

ICS-OS Lab 01: Building ICS-OS

Objectives

At the end of this activity, you should be able to:

1. build the ICS-OS kernel binary image;
2. build the ICS-OS distribution floppy disk image; and
3. boot ICS-OS and run two commands.

1 Introduction

ICS-OS is an instructional operating system that can be used for understanding different operating system concepts. An operating system is no different from other software in that it is written in a programming language, such as C.

The build process creates the compressed kernel binary image (`vmdex`) and the floppy disk image (`ics-os-floppy.img`). Since ICS-OS has a relatively large number of source files, the `Make` utility is used for the build. You can examine the contents of `Makefile` to learn more of the details how this is done.

To start ICS-OS, the floppy disk image is set as the boot device in Qemu, which is an emulator for various microarchitectures. the floppy disk image contains the GRUB bootloader which transfers control to the ICS-OS kernel (in `vmdex`). After the boot process, a prompt is provided to users to enter commands.

2 Prerequisites

The recommended development environment is Ubuntu 16.04. Tasks described here may or may not work on other Linux distributions. Familiarity with the command line is also needed.

`docker`¹ and `docker-compose`² should also be installed to build in the latest version of Ubuntu. Check `ics-os/Dockerfile` for the dependencies needed to build ICS-OS.

3 Deliverables and Credit

Perform the tasks below and capture screenshots while you do them. Submit a PDF file containing the screenshots with captions. Do not forget to put your name and laboratory section. Credit is five (5) points.

4 Tasks

Task 1: Clone the repository and explore the source tree

ICS-OS³ is open source and is hosted on Github. Run the following command to checkout the source code and explore the source tree.

```
$git clone https://github.com/srg-ics-uplb/ics-os.git
$cd ics-os/ics-os
```

¹<https://docs.docker.com/engine/install/ubuntu/>

²<https://docs.docker.com/compose/install/>

³<https://github.com/srg-ics-uplb/ics-os/>

Task 3: Build

Run the commands below to build. The commands should be repeated whenever changes in the kernel source code are made. You will modify the kernel source code in future homeworks.

```
$make clean  
$make vmdex  
$make floppy
```

Task 4: Boot

Start Qemu with the floppy image as boot device using the command below.

```
$make boot-floppy
```

Task 5: Run ICS-OS commands

Once the ICS-OS command prompt (%) appears, type **help**. Examine the list of commands and run two of these commands. Do not forget to capture screen shots of the outputs.