages they offer but very little security analysis has been done so far on Docker containers & container environments

Goals:

1. Security analysis of Docker containers and container environments
2. Providing recommendations on securing containers and container environments

Approach



(Modified version of image on www.docker.com)

Pipeline Approach targeting above components

Results

- Docker claims that it is “Secure by Default”. My research proved that it has “Weak security defaults” (No memory limits, allowing inter-container communication, read-write mounts, running as root etc.). In short, **“Docker containers are not secure by default, we have to make them secure!”**
- Images
  a) “Docker hub” registry Images have critical vulnerabilities such as Heartbleed, Shellshock etc.
  b) Enterprises should build in-house registries by writing, building and managing images securely following 15+ secure rules from the references (follow version pinning mechanism for images & packages, use no-cache flag, binary level image scans, ripping of excess permissions of setuid & setgid, enabling Content Trust, using USER instruction etc.) to avoid security issues such as privilege escalations, MITM, RCE etc.
  c) Home users are encouraged to review, scan images before using with tools like Twistlock, Nautilus etc.
- Containers
  Do not use docker0 bridge (MITM), set memory limits (DOS) , bind incoming traffic (Info disclosure), beware of non namespaced components (key ring, kernel modules..), Docker commands caching issue etc.

Future Work: Hardware isolation to containers , migrating virtual machines security to containers (Encrypted images, FACE-CHANGE etc.)

References / For full results:

1. Breaking && Securing your ‘Docker’ized environments by Manideep Konakandla @OWASP AppsecUSA’16
2. CIS Docker 1.12 benchmark by Manideep Konakandla et al.