



ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ

Τμήμα Πληροφορικής

ΕΠΛ 371 - Προγραμματισμός Συστημάτων

ΑΣΚΗΣΗ 2 - Αποστολή/Ανάκτηση/Ανάλυση Αρχείων Κειμένου χρησιμοποιώντας το πρωτόκολλο FTP και το Κέλυφος Bash

Διδάσκων: Δημήτρης Ζεϊναλιπούρ

Υπεύθυνος Εργαστηρίου: Παύλος Αντωνίου

Ημερομηνία Ανάθεσης: Πέμπτη 19/02/15

Ημερομηνία Παράδοσης: Πέμπτη 5/03/15 και ώρα 12:00 (14 μέρες)

(η λύση να υποβληθεί σε zip μέσω του Moodle και ο κώδικας να
να παραδοθεί εκτυπωμένος στο εργαστήριο)

<http://www.cs.ucy.ac.cy/courses/EPL371>

I. Στόχος Άσκησης

Στόχος αυτής της άσκησης είναι η εξοικείωση με προχωρημένες τεχνικές προγραμματισμού στο κέλυφος Bash, και η εκτίμηση της ευκολίας με την οποία μπορεί κανείς να δημιουργήσει ένα σύνθετο σύστημα μέσω προγραμμάτων ωφελιμότητας (system utilities). Συγκεκριμένα, σε αυτή την άσκηση θα έχετε την ευκαιρία να κάνετε χρήση των εξής εννοιών που έχετε διδαχθεί στα πλαίσια του μαθήματος: εντολή *echo*, πίνακες, συνθήκες ελέγχου, δομές επανάληψης, κανονικές εκφράσεις, επεξεργαστές ροών (*sed*, *awk*) και χρήση συναρτήσεων με τα προαναφερθέντα. Το θέμα της άσκησης είναι η υλοποίηση ενός προγράμματος αποστολής, ανάκτησης και ανάλυσης αρχείων κειμένου (text files) πάνω από το πρωτόκολλο FTP (File Transfer Protocol) χωρίς την χρήση κάποιου έτοιμου εργαλείου (π.χ., WinSCP, FileZilla, κτλ). Οι λειτουργίες του προγράμματος σας και το αναμενόμενο αποτέλεσμα περιγράφονται αναλυτικότερα στην συνέχεια.

II. Προαπαιτήσεις

Το πρωτόκολλο FTP λειτουργεί με το μοντέλο πελάτη εξυπηρετητή (client-server). Περισσότερες λεπτομέρειες θα δώσουμε πιο κάτω. Το στοιχείο του εξυπηρετητή ονομάζεται FTP daemon. Για το σκοπό της άσκησης θα πρέπει να εγκαταστήσετε ένα FTP daemon πάνω στη δική σας εικονική μηχανή (VPS).

Οι πιο κάτω εντολές παρουσιάζουν πως μπορείτε να εγκαταστήσετε και να διαχειριστείτε τον FTP daemon:

- `export http_proxy='http://proxy.cs.ucy.ac.cy:8008'`
Όπως ξέρετε από την πρώτη άσκηση, η εντολή αυτή ορίζει τον proxy server για να έχουμε πρόσβαση στο διαδίκτυο για κατέβασμα πακέτων.
- `yum install vsftpd`
Εντολή για εγκατάσταση του FTP daemon vsftpd
- `service vsftpd status`
Εντολή που δείχνει την κατάσταση του daemon. Προς το παρόν, λόγω του ότι είναι ανενεργός θα εκτυπωθεί το μήνυμα: “vsftpd is stopped”

<http://www.cs.ucy.ac.cy/courses/EPL371>

- `service vsftpd start`
Εντολή που ενεργοποιεί τον FTP daemon και εκτυπώνεται το μήνυμα
“Starting vsftpd for vsftpd: [OK]”
Οπότε αν δώσουμε την εντολή: “`service vsftpd status`” θα λάβουμε την
απάντηση “vsftpd (pid 13323) is running...”
- `vi /etc/vsftpd/vsftpd.conf`
Η εντολή αυτή ανοίγει το configuration file του FTP daemon. Αν κάνουμε
οποιαδήποτε αλλαγή στο αρχείο αυτό πρέπει να επανεκκινήσουμε το daemon με
την εντολή “`service vsftpd restart`” και θα λάβουμε την απάντηση:
“Shutting down vsftpd: [OK]”
“Starting vsftpd for vsftpd: [OK]”
- `service vsftpd stop`
Η εντολή αυτή απενεργοποιεί τον FTP daemon εκτυπώνοντας το πιο μήνυμα:
“Shutting down vsftpd: [OK]”

