Name: Rui Shu

Hard Problem Area: Resilient Architecture

Overall Research Goal(s):
*What do you ultimately hope to show with your research? This section can involve some jargon, but try to relate these goals to your broad impact section. Start with the larger goal(s) and narrow the scope towards your specific study (and specific goals, below).*

In our previous work, we proposed a scalable Docker image vulnerability analysis (DIVA) framework for automatically discovering, downloading, and analyzing vulnerabilities in images from Docker Hub. DIVA also accesses vulnerability propagation between images. We use DIVA to analyze over 300,000 images and found significant and pervasive vulnerabilities in Docker Hub images. We also found strong correlations between top influential images and top ranked vulnerable packages, which implies that the widespread image vulnerabilities are likely the result of propagation from a small number of influential images.

These findings demonstrate a strong need for more automated and systematic methods of applying security updates/patches to Docker images. In the future, we plan to develop a scalable image security patching technique that ensure both correctness and efficiency. DIVA provides a foundation to meet the need.

Broad Impact:
*Why is your research important? This section should avoid any technical jargon and should be meaningful to the general public. Try to keep this down to five sentences. This should be hierarchical: the broad impact decomposed into more specific impacts that connect your overall research goals to your more specific goals.*

Docker containers have recently become a popular approach to provisioning multiple applications over shared physical hosts in a more lightweight fashion than traditional virtual machines. This popularity has led to the creation of Docker Hub registry, which distributes a large number of official and community images. In January 2015, one survey of enterprises indicated that security was a top concern when deciding whether to deploy containers, and it also found that of various security concerns, vulnerability and malware concerns were the greatest. Our work will endeavor to mitigate the security threats in Docker images. As a result, our research will help users run applications in secure container environment.

Specific Research Goals:
*Lay out the steps you are going to take to achieve your overall research goal. You can get technical here.*

Our specific research goals are as follows:

1) To propose an approach to dynamically patch security vulnerabilities in applications in the running container (images).

2) To ensure the correctness of the security patching. We need to design a way to test dynamic updates (e.g., a certain CVE is fixed in patched version), which means the
security patch will not bring side-effects or influence current running application in container. If the patching has a bad influence, we should find a way to address it.

3) To achieve a high efficiency (e.g., will not incur the downtime of current running application, and incur a low overhead to the host)

4) To make our security patching technique scalable, which means this technique can scale to a large number of vulnerable images (we may leverage the information that images may share layers or the parent-child relationship between images).

**Proposed Data Collection (if applicable):**

*What data will you collect to answer your research goals? How will you collect it? Will it be an observational study, randomized comparative experiment, or simulation study? Include potential biases and be prepared to explain how the data will achieve your specific research goals.*

In our previous work, we have leveraged Clair to detect vulnerabilities in each image. Clair is an open-source tool from CoreOS designed to identify known vulnerabilities in container images against CVE database. Clair has been primarily used to scan images in CoreOS’s private container registry, Quay.io, but it can also analyze Docker image.

We have collected vulnerabilities from 356,218 Docker images. These data will be the foundation of our further study. Additionally, we will collect several information from unpatched images and patched images (e.g., the list of vulnerabilities) We may also collect host resource usage to identify the overhead of our patching technique.

**Success Criteria:**

*How will you determine whether you satisfied your specific and overall research goals?*

The primary success criterion is the correctness of our security patching, i.e., the patch will fix vulnerabilities and will not disturb current running application that initialized from unpatched image.

In addition, we plan to reduce the overhead of patching to a low level, and make our technique be applicable to a large number of images.

**Anticipated Difficulties, Limitations, and Criticisms:**

*What will make the above specific research goals difficult to achieve? How do you plan on dealing with these difficulties if they arise?*

The major difficulties may arise from the following aspects:

1) How to dynamically patching an application image, since the type of application in image might be different (e.g., single process application or multi-thread application)

2) How to verify the success dynamic update, e.g., a set of test suite could be designed to test dynamic updates, we may analyzed the cases that patches fail.

3) How to reduce the patching overhead. Our technique is a dynamic security patching for running applications initialized from images, incurred overhead to both application and underlying host should be taken care of.

4) How to scale the patching technique to a large number of images. One thought is to identify similar characteristics between a group of images for batch patching, this may reduce the work for duplicated patching.