## 使用Docker搭建Web漏洞测试环境

[MyKings](http://www.freebuf.com/author/MyKings" \o "由 MyKings 发布) 2016-12-27 共**92414**人围观 ，发现 **11** 个不明物体 [WEB安全](http://www.freebuf.com/articles/web)[工具](http://www.freebuf.com/sectool)

**\* 本文原创作者：MyKings，本文属FreeBuf原创奖励计划，未经许可禁止转载**

由于一直在做 Web 漏洞扫描器的开发, 那么就必然少不了 Web 的漏洞测试环境, 其中就包括 bWAPP、DVWA、OWASP WebGoat 等这些国际品牌。

这些漏洞环境一般搭建比较繁琐, 而且出问题后有不能像 git 那样方便的’回滚’操作, 当然你可以使用 esxi 来管理, 不过虚拟机仍然会存在定期快照、回滚操作较长等繁重操作。

**那有没有轻量级的能够快速搭建 Web 漏洞的测试环境呢? 答案是有的, 那就是 docker! 接下来我们就使用 docker 进行环境搭建。这里所涉及的项目已经创建在了 github([docker-vulnerability-environment](https://github.com/MyKings/docker-vulnerability-environment" \t "/home/zhang/Documents\\x/_blank))上。**

## 1、Docker 基本命令

这里不会过多去讲解 docker, 只会对常用命令进行简要说明。相关详细的使用请参考: [官方文档](https://docs.docker.com/" \t "/home/zhang/Documents\\x/_blank)。

使用 docker 前, 需要先安装 docker 的环境, 这里假设环境已安装好。

### 1.1 基本

docker info

$ docker info

docker stats

$ docker stats

docker version

$ docker version

### 1.2 镜像

docker search

$ docker search ubuntu

docker pull

$ docker pull ubuntu

docker images

$ docker images

docker rmi

$ docker rmi <镜像ID>

docker build

$ docker -t <镜像名称>

### 1.3 容器

docker run

$ docker run -it –name <容器名称> -p 0.0.0.0:8080:80 ubuntu:14.04 /bin/bash

docker start

$ docker start <容器名称>

docker stop

$ docker stop <容器名称>

docker ps

$ docker ps -a

docker inspect

$ docker inspect <容器名称>

docker rm

$ docker rm <容器名称>

## 2、Dockerfile 编写

### 2.1 FROM

定义容器的基础系统来自哪个 image, 如: FROM ubuntu:14.04, image 仓库为 ubuntu tag 为 14.04的系统。

### 2.2 MAINTAINER

作者相关信息说明, 一般书写方式为: MAINTAINER <作者名称> <作者邮箱地址>

### 2.3 ADD

从宿主机拷贝文件到容器内, 一般格式: ADD ./lib /home/lib, 将当前目录下的 lib 目录中的内容拷贝到容器的 /home/lib/目录下。

### 2.4 RUN

在容器内运行命令, 如: RUN apt-get -y update。

### 2.5 WORKDIR

切换容器内的当前工作目录, 如: WORKDIR /home/lib/。 当你登陆到容器内后,默认就进入 /home/lib目录。

### 2.6 EXPOSE

暴漏端口, 把容器端口暴漏给宿主机, 方便在宿主机访问, 如:EXPOSE 1000 。

### 2.7 CMD

执行一系列命令, 如: CMD ["sh", "./startup.sh"]。

### 2.8 Dockerfile 样例

github地址: [https://github.com/MyKings/docker-vulnerability-environment/blob/master/bWAPP/Dockerfile](https://github.com/MyKings/docker-vulnerability-environment/blob/master/bWAPP/Dockerfile" \t "/home/zhang/Documents\\x/_blank)

FROM ubuntu:14.04

MAINTAINER MyKings

# 使用国内淘宝源

ADD sources.list /etc/apt/

# 安装服务

RUN apt-get -y update

RUN apt-get -y install php5 php5-mysqlnd mysql-server wget unzip

# 启动 mysql 并设置 root 密码

RUN /etc/init.d/mysql start &&\

mysql -e "grant all privileges on \*.\* to 'root'@'localhost' identified by 'bug';"&&\

mysql -u root -pbug -e "show databases;"

