

LUIZ ZENHA

POSTECH

SOFTWARE ARCHITECTURE

DOCKERIZAÇÃO

AULA 04

SUMÁRIO

O QUE VEM POR AÍ?	3
HANDS ON	4
SAIBA MAIS.....	5
O QUE VOCÊ VIU NESTA AULA?	10
REFERÊNCIAS.....	11
PALAVRAS-CHAVE	12

O QUE VEM POR AÍ?

Vamos discutir sobre o Docker Hub, como utilizá-lo e os processos para realizar a publicação das imagens de forma pública no registry.

Veremos, também, como verificar a existência de possíveis problemas de vulnerabilidade nas imagens criadas ou mesmo em imagens que baixamos da internet.

EMANIP

HANDS ON

O Docker Hub permite que possamos realizar o upload das nossas imagens do Docker. Dessa maneira, nos próximos vídeos, vamos realizar um cadastro no Docker Hub e subir nossa primeira imagem para esse sistema.

EMANUELO

SAIBA MAIS

DOCKER HUB

O Docker Hub, como falamos anteriormente, é o registry fornecido pela Docker que permite aos desenvolvedores compartilhar e gerenciar imagens de contêiner Docker. Ele funciona como um repositório centralizado de imagens de container Docker, permitindo que os usuários compartilhem e baixem imagens facilmente.

Além de permitir o compartilhamento de imagens de container, o Docker Hub também fornece recursos para gerenciar essas imagens, incluindo recursos de controle de versão, gerenciamento de acesso e integração com ferramentas de integração contínua e entrega contínua (CI/CD).

O Docker Hub é amplamente utilizado pela comunidade de desenvolvimento de software, permitindo que os desenvolvedores publiquem suas próprias imagens personalizadas, compartilhem suas soluções e baixem imagens prontas para uso, economizando tempo e esforço na configuração de ambientes de desenvolvimento e produção.

Para poder publicar uma imagem e distribuí-la é necessário realizar o cadastro no site: <<https://hub.docker.com/signup>>, criando assim um usuário e senha.

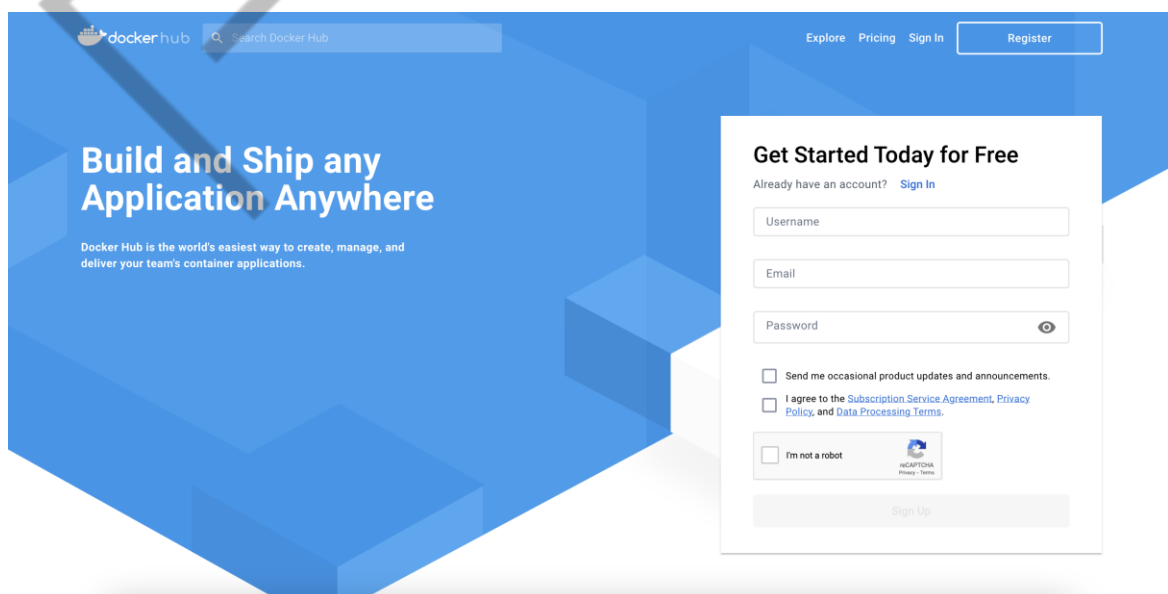


Figura 1 - Página do Docker Hub
 Fonte: <https://hub.docker.com/> (2023)

Na utilização de forma privada é possível evitar que outras pessoas encontrem sua imagem na busca e não tenham permissão de baixá-la. Para isso, é necessário assinar um dos pacotes do serviço.

O processo para publicar uma imagem necessita que cada imagem possua um identificador exclusivo conhecido como "tag", que é uma sequência de caracteres alfanuméricos separados por um caractere ":".

A tag é usada para identificar uma versão específica de uma imagem de contêiner. Por exemplo, uma imagem de contêiner pode ter várias tags, como "v1.0", "v1.1", "latest" etc. Cada tag representa uma versão específica da imagem e pode ser referenciada ao iniciar um contêiner a partir dessa imagem.

