

# Console access

This page documents the design and requirements for providing console access to running stack services.

## "Exec" and terminal emulators

Docker and Kubernetes both support the "exec" command, which supports running arbitrary commands in a running container. The "exec" command includes options for creating an interactive session and allocating a pseudo-terminal (pty) to enable interactive access with system commands, such as shells.

In most cases, users are running the "exec" command from the command line and therefore from a terminal emulator (i.e., xterm, terminal, iTerm, etc). The terminal emulators support specific features of older [hardware-based terminals](#). A variety of [terminal emulators](#) are available, including [Javascript](#).

## Requirements:

Epic: User can access console of running service via GUI and CLI

Story: User can access console of running service via GUI

- Preconditions:
  - User is logged in
  - Stack exists and is running
- Primary flow:
  - GUI displays a button/link for each running stack service
  - User selects button/link
  - Console is displayed in new window or tab
  - User can interact with container as expected (navigate, edit files, etc)
- Alternative flows
  - User can access console of multiple containers at the same time
  - User can access the console of the same container in multiple sessions
  - If stack is stopped during active session, terminal session is terminated
  - Interaction with terminal continues current user session (i.e., token refresh)

Story: User can access console of running service via CLI

- Preconditions:
  - User is logged in via CLI
  - Stack exists and is running
- Primary flow:
  - `ndslabsctl exec stack-service-id`
  - User can interact with container as expected (navigate, edit files, etc)
- Alternative flows
  - User can access console of multiple containers at the same time
  - User can access the console of the same container in multiple sessions
  - If stack is stopped during active session, terminal session is terminated
  - Interaction with terminal continues current user session (i.e., token refresh)

## Prototype:

A prototype has been provided using [term.js](#) with a simple go server:

```
cd $GOPATH/src/github.com
```

```
git clone https://github.com/craig-willis/terminal-test
```

```
cd terminal-test/server
```

```
go build
```

```
./server
```

Open index.html in browser:

#### Terminal div

```
Welcome to NDS Labs  
clowder@spvmd7-imagepreview-okoggt:/$ pwd  
/  
clowder@spvmd7-imagepreview-okoggt:/$ ls  
bin  dev          etc  lib  media  opt  root  sbin  sys  usr  
boot  entrypoint.sh  home  lib64  mnt    proc  run   srv   tmp  var  
clowder@spvmd7-imagepreview-okoggt:/$
```

## References

- <http://www.linusakesson.net/programming/tty/>