iCompute.us Web Portal User Guide

Basil - Containerization and Deployment Tool

Welcome to the iCompute.us web portal, your gateway to utilizing Basil, a powerful tool for semi-automatic containerization, deployment, and execution of applications and workflows on various cloud computing and supercomputing platforms. This user guide will walk you through the process of using Basil via the icompute.us web portal.

Table of Contents

1. Introduction to Basil

Basil is a versatile tool that simplifies the process of creating Docker images without requiring in-depth knowledge of image creation. It accepts recipes in various formats (Makefiles/CMakefiles, scripts, commands, or predefined text templates), generates Dockerfiles, builds images, scans for vulnerabilities, signs images, and allows deployment on diverse hardware platforms. The homepage has information regarding the current functionalities available as well as new updates listed under "Project News" section.

2. Getting Started

- Access iCompute.us: Visit icompute.us and login using your credentials. There are two ways to login, you can either create a new account linked to your email address, or you can choose to login with CILogon, which is available for University or Academic research email accounts.

3. Creating Docker Images

- Once logged in, navigate to either the "Terminal" tab or the "iSpec" tab.

- In the "Terminal" tab, you can interactively provide specifications for building a Docker image by uploading the necessary script and software files, navigating to that directory and then typing *basil -b*. Complete the required input sections with the correct information regarding your software and the Dockerfile will be created and shared with you right in your

project directory, or you can choose to push it to a Docker Hub account. You can also choose to have the link to download sent to you by email. There is a video demonstration of this process available under the "Help" tab listed as "Video Demos".

- In the "iSpec" tab, you can specify the recipe for image creation using scripts, commands, or predefined text templates. Upload any relevant files needed for building the Dockerfile and follow along the process that involves filling out necessary input fields to build your Dockerfile and in the end you can choose to receive your Dockerfile one of two ways:

    a. Push the generated Docker image to a Docker Hub account by entering the username and access token.

    b. Receive an email with the link to download the generated Docker image.

4. Testing and Deploying Images

- Generate Image: After specifying the recipe, click the appropriate button to generate the Dockerfile. In the iSpec tab, you will be shown a completed YAML file based on your input, which is used by Basil to make the Dockerfile and generate the Docker image in the background. The customer will be sent an email regardless of the action that will contain information and steps on what to do next relevant to your choices.

- Build Image: Basil will automatically build the image using the generated file. The image will either be sent to a Docker Hub account or shared with you via a link to download the TAR file that contains the image. For more information on how to run the image using both methods, a video demonstration is available under the "Help" tab listed under "Video Demos".

- Deployment: Images can be deployed on user systems or in the cloud on various hardware platforms based on customer needs.

5. Managing Images

- View Images: You can view your generated images in either the Docker Hub account it was pushed to, or you can check your emails for an email by (icomputeus@gmail.com) that will include some attachments as well as a link to download the TAR file, that is the image.

- Scan for Vulnerabilities: Basil automatically scans generated images for vulnerabilities, you can review the scan results and address any issues. The vulnerability analysis run by SNYK is attached as a text file named 'vulnerability_analysis.log' in the email sent to you after generation.

- Sign and Release: If desired, you can sign images and release them in public registries with appropriate licenses.

6. Running created Docker image in user environment

If you wish to build a Docker image on the system of your choice using the Dockerfile generated by Basil, then please make sure that you have Docker installed on the system. Please review the steps below to install Docker. Docker is required even if you would just like to run the Docker image as well, so it is a good idea to install it beforehand.

Steps for installing Docker on Windows:

1. Go to the Docker website: https://www.docker.com/get-started

2. Download the Docker Desktop Installer.exe file

3. Run the .exe file to run and finish the installation by following the on-screen instructions

Steps for installing Docker on macOS:

1. Open a terminal window

2. Run the following command to download the installer for Docker:

   curl -fsSL https://get.docker.com -o get-docker.sh

3. Now run this command to install Docker:

   sudo sh get-docker.sh

Steps for installing Docker on Linux:

1. Open a terminal window

2. Run the following command:

   sudo apt-add-repository https://download.docker.com/linux/ubuntu/repo/deb

3. Next, run this command:

   sudo apt update

4. Now, run this command:

   sudo apt install docker-ce

5. Finally, run this command to check if Docker has been installed correctly:

   docker version

Ways to run Docker image on user systems:

#1 INSTRUCTIONS FOR RUNNING DOCKER IMAGE ON THE SYSTEM OF YOUR CHOICE

Case 1: The Docker image is available in your or someone else's Docker Hub account

Step 1: Go to the Docker Hub account from which you would like to pull a Docker image, locate the image of your choice, and copy the Docker Pull Command provided on the page. For example: docker pull basilproject/alpine .

Step 2: Paste the command in a terminal session and hit enter, it should pull the image from Docker Hub to your system. Step 3: Run the Docker image using he docker run command as follows: 'docker run replace-this-with-your-image-name' and it should run your image and perform the task that it was programmed to do.

Case 2: Your or someone else's Docker image that may be available in Basil project's Docker Hub account

Step 1: Go to https://hub.docker.com/ and type 'basilproject' in the search bar located in the top left part of the website. You do not need to have a Docker Hub account to perform this action.

Step 2: Locate the image that was pushed here, if you selected to push your image, the image name will have your username in it to make it easier to identify the image.

Step 3: Copy the Docker Pull Command provided on the webpage and paste it into your terminal session and hit enter.

Step 4: Once the pull has completed, run 'docker run [image name' command and it should run your image.

Case 3: Did not push the image to Docker Hub

Step 1: Check your email, you should have an email from 'icomputeus@gmail.com'. It should have a link to download the file as well as instructions on how to run it, they are also provided here.

Step 2: Click the link to download the file, once the file has downloaded, open terminal and navigate to the folder where the .tar.gz file is located.

Step 3: Run this command, docker load < [image name.tar.gz] and wait for the image to be loaded.

Step 4: Confirm the image is available on your system by running 'docker image ls'.

Step 5: Run the command, 'docker run [image name]', and you are done!

#2 INSTRUCTIONS FOR BUILDING THE DOCKER IMAGE FROM THE BASIL GENERATED DOCKERFILE

Note: You will need "sudo"/"root"/"admin" level permissions on the system on which you would need to build a Docker image.

Step 1: If you used the iSpec form om the Basil website to generate the Docker image, then you can download the Dockerfile from the link provided in the email notification you received from the Basil project. If you ran the "Basil" tool on the Basil website through the terminal tab, then would can click the "Refresh List" button on the terminal page, select the Dockerfile that you wish to download from the drop down menu, and click on the "Download" button.

Step 2: To build an image based on the instructions on the Dockerfile, run the following command 'docker build .' from the folder containing the Dockerfile.

Step 3: Once the docker image is built successfully, type 'docker run [image name]' to run the image.

#3 INSTRUCTIONS FOR RUNNING THE IMAGE ON HPC SYSTEMS USING SINGULARITY

Step 1: If you are working on an HPC system that is provisioned with the module system, you can run the "module avail" command and check if Singularity is already installed. If Singularity is

available as a module, you can load it by using the "module load [singularity-module-name]" command.

Step 2: Use the following command to pull an image from Docker Hub and convert it into Singularity Image Format (SIF) file.

'singularity pull [image name].sif docker://[docker image name]:[tag]' *# remove [] in actual use*

Step 3: Run the Singularity image by using the command as follows

'singularity run [image name].sif'

PLEASE NOTE: If you already have the image, you may not need to rebuild it. You can confirm if the image is available on your system by running the 'docker image ls' command in the terminal.

7. Interacting with Basil via Command Line

- Command-Line Mode: You can interact with Basil via the command line using the "Terminal" tab. Execute commands to specify image specifications, generate files, build images, and more. To start this process, simply head to your project directory and type the command 'basil -b'.

8. Troubleshooting

If you encounter issues during any step of the process, refer to the following resources:

- Documentation: Check the documentation section of iCompute.us for detailed instructions.

- Support: Reach out to the iCompute.us support team for assistance.

- Video Demos: Listed under the "Help" tab, this section has a set of video demonstrations that include various use cases and cover the functionalities offered by Basil.

9. FAQs

For frequently asked questions, visit the "FAQ" section on the iCompute.us website.

Congratulations! You've successfully learned the basics of using Basil through the iCompute.us web portal. With Basil's powerful capabilities, you can simplify the containerization, deployment, and execution of your applications and workflows on various computing platforms.