Η πρόσβαση σε έναν εξυπηρετητή FTP μπορεί να επιτευχθεί με δύο τρόπους:

- Ανώνυμη
- Πιστοποιημένη

Στη λειτουργία ανώνυμης πρόσβασης, οι απομακρυσμένοι υπολογιστές-πελάτες μπορούν να έχουν πρόσβαση στο εξυπηρετητή FTP χρησιμοποιώντας τον προεπιλεγμένο λογαριασμό χρήστη (default username) που ονομάζεται “anonymous” ή “ftp” και στέλνοντας οτιδήποτε (ακόμα και κενό) κωδικό πρόσβασης (password). Η λειτουργία ανώνυμης πρόσβασης είναι ενεργοποιημένη όταν στο αρχείο `/etc/vsftpd/vsftpd.conf` υπάρχει έξω από τα σχόλια (#) η εντολή `anonymous_enable=YES`. Από προεπιλογή, στους ανώνυμους χρήστες δεν επιτρέπεται να ανεβάζουν (upload) αρχεία στον εξυπηρετητή FTP. Για να αλλάξετε αυτή τη ρύθμιση πρέπει να ανοίξετε το αρχείο `/etc/vsftpd/vsftpd.conf`, να βγάλετε από τα σχόλια την εντολή `anon_upload_enable=YES` και να επανεκκινήσετε τον FTP daemon. Μέσω της ανώνυμης λειτουργίας ο απομακρυσμένος πελάτης θα έχει πρόσβαση (για αποστολή και λήψη δεδομένων) στον κατάλογο `/var/ftp`. Στον κατάλογο αυτό πρέπει να έχει δικαιώματα εγγραφής (write) ο χρήστης ftp. Αν θέλετε να αλλάξετε τον ιδιοκτήτη του καταλόγου `/var/ftp` από root σε ftp τότε δώστε την εντολή “`chown -R ftp /var/ftp`”. Αξίζει να λεχθεί ότι η ενεργοποίηση της ανώνυμης αποστολής δεδομένων στον εξυπηρετητή FTP αποτελεί σοβαρό κίνδυνο για την ασφάλεια του εξυπηρετητή, ωστόσο το VPS είναι προσβάσιμο μόνο από το τοπικό δίκτυο του τμήματος, επομένως αυτός ο κίνδυνος είναι μικρός!

Στη λειτουργία με πιστοποίηση ταυτότητας ένας χρήστης πρέπει να έχει ένα λογαριασμό και έναν κωδικό πρόσβασης.

Στην άσκηση αυτή, θα χρησιμοποιήσουμε την λειτουργία ανώνυμης πρόσβασης για να αποφύγουμε την διαδικασία περαιτέρω ρυθμίσεων στον εξυπηρετητή FTP. Αξίζει να λεχθεί όμως ότι υπάρχει δυνατότητα όποιος επιθυμεί να χρησιμοποιήσει την πιστοποιημένη λειτουργία.

III. Αποστολή / Ανάκτηση αρχείων μέσω πρωτοκόλλου FTP

Σε αυτή την ενότητα της εκφώνησης θα δούμε πως μπορεί κανείς να αποστείλει και να ανακτήσει αρχεία μέσω του κελύφους Bash. Αρχικά θα δείξουμε πως επιτυγχάνεται σύνδεση μεταξύ πελάτη και εξυπηρετητή σε υψηλό επίπεδο, στη συνέχεια θα παρουσιάσουμε τις

σημαντικότερες εντολές του πρωτοκόλλου FTP και τέλος θα δείξουμε πώς να εγκαθιδρύσετε ένα FTP κανάλι επικοινωνίας μέσω εντολών του κελύφους Bash. Στόχος μας δεν είναι να επεξηγήσουμε σε βάθος το πρωτόκολλο μεταφοράς αρχείων FTP (Request For Comments 959: <http://tools.ietf.org/html/rfc959>), διότι αυτό είναι το αντικείμενο μελέτης άλλων μαθημάτων.

