

## Εργαστηριακές Ασκήσεις 6η

### Εργαστηριακή άσκηση 1η

Σε αυτή την άσκηση χρησιμοποιείται η γλώσσα σεναρίου Perl για την αντικατάσταση της εντολής `logger(1)`.

1. Προσπαθήστε να εκτελέσετε την εντολή `logger` εισάγοντας στο τερματικό σας “`logger hello`” και έπειτα χρησιμοποιήστε την εντολή `tail` στο αρχείο `/var/log/messages` για να δείτε τις εγγραφές καταγραφής. Τι παρατηρείτε;
2. Δίνεται το παρακάτω σενάριο σε Perl που εμφανίζει την φράση “`hello world`” στο `syslog` μαζί με το χαρακτηριστικό PID της διεργασίας `logger.pl`.

```
#!/usr/bin/perl
```

```
use strict;                               # compile-time checks
use warnings;                             # enable run-time warnings

use Sys::Syslog qw(:DEFAULT setlogsock); # library for syslog functions

setlogsock('unix');                       # use a unix domain socket
openlog("logger.pl", 'pid', 'user');      # register ourselves
syslog('info', 'hello world');           # the syslog call
closelog();                               # close the socket
```

Εκτελέστε το σενάριο και επιβεβαιώστε ότι λειτουργεί σωστά ελέγχοντας το αρχείο `/var/log/messages`.

3. Αφού κατανοήσετε την λειτουργία του σεναρίου, εξηγήστε την λειτουργία των εντολών `setlogsock`, `openlog`, `syslog` και `closelog`.

### Εργαστηριακή άσκηση 2η

Σε αυτή την άσκηση ασχολείται με την διαχείριση του πυρήνα με την μέθοδο της δυναμικής φόρτωσης προγραμμάτων οδήγησης συσκευών.

1. Οι φορτώσιμες μονάδες του πυρήνα αποθηκεύονται στην θέση `/lib/modules/version`, όπου `version` είναι η τρέχουσα έκδοση του πυρήνα. Μπείτε στο παραπάνω κατάλογο και κάντε πλοήγηση στους διάφορους υποκαταλόγους για να δείτε τα αρχεία ή μονάδες του πυρήνα (αρχεία `.ko`).
2. Βρείτε που βρίσκεται η μονάδα `vfat` η οποία είναι οδηγός για το σύστημα αρχείων FAT των Windows με υποστήριξη μεγάλων ονομάτων αρχείων. Υπόδειξη: τα αρχεία των μονάδων τελειώνουν σε `.ko`
3. Η εντολή `/sbin/lsmmod` χρησιμοποιείται για την εμφάνιση των τρέχουσων φορτώσιμων μονάδων. Εκτελέστε την εντολή για να δείτε ποιες μονάδες είναι φορτωμένες στο σύστημα σας.
4. Η εντολή `/sbin/insmod` χρησιμοποιείται για την φόρτωση μιας μονάδας του πυρήνα. Προσπαθήστε να φορτώσετε μια συσκευή `dummy`. Επιβεβαιώστε ότι φορτώθηκε η μονάδα και πείτε αν η μονάδα αυτή χρησιμοποιείται.
5. Η εντολή `/sbin/rmmod` χρησιμοποιείται για την ξεφόρτωση μιας μονάδας του πυρήνα. Ξεφορτώστε την μονάδα `dummy`. Ελέγξτε αν ξεφορτώθηκε η μονάδα αυτή.

6. Δοκιμάστε να φορτώσετε την μονάδα `hisax` η οποία είναι ένα πρόγραμμα οδήγησης για κάποιες κάρτες `isdn`. Σε αυτή την περίπτωση θα εμφανιστούν σφάλματα διότι λείπουν σύμβολα (`unresolved symbols`). Αυτό συμβαίνει γιατί η μονάδα `hisax` έχει εξαρτήσεις σε άλλες μονάδες και έτσι δεν μπορεί να φορτωθεί μέχρι να ικανοποιηθούν όλες οι εξαρτήσεις. Προσδιορίστε ποια είναι αυτά τα σύμβολα, εκτελώντας την εντολή `dpmsg`. Επομένως, για να φορτωθεί η μονάδα `hisax` πρέπει να φορτωθούν ξεχωριστά οι μονάδες που εξαρτώνται πράγμα που είναι μια ασύμφορη λύση στην περίπτωση που υπάρξουν πολλές εξαρτώμενες μονάδες. Για το λόγο αυτό υπάρχει μια καλύτερη λύση που θα δούμε αμέσως παρακάτω.
7. Πολλές μονάδες φορτώνονται αυτόματα όταν είναι απαραίτητοι παρά την χειροκίνητη φόρτωση. Έτσι, για να φορτωθούν συγκεκριμένες μονάδες είναι σημαντικό να γνωρίζουμε ποιες μονάδες εξαρτώνται. Η πληροφορία αυτή αποθηκεύεται σε ένα αρχείο εξάρτησης μονάδων που λέγεται `modules.dep`. Το αρχείο αυτό βρίσκεται κάτω από το κατάλογο `modules`. Η κάθε γραμμή του αρχείου `modules.dep` έχει την μορφή  
`module : dependee-modules-list`  
όπου `dependee-modules-list` είναι μια λίστα των μονάδων που εξαρτάται η μονάδα `module`. Προσδιορίστε ποιες μονάδες εξαρτάται η μονάδα `hisax`;
8. Υπάρχει ένα άλλο χρήσιμο εργαλείο που επιλύει αυτόματα τις εξαρτήσεις των μονάδων και λέγεται `modprobe`. Προσπαθήστε να φορτώσετε την μονάδα `hisax` με το εργαλείο `modprobe`.
9. Επιβεβαιώστε αν φορτώθηκε η μονάδα `hisax` και οι υπόλοιπες μονάδες.
10. Τέλος, ξεφορτώστε την μονάδα `hisax` και τις υπόλοιπες μονάδες.