

End Point Software/Application Management Portal

Tushar Dahibhate¹, Krish Gambhir², Shruti Kothari³ &
Sushant Dharmadhikari⁴

Department of Computer Engineering, Sinhgad College of Engineering, Vadgaon(Bk), Pune

Abstract: The installation of any software requires prior knowledge of the software dependencies which only a person with some technical background will have. The installation process could be tedious and time consuming at times. Endpoint Software/Application management portal will work to manage software and applications on any endpoint. It will be used to automate the process of software installation to reduce the manual efforts required during installation of a software such as- extraction, installation, post-configurations, etc. and will also be time saving.

The software will consist of a Web UI that would be used to take the requirements from the user such as-name of the software, version, and end-point operating system. The extraction and installation of the software will be done by the chef tool. Installation of applications would be done by the docker tool. Rest API will act as an interface between the front end and the back end.

Keywords: Chef Tool, Docker Tool.

1. Introduction :

Endpoint Software/Application management portal will work as a platform to manage software and applications on any endpoint. It will be used to automate the process of software installation. It will reduce the manual efforts required during installation of a software such as- extraction, installation, post-configurations, etc. and will also The basic components for the tool are as below.

Components:

- Chef client
- Docker Engine- Docker engine must be present on client machine. It is responsible to create docker containers. Docker containers are light weight containers which wrap up a piece of software in a complete file system along with every file system that a software needs to run.
- Docker image is a software which we load into the container. It is created using a Docker file.

be time saving. It also provides the development environment for deploying various applications inside isolated containers in a platform independent manner.

The extraction and installation of the software will be done by the chef tool. Providing the development environment will be done using Docker tool.

2. Literature survey :

There exists some tools for automation of the installation of software. Ninite is a tool supported only for the Windows operating system which provides various softwares to install and automate the installation process. There is no such tool that supports the linux operating system.

Docker is a tool that can package an application and its dependencies in a virtual container that can run on any Linux server. This helps enable flexibility and portability on where the application can run. Docker uses a client-server architecture.

Chef is a powerful automation platform that transforms complex infrastructure into code. Chef is built around simple concepts: achieving desired state, centralized modeling of IT infrastructure, and resource primitives that serve as building blocks. These concepts enable you to quickly manage any infrastructure with Chef.

3. Block Diagram :

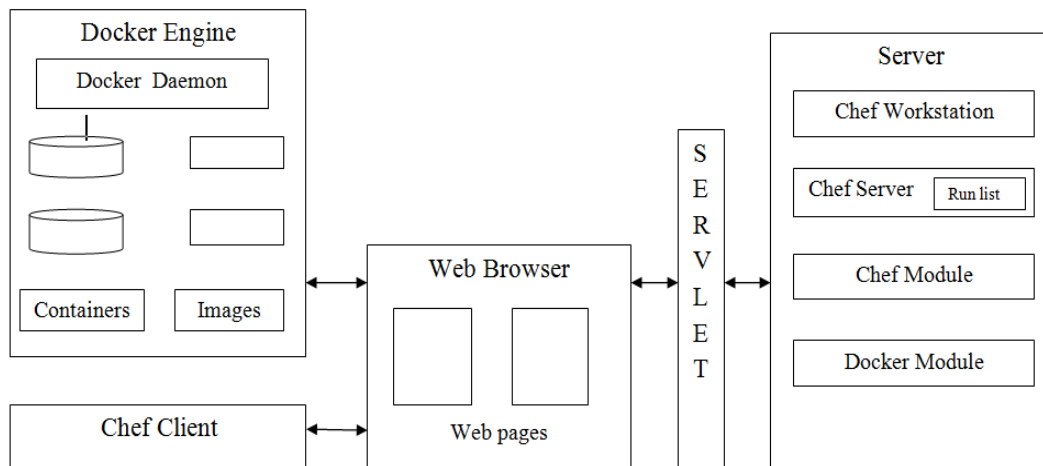
A Docker file is a text file which contains all the commands that a user could call on the command line to assemble an image.

- Docker Daemon – Docker daemon communicates with the Docker engine and manages the Docker containers.
- Web Browser
- Servlet
- Server

The server works as a store to keep the recipes for the software and the client lists and information of the client. The requests are transferred from the client to the server

through the servlet and executed by the chef

client/ docker engine at the client side.



3.1 Architecture

Working of the tool:

- **Software Installation :**

Step 1: Accepting the user input.

- Software Name
- Operating System
- Version
- Credentials
- IP Address

Step 2: Installation of Chef client.

Step 3: Searching if the software is present in the database.

Step 4: Creation of recipe corresponding to the software.

Step 5: Running the chef recipe for installation of software.

- **Application Deployment:**

Step 1: Accepting the user input.

- Application environment
 - Version
 - Path of the file to be executed(file/files must be provided in the form of a tar archive)
 - Credentials
 - IP Address
- c) Direct installer path for the software

Step 2: Searching the Dockerfile corresponding to the development environment.

Step 3: Building the Dockerfile to create a Docker Image.

Step 4: Creating and Starting docker container using the Docker image built before.

Step 5: Copying the file into the container.

Step 6: Providing the details of the container to the user.

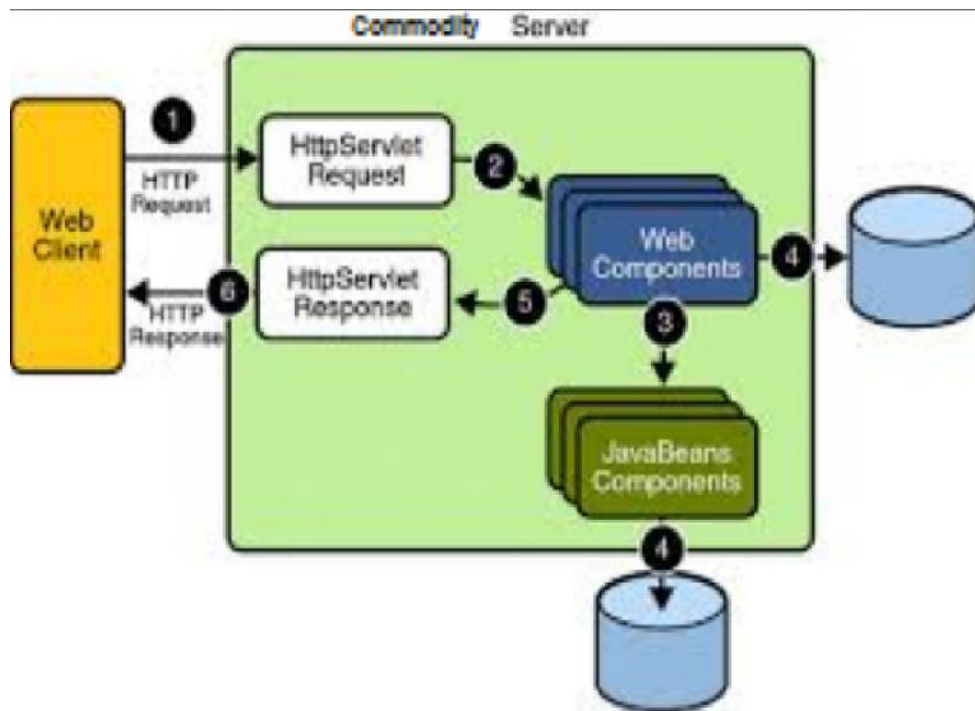
4. System Features :

1. UI features- Interface for providing installation details

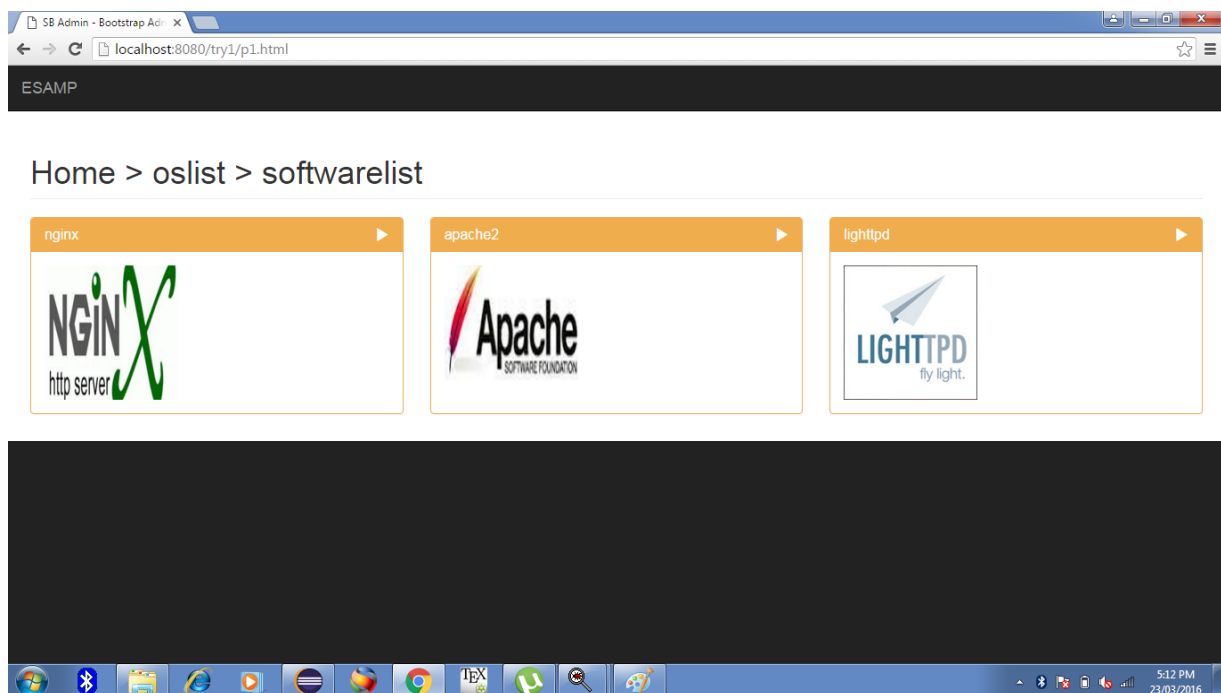
Our system will be based on Client-server architecture. We have employed a powerful commodity server. The server will be responsible for accepting client requests and giving them results accordingly.

