Aguascalientes Local Chapter

2nd Meeting
About – Chapter Leader

• Juan Gama
  – Application Security Engineer @ Aspect Security
  – 9+ years in Appsec, Testing, Development
  – Maintainer of OWASP Benchmark
  – I like GIFs!
Docker
What is Docker?

- "Docker is the world's leading software containerization platform"
What is a container?

• Consists of an entire runtime environment: an application, plus all its dependencies, libraries and other binaries, and configuration files needed to run it, bundled into one package.
Docker invented containers?
Docker vs LXC, Jails, Vagrant

• LXC runs in the host but has it's own section of RAM, CPU, disk, etc. Closer to a VM. Docker can be just one process, needs a volume.

• Vagrant is a script for VMs.
Docker vs Virtualization

• Virtualization includes an entire operating system as well as the application. Docker sits on top of the OS
Docker vs Virtualization
Docker vs Virtualization

[Diagram showing Docker engine, host operating system, and infrastructure layers with three applications labeled APP 1, APP 2, and APP 3, each containing BINS/LIBS.]
Why Docker?

• Solves dependency problems and the problem of ancient times:
  
• "It works on my machine!"
Docker Components

- Docker Engine
- Docker Hub
Docker Engine

• **Docker daemon**
  – Runs on the host machine

• **Docker Client**
  – CLI used to interact with the daemon

• **Windows and OSX**
  – docker-machine (small linux running the Docker daemon) - Needs Virtualbox
Docker Workflow Components

• Docker image
  – Has the env, your application, OS, dependencies,

• Docker Container
  – Created from images, start, stop, move, delete

• Docker Registry
  – Public and private repo to store images

• Dockerfile
  – Automates image construction
Docker

- Docker Container
- Docker Composer
- Docker Swarm
Demo

DEMO GODS

PLEASE LET THESE DEMOS WORK

meme-generator.net
Docker Security

• Quite secure.

• Namespaces for isolation: processes running within a container cannot see, and even less affect, processes running in another container, or in the host system.

• Each container also gets its own network stack.

• Control Groups for resource accounting and limiting, ensure that each container gets its fair share of memory, CPU, disk I/O; and, more importantly, that a single container cannot bring the system down by exhausting one of those resources.
Docker Security

• Only trusted users should be allowed to control your Docker daemon.

• “root” within a container has much less privileges than the real “root”. For instance, it is possible to:
  – deny all “mount” operations;
  – deny access to raw sockets (to prevent packet spoofing);
  – deny access to some filesystem operations, like creating new device nodes, changing the owner of files, or altering attributes (including the immutable flag);
  – deny module loading;
  – and many others.
Docker Security

• Additional: AppArmor, SELinux, GRSEC
• Run inside a VM
• Compromised images
• DOS
• https://www.docker.com/docker-security