# 切换工作目录

WORKDIR /var/www/html/

# 本地拷贝#ADD ./bWAPP\_latest.zip /var/www/html/bWAPP\_latest.zip#RUN unzip /var/www/html/bWAPP\_latest.zip

# 下载 bWAPP

RUN wget http://jaist.dl.sourceforge.net/project/bwapp/bWAPP/bWAPP\_latest.zip && unzip bWAPP\_latest.zip

# 删除默认首页

RUN rm /var/www/html/index.html

# 拷贝启动脚本ADD ./startup.sh ./

EXPOSE 80

CMD ["/bin/bash", "/var/www/html/startup.sh"]

## 3、bWAPP 环境搭建

### 3.1 克隆项目

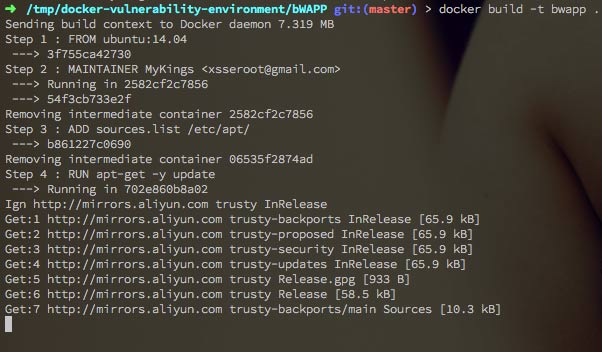
$ git clone git@github.com:MyKings/docker-vulnerability-environment.git# 进入到 bWAPP 目录

$ cd ./docker-vulnerability-environment/bWAPP

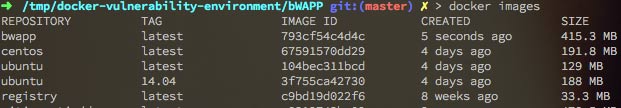
### 3.2 创建镜像

开始创建 image 镜像。

$ docker build -t bwapp .

[](http://image.3001.net/2016/12/b0da275520918e23dd615e2a747528f1.jpg)

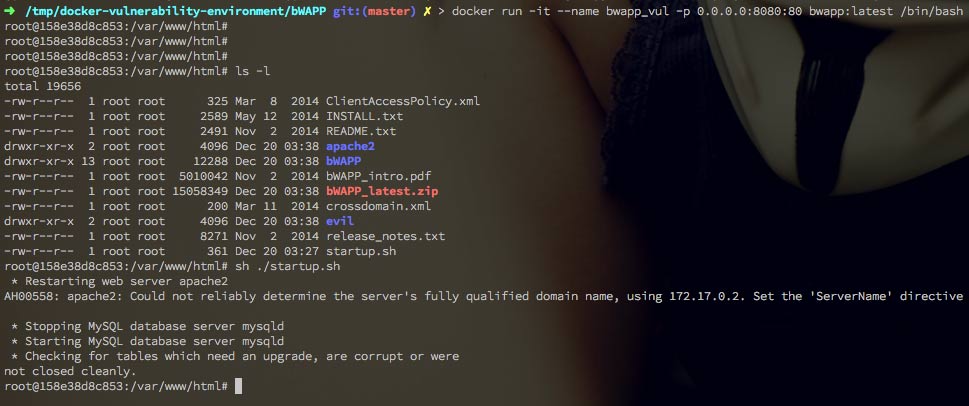
等待上面的指令操作完成, 使用 docker images 来查看镜像列表, 可以发现我们 build 好的 bwapp 镜像。

[](http://image.3001.net/2016/12/78805a221a988e79ef3f42d7c5bfd418.jpg)

### 3.3 创建容器

$ docker run -it --name bwapp\_vul -p 0.0.0.0:8080:80 bwapp:latest /bin/bash

创建容器并启动交互模式, 把容器的 80 端口映射到宿主机的 8080 端口上。 sh ./startup.sh 启动后端服务。

[](http://image.3001.net/2016/12/5f0b6ebc4bea10285ba2b8a6ce78b863.jpg)

### 3.4 开始测试

使用浏览器打开 <http://127.0.0.1:8080/bWAPP/> 地址开始你的 Hack 之旅吧!

[](http://image.3001.net/2016/12/54572388151da69dffdd4ace119a5540.jpg)

**\* 本文原创作者：MyKings，本文属FreeBuf原创奖励计划，未经许可禁止转载**