

# TNE30019/TNE80014 – Unix for Telecommunications

## System Bootup

Dr. Jason But

Swinburne University

Dr. Jason But

TNE30019/TNE80014 – System Bootup

## System Startup – UEFI/BIOS

- Initial OS to manage hardware (UEFI/BIOS)
- Firmware on motherboard
- Loaded after turning on computer
- Has basic functionality to access hardware
- Knows bootable hardware devices
  - User can define device order in setup
- Knows how to hand over to hardware for booting
  - Can start boot loader from hard disk
  - Can start PXE stack on network card

Dr. Jason But

TNE30019/TNE80014 – System Bootup

## Outline

- How Unix System Starts Up
- What Happens after System Boots
- Startup Scripts
- The login Process
- The shell

Dr. Jason But

TNE30019/TNE80014 – System Bootup

## System Startup – Boot Loader

- How does an OS load?
- **Boot Loader** is small program that loads and runs OS kernel
- Needs to know where kernel is stored on disk
- Often provides some means to
  - Select which kernel to boot
  - Set parameters for kernel to boot

### Boot Loader Functions

- 1 Loads kernel executable
- 2 Starts kernel

Dr. Jason But

TNE30019/TNE80014 – System Bootup

## System Startup – Kernel

- Is the actual OS

### Kernel Startup Functions

- 1 Run system checks, e.g memory test
- 2 Setup memory management, CPU, interrupt handling
- 3 Detect hardware devices
- 4 Load initial RAM disk (Linux)
- 5 Load device drivers
  - Drivers create devices in /dev
  - Configures hardware ready for use
- 6 Start idle process, scheduler and other key kernel processes (**PID=0**)
- 7 Mount root file system
- 8 Launch `init` – process initialisation daemon (**PID=1**)

## System Startup – Run Levels

- Unix has concept of booting into different run-levels
- Different start-up scripts/services are run depending on chosen run-level
- Eight runlevels, three of which are “standard”

### Standard and common run-levels

- 0 Halt (standard)
- 1 Single-user mode (standard)
- 2 Local multi-user console (common)
- 3 Multi-user console (common)
- 5 Multi-user graphical desktop (common)
- 6 Reboot (standard)

## System Startup – `init`

- `init` is first process and manages startup of system processes
- `/etc/inittab` is the configuration file for `init`
  - Specifies default run-level
  - If no default run-level `init` will prompt user
- `init` launches `rc` with run-level as parameter
- `init` starts virtual terminals
- On Linux, big push to replace `init` with `systemd`

### rc script

Shell script to start basic services, daemons and applications

Linux `/etc/init.d/rc`

FreeBSD `/etc/rc`

## System Startup – `rc`

- This is where you can customise system
- Script decides what processes to launch

### FreeBSD

- Runs scripts in `/etc/rc.d` with the `start` flag
  - Example: `/etc/rc.d/ntpd start` – Launches `ntpd` daemon
- Scripts written so processes are launched in correct order
- Variables configured in `/etc/rc.conf` tell scripts in `/etc/rc.d` whether to actually launch that process and command line parameters to use

### Linux

- Directories maintained for each run level – location platform specific
- Each directory contains sym links to scripts to execute

## System Shutdown

- Signal is sent to `init` to terminate
- On FreeBSD `init` runs `/etc/rc.shutdown`
  - Script tries to stop all processes launched by `rc`
- On Linux `init` executes scripts in `rc0.d` or `rc6.d`

### `init` – Final Shutdown

After all services stopped

- Terminate any remaining processes
- Flush disks
- Terminate and shuts down system

## The login Process

- Virtual terminals started by `init` will launch `login` application
- This is runnable process (`/bin/login`)

### `login` process

- Displays the “login:” prompt
- Prompts for username and password
- Checks username and password against database
- If user is properly authenticated, grants access

### `login` is **NOT** user interface

- Sets environment variables
- Launches yet another process – the **shell**
- Shell provides user interface

## Your system is booted ... and then

### User Interaction

- Console: keyboard + mouse + display
- On servers user interaction is typically done via network connections (remote console)
- On clients need console for user interaction
- Console access on servers is also useful in emergencies

### Unix systems output to **console**

- Traditionally terminal connected to serial port
- Nowadays console is tied to virtual terminal (`/dev/ttyv0`)
- Graphics card driver ties this terminal to display

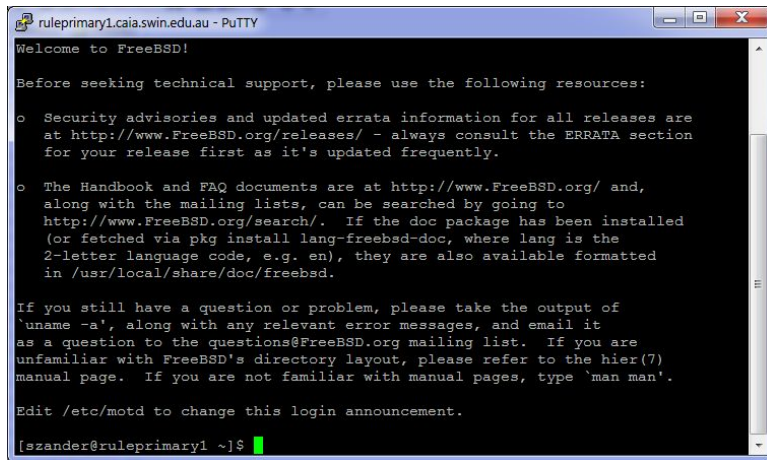
## The Shell

- Shell is basically command interpreter
  - Prompt user for input
  - Execute user's commands
  - Display output
- Which shell `login` starts defined in user database

### Many different shells with different functionality

`sh` Bourne shell  
`bash` Bourne-again shell  
`csh` C shell  
`tcsh` TENEX C shell  
`ksh` Korne shell  
`zsh` Z shell

# We Can Start Working Now



A screenshot of a PuTTY terminal window titled 'ruleprimary1.caia.swin.edu.au - PuTTY'. The terminal displays the FreeBSD login sequence. It starts with 'Welcome to FreeBSD!', followed by a message about technical support resources. Then, it lists two resources: security advisories and the Handbook/FAQ. After that, it provides instructions on how to ask for help via the mailing list. Finally, it prompts the user to edit /etc/motd. The prompt '[szander@ruleprimary1 ~]\$' is shown at the bottom with a green cursor.

```
ruleprimary1.caia.swin.edu.au - PuTTY
Welcome to FreeBSD!

Before seeking technical support, please use the following resources:

o Security advisories and updated errata information for all releases are
  at http://www.FreeBSD.org/releases/ - always consult the ERRATA section
  for your release first as it's updated frequently.

o The Handbook and FAQ documents are at http://www.FreeBSD.org/ and,
  along with the mailing lists, can be searched by going to
  http://www.FreeBSD.org/search/. If the doc package has been installed
  (or fetched via pkg install lang-freebsd-doc, where lang is the
  2-letter language code, e.g. en), they are also available formatted
  in /usr/local/share/doc/freebsd.

If you still have a question or problem, please take the output of
`uname -a`, along with any relevant error messages, and email it
as a question to the questions@FreeBSD.org mailing list. If you are
unfamiliar with FreeBSD's directory layout, please refer to the hier(7)
manual page. If you are not familiar with manual pages, type `man man`.

Edit /etc/motd to change this login announcement.

[szander@ruleprimary1 ~]$
```