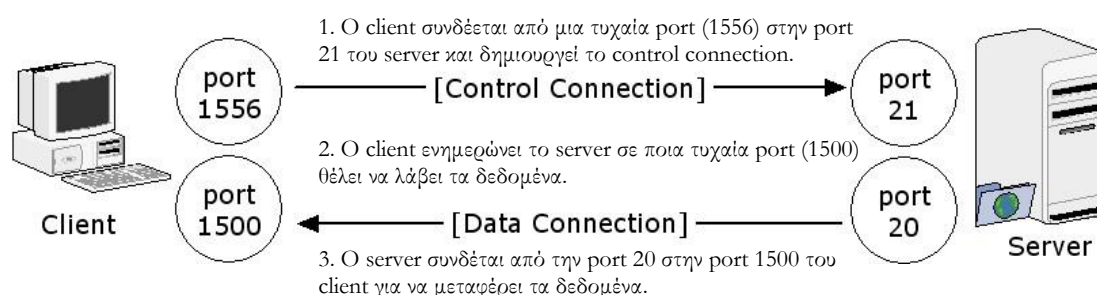
1. Τρόπος Λειτουργίας Πρωτοκόλλου

Το πρωτόκολλο FTP λειτουργεί με βάση το μοντέλο πελάτη-εξυπηρετητή (client-server) επιτρέποντας την αποστολή/ανάκτηση αρχείων μεταξύ των 2 αυτών οντοτήτων.

Αρχικά ο FTP server ανοίγει την θύρα (port) 21 περιμένοντας έναν FTP client να συνδεθεί. Στη συνέχεια ο client ξεκινά μια νέα σύνδεση από μια τυχαία θύρα προς την θύρα 21 του server. Μόλις γίνει η σύνδεση παραμένει ανοιχτή για όλη τη διάρκεια της συνόδου FTP. Η συγκεκριμένη σύνδεση ονομάζεται σύνδεση ελέγχου (**control connection**) και μέσα από το κανάλι αυτό στέλνονται όλες οι εντολές από τον client και οι απαντήσεις από το server, αλλά όχι δεδομένα (αρχεία). Έπεται η δημιουργία της σύνδεσης δεδομένων (**data connection**), της σύνδεσης με την οποία μεταφέρονται τα δεδομένα. Υπάρχουν δύο τρόποι για να δημιουργηθεί η σύνδεση δεδομένων μέσω της σύνδεσης ελέγχου, (α) με χρήση της ενεργητικής λειτουργίας (active mode) στην οποία τα δεδομένα σπρώχνονται (push) από τον server στον client ή (β) με χρήση της παθητικής λειτουργίας (passive mode) στην οποία τα δεδομένα αντλούνται (pull) από τον client.

Ενεργητική λειτουργία

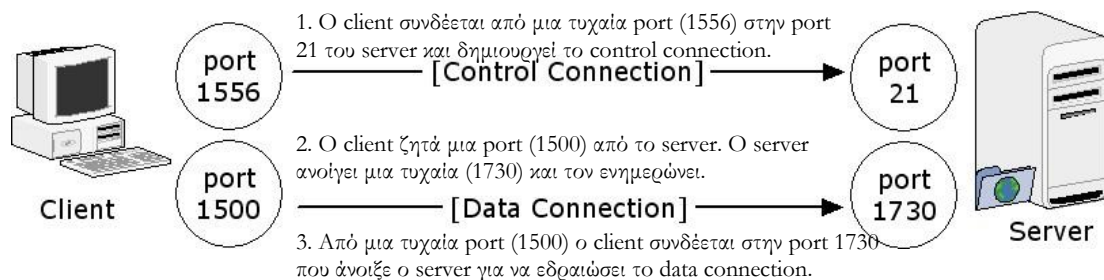
Στην ενεργητική λειτουργία (active mode) ο FTP client διαλέγει μια τυχαία θύρα στην οποία δέχεται τα δεδομένα της σύνδεσης. Ο client στέλνει τον αριθμό της θύρας, στην οποία επιθυμεί να "ακούει" (listen) για εισερχόμενες συνδέσεις. Ο FTP server δημιουργεί μια σύνδεση από την θύρα 20 στην ανοιχτή θύρα του client για τη μεταφορά των δεδομένων. Οποιαδήποτε πληροφορία ζητήσει ο client, ανταλλάσσεται με βάση αυτή τη σύνδεση, που βασίζεται στο TCP. Όταν η μεταφορά ολοκληρωθεί ο server κλείνει τη σύνδεση αποστέλλοντας ένα πακέτο FIN, όπως σε κάθε σύνδεση βασισμένη στο TCP. Κάθε φορά που ο client ζητάει δεδομένα, δημιουργείται κατά παρόμοιο τρόπο μια σύνδεση δεδομένων και η διαδικασία επαναλαμβάνεται.



Εικόνα 1: Διαγραμματική και διαλογική αναπαράσταση της ενεργητικής λειτουργίας FTP.

