**Attack Scenario - Trojanized Container:**

We have implemented a python script that can inject a backdoor on the host machine by using a trojanized container. This backdoor when inputted with the attacker machine’s IP address gives out a reverse shell of the host machine’s container with root access on the attacker’s machine. The attack could become very dangerous as the host or the user might never know that his system is being accessed by an attacker remotely.

1. Implementation:

1.1. Dockerscan: For implementing this attack, we have used a tool called “Dockerscan” which is an analyzing and a modifying tool of Docker. This tool offers various features such as Scan, Registry, and Image actions. For this project, we have focused on image actions specifically.

1.1.1.  Image actions: The image actions contain internal features such as Analyze, Extract, Info and Modify. As we had to modify the image, we have used modify feature in which there is an object called “trojanize”. [[1]](#endnote-1)

1.2.  Trojanizing an image: We have now pulled an ubuntu latest image from the docker repository and exported the necessary variables required by the dockerscan to run. The image is modified by using the trojanized object and is placed on the IP address of the host system and is exposed on port 4444. This creates a .tar file which is saved in the local directory of the host system. The .jar file contains malicious files which are used to create backdoors on the host machine.

1.3.  Loading the Trojanized container: For this part, we have built the image and deployed the container using the .tar file onto the host system.

1.4.  Running the Trojanized container:  To run the container, we have written a python script that automates the attack. A Netcat listener is run on the attacker’s machine which listens to the port 4444 of the host machine and gives out a reverse shell of the host machine’s container with root access on the attacker’s machine.[[2]](#endnote-2)

1. Screenshots of working of the attack:
   1. Running the python script:

A picture containing text, monitor, screenshot, indoor

Description automatically generated

2.2 Getting the IP address of the attacker system:

The Host system is on the left side (Ubuntu 18.04 OS) and the attacker’s system is on the right side (Elementary OS)

A screenshot of a computer

Description automatically generated with medium confidence

2.3 Trojanizing the image and deploying the container:

A screenshot of a computer

Description automatically generated with medium confidence

2.4 Getting the reverse shell on the attacker’s machine:

After getting the reverse shell, we can access the /etc/shadow file of the host system on the attacker’s machine.

A screenshot of a computer

Description automatically generated with medium confidence

1. https://github.com/cr0hn/dockerscan [↑](#endnote-ref-1)
2. https://www.udemy.com/share/101t7qCUcZdl9VRXQ=/ [↑](#endnote-ref-2)