There are three ways provided for the user to put his request up for the server:

- a) Name of the software
- b) URL for the software



4.1. A Web Server Workflow



4.1.b UI sample

2. Software Installation

This feature will provide the option for the selection of the OS at the endpoint.

Depending on the OS selected the user then will provide the software name, IP address and other credentials.

2.1 Chef Client

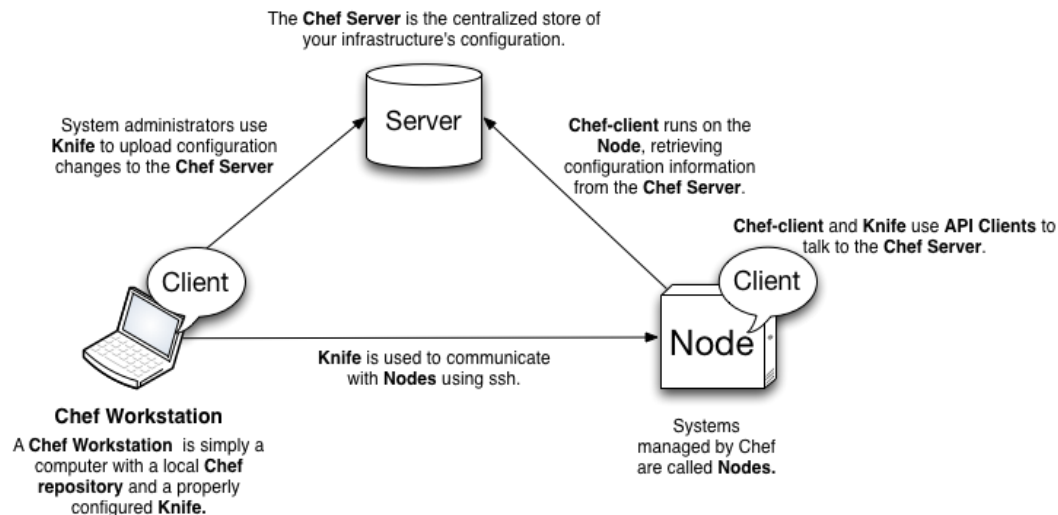
The chef client will take the responsibility to install the software automatically by running the corresponding recipe.

2.2 Chef Workstation

The chef-workstation will be responsible to manage the nodes in the network.

2.3 Chef Server

The chef server stores the recipes for the software and the client lists and information of the client. When a client is bootstrapped or registered to a server, the server installs the chef client during this process on the client side.



4.2.a Chef Workflow

3. Application Environment provision

- User will provide the environment and the version along with the path of the file to be deployed and the user credentials.

3.1 Docker functionality

Users need to have Docker installed on their system with remote API feature enabled and started. As Docker will be installed on the client side itself, it will have all the containers as well as the Docker images. Server will be responsible to fire the API requests to the Docker daemon installed on the client machine, which in turn will be responsible for building the image and spawning the containers. The remote API provided by Docker is RESTful. Once the user provides all the details in the UI, server will be responsible to fire appropriate API requests to the Docker daemon present on the client. Docker daemon will then communicate with Docker engine to build the Docker image and spawn the container using it. The container spawned will have the necessary environment that the user desires.

5. Conclusion:

The End point Software and Application Management Portal has been successfully designed and tested. The user can get the software or application environment installed through the Chef and Docker Tools, thus making the system independent and automated. The required software or application can be easily deployed on the client machine and can be made ready to use without any human intervention.

References:

- [1] www.wikipedia.org
- [2] www.youtube.com
- [3] www.learn.chef.io
- [4] Steven c. Markey REST in the cloud www.ibm.com
- [5] docs.docker.com
- [6] www.digitalocean.com/community/tags/docker?type=tutorials
- [7] blog.flux7.com/blogs/docker/docker-tutorial-series-part-1-an-introduction