Παθητική λειτουργία

Στην παθητική λειτουργία (passive mode) ο client ζητά από τον server να διαλέξει μια τυχαία θύρα, στην οποία θα "ακούει" (listen) για την σύνδεση δεδομένων (data connection). Ο server ενημερώνει τον client για την θύρα την οποία έχει διαλέξει και ο client συνδέεται σε αυτή για τη μεταφορά των δεδομένων. Η μεταφορά ολοκληρώνεται όπως και στην ενεργητική λειτουργία (active mode), αφού η σύνδεση δεδομένων βασίζεται στο TCP.



Εικόνα 2: Διαγραμματική και διαλογική αναπαράσταση της παθητικής λειτουργίας FTP.

2. Εντολές FTP

Οι εντολές που περιγράφονται πιο κάτω είναι ένα υποσύνολο των εντολών που υποστηρίζει το πρωτόκολλο FTP, με τις οποίες θα ασχοληθούμε στην παρούσα άσκηση. Όπως έχει λεχθεί και προηγουμένως, οι εντολές αυτές θα στέλνονται από το client στο server μέσω της σύνδεσης ελέγχου.

Εντολές ελέγχου πρόσβασης

USER <username>	καθορισμός κωδικού που είναι καταχωρημένος στο server	
PASS <password>	καθορισμός συνθηματικού που είναι καταχωρημένος στο server	
CWD <pathname>	αλλαγή καταλόγου	(όπως cd)
CDUP	μετάβαση στο «γονικό» κατάλογο	(όπως cd ..)
QUIT	έξοδος	

Εντολές προσδιορισμού παραμέτρων

PORT h1,h2,h3,h4,p1,p2	Δήλωση ενεργητικής λειτουργίας: κοινοποίηση τοπικής θύρας δεδομένων. Επεξήγηση των παραμέτρων της εντολής γίνεται στο παράδειγμα που ακολουθεί.
PASV	Δήλωση παθητικής λειτουργίας: αίτηση για αποστολή θύρας αφούγκρασης (listening) του server για παραλαβή δεδομένων
TYPE	καθορισμός τύπου δεδομένων: ascii (για κείμενο), image (για εικόνες και binary αρχεία), default τιμή: ascii

Εντολές υπηρεσίας

RETR <remote_pathname>	ανάκτηση (retrieve) αρχείου. Η εντολή αυτή μπορεί να εκτελεστεί αν υπάρχει ανοικτή σύνδεση δεδομένων (δηλαδή μετά από την εντολή PORT ή PASV).	(όπως cp remote_file local_file)
STOR <remote_pathname>	αποστολή (store) αρχείου. Η εντολή αυτή μπορεί να εκτελεστεί αν υπάρχει ανοικτή σύνδεση δεδομένων (δηλαδή μετά από την εντολή PORT ή PASV).	(όπως cp local_file remote_file)
APPE <pathname>	αποστολή αρχείου και προσάρτηση σε υφιστάμενο αρχείο κειμένου που βρίσκεται στο server. Η εντολή αυτή μπορεί να εκτελεστεί αν υπάρχει ανοικτή σύνδεση δεδομένων (δηλαδή μετά από την εντολή PORT ή PASV)	(όπως cat local_file >> remote_file)
ABOR	διακοπή (abort) προηγούμενης εντολής υπηρεσίας	
PWD	εμφάνιση τρέχοντος καταλόγου	(pwd)
LIST	μεταφορά της λίστας των αρχείων	(ls -la) Η εντολή αυτή μπορεί να εκτελεστεί αν υπάρχει ανοικτή σύνδεση δεδομένων (δηλαδή μετά από την εντολή PORT ή PASV). Η εντολή αυτή επιστρέφει παρόμοια αποτελέσματα με την εντολή ls -la. Οι κατάλογοι επισημαίνονται με "d".

DELE <pathname>	διαγραφή αρχείου	(rm)
MKD <pathname>	δημιουργία καταλόγου	(mkdir)
RMD <pathname>	διαγραφή καταλόγου	(rmdir)
SIZE <pathname>	μέγεθος αρχείου (οκτάδες, 8-bit byte) σα δεκαδικός αριθμός	
STAT <pathname>	εάν δοθεί όρισμα ένα directory τότε είναι το ίδιο με την εντολή list με τη διαφορά ότι η απάντηση από το server έρχεται μέσα από τη σύνδεση ελέγχου (χωρίς να πρέπει να ανοίξει σύνδεση δεδομένων)	

Περισσότερες πληροφορίες μπορείτε να βρείτε στο RFC959 (<http://www.ietf.org/rfc/rfc0959.txt>) ή στον πιο κάτω σύνδεσμο: <http://www.nsftools.com/tips/RawFTP.htm>

3. Παράδειγμα σύνδεσης FTP με το κέλυφος Bash

Ας δούμε τι γίνεται όταν κάποