## 11.2 Remote access to containers

This section describes environments in which an instructor or red team member is to interact with containers within the lab, e.g., to perform penetration testing. This interaction would occur via computers external to the lab exercise, e.g., networked to a server hosting VMs. The strategy employed to achieve this depends on whether the lab utilizes GNS3, (which manages the virtual networks without relying on Docker networking).

### 11.2.1 Remote access without GNS3

Docker port publishing provides external network access to containers. For example, remote ssh access to a specific container within the lab can be achieved as follows:

- Use the `PUBLISH` directive in the start.config to bind a container port to a host VM port, e.g.,

  ```
  PUBLISH 0.0.0.0:60020:20/tcp
  ```

- Use port forwarding to bind the VM port to a server port. Here, the host port would differ for each VM on a server as a means of naming the VM whose lab is to accessed. For example, on VirtualBox, a port forwarding entry might be:

  | Host IP | Host Port | Guest IP | Guest Port |
  |---|---|---|---|
  | 0.0.0.0 | 61022 | 0.0.0.0 | 60022 |

The above example would then allow an external computer to ssh into the selected container using port 60122, assuming the container has SSH enabled (see the telnet-lab server container for an example). Authentication to control who can SSH into a given container could be provided through use of SSH keys. This remotely accessed container can be hidden from the student, and provide the instructor or red-team participant with a means to probe and attempt to compromise the other computers within the Labtainers exercise network.

### 11.2.2 Remote access with GNS3

For labs that run in the GNS3 environment, remote network access is provided through use of the GNS3 *cloud* endpoint device, which interacts with an Ethernet network interface. In this example, access is provided from external to the VM – with no network access to the container from within the VM.

The following assumes your VM has a virtual Ethernet interface named `enp0s3`, with IP an address on the `10.0.2.0/24` subnet. On your VM, find the Ethernet interface that has an assigned IP address. Alternately you could define the VM to share a physical host network, but that is outside the scope of this example.

Define a component within your Labtainers lab that is be remotely accessed, e.g., a workstation or router, and assign it an IP address on the `enp0s3` interface subnet, e.g., `10.0.2.100`. Within the start.config file, provide the container with the `KICK_ME <LAN>` attribute, where LAN is the name of the network intended to be connected to the cloud component. Then, when defining the GNS3 network topology, i.e., creating and connecting links:

- Select a `Cloud` component from the `Browse End Devices` menu, and drag it to the desktop. (computer terminal icon).