Ao atualizar uma imagem de contêiner, os desenvolvedores geralmente criam uma nova tag para a nova versão, mantendo as tags anteriores intactas. Isso permite que outros usuários continuem a usar as versões anteriores, enquanto os desenvolvedores trabalham na nova versão.

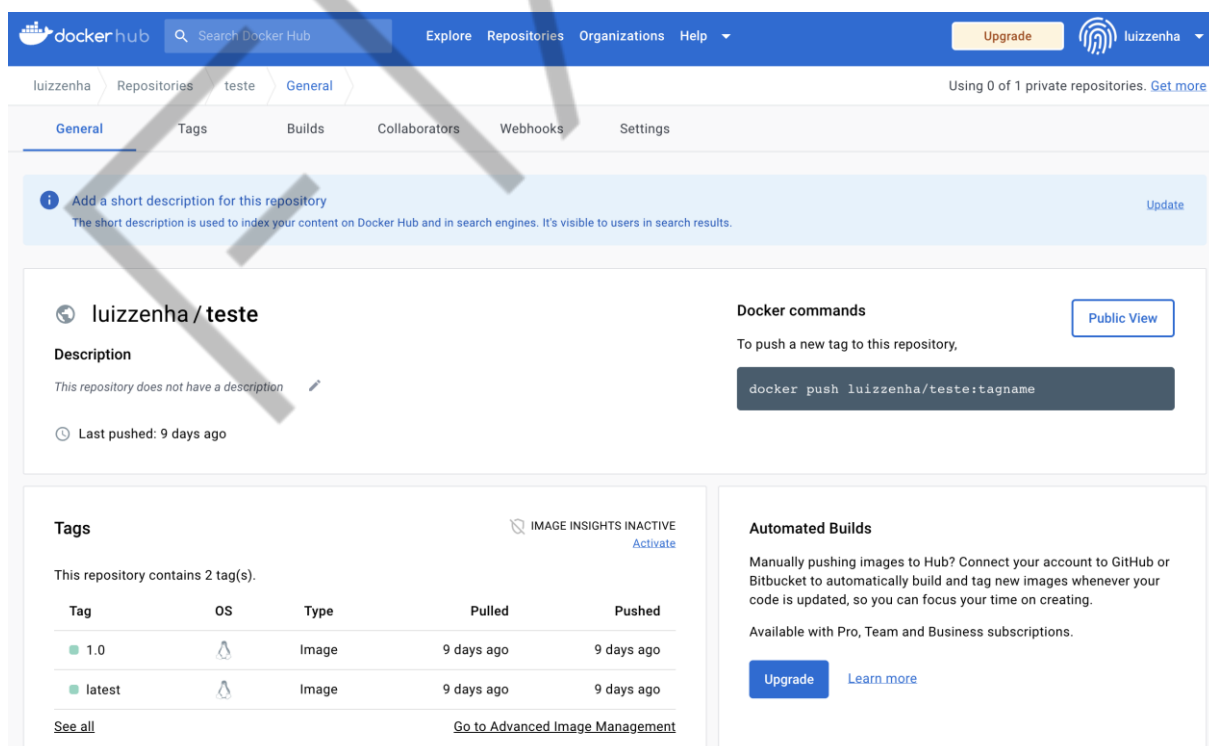


Figura 2 - Página de detalhamento de uma imagem publicada no Docker Hub
 Fonte: <https://hub.docker.com/> (2023)

O versionamento de imagem é uma prática importante no desenvolvimento de aplicativos baseados em containers, pois permite que as versões anteriores das imagens sejam mantidas e atualizadas sem afetar a funcionalidade do aplicativo. Além disso, permite que os desenvolvedores gerenciem e implementem várias versões de um aplicativo em diferentes ambientes (por exemplo, desenvolvimento, teste e produção) sem conflitos ou erros.

O comando utilizado é 'docker tag', no qual passamos as seguintes informações:

```
$ docker tag <ImagemLocal>:<TagLocal>  
<SeuUsuario>/<NomeDesejado>:<TagFinal>
```

Comando de prompt 1 – Comando para versionar imagem
Fonte: Elaborado pelo autor (2023)

É importante termos o cuidado de colocar o mesmo usuário do Docker Hub que está logado no Docker cli ou Docker desktop para que, no momento da publicação, não ocorra problemas.

Após definirmos a tag, o Docker entende que ela está apta para ser publicada por meio do comando 'docker push', que realizará o upload dessa imagem para o Docker Hub.

```
$ docker push <SeuUsuario>/<NomeDesejado>:<Versão>
```

Comando de prompt 2 – Comando para publicar imagem
Fonte: Elaborado pelo autor (2023)

Além de garantir que outros devs e seus sistemas consigam acessar e baixar suas imagens, é muito importante garantir a segurança antes de executar um container.

Podemos verificar as imagens baixadas com um comando disponível juntamente com os demais comandos do Docker, o 'docker scan'.

DOCKER SCAN

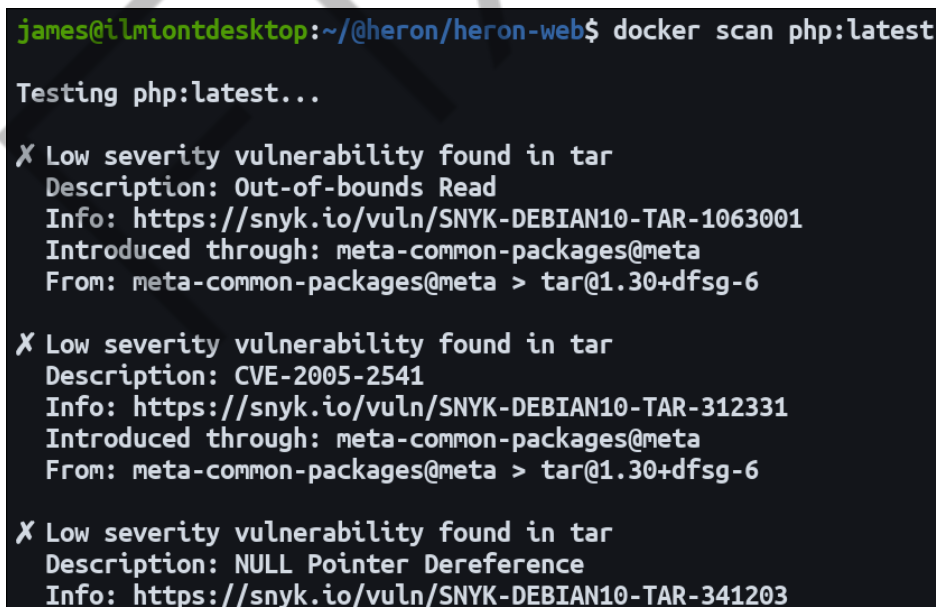
O comando 'docker scan' é uma ferramenta de segurança integrada ao Docker, que permite analisar imagens de container em busca de vulnerabilidades conhecidas e outros problemas de segurança. O "docker scan" é executado no cliente Docker, permitindo que os usuários analisem imagens de contêineres antes de executá-las.

O comando "docker scan" utiliza o serviço de varredura de vulnerabilidades Snky para analisar a imagem do container em busca de vulnerabilidades conhecidas em bibliotecas, sistemas operacionais, arquivos de configuração e outros elementos da imagem. O resultado da análise inclui uma lista de vulnerabilidades encontradas e suas informações detalhadas, como a gravidade da vulnerabilidade e a solução recomendada.

O "docker scan" é fácil de usar. Basta executar o comando "docker scan" seguido do nome da imagem do container que deseja analisar. Por exemplo:

```
$ docker scan image_name
```

Comando de prompt 3 – Comando para publicar imagem
Fonte: Elaborado pelo autor (2023)



```
james@ilmiontdesktop:~/@heron/heron-web$ docker scan php:latest
Testing php:latest...
X Low severity vulnerability found in tar
  Description: Out-of-bounds Read
  Info: https://snky.io/vuln/SNYK-DEBIAN10-TAR-1063001
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > tar@1.30+dfsg-6

X Low severity vulnerability found in tar
  Description: CVE-2005-2541
  Info: https://snky.io/vuln/SNYK-DEBIAN10-TAR-312331
  Introduced through: meta-common-packages@meta
  From: meta-common-packages@meta > tar@1.30+dfsg-6

X Low severity vulnerability found in tar
  Description: NULL Pointer Dereference
  Info: https://snky.io/vuln/SNYK-DEBIAN10-TAR-341203
```

Figura 3 - Retorno de um Docker scan

Fonte: <https://www.howtogeek.com/devops/how-to-use-docker-scan-to-find-vulnerabilities-in-your-images/> (2023)

Ao executar o comando, ele analisará e exibirá um relatório completo de todas as vulnerabilidades existentes, terá inclusive um link explicando como corrigir aquela vulnerabilidade em questão. Lembre-se de que esse processo de scan deve ser algo constante durante a criação e utilização de containers, pois novas vulnerabilidades podem sempre ser encontradas.

EMANIP

O QUE VOCÊ VIU NESTA AULA?

Nesta aula, aprendemos a como utilizar o Docker Hub e como publicar imagens no registry público do Docker. Vimos também que além de baixar, criar e utilizar imagens, precisamos nos preocupar com vulnerabilidades, e descobrimos que o próprio docker já nos fornece ferramentas para validarmos e verificarmos as imagens que temos localmente em nosso computador.

Não se esqueça de que estamos disponíveis na comunidade do Discord para auxiliar você no que for necessário! Explore!

REFERÊNCIAS

VULNERABILITY scanning for Docker local images. **Docker.com**. 2023. Disponível em: <https://docs.docker.com/engine/scan/>. Acesso em: 14 mar. 2023.

WHAT Is Docker Hub? Explained With Examples. **SimpliLearn**. 2023. Disponível em: <https://www.simplilearn.com/tutorials/docker-tutorial/docker-hub>. Acesso em: 14 mar. 2023.

EMEND

PALAVRAS-CHAVE

Docker. Docker Hub. Docker Scan. Vulnerabilidade. Container.

EMSE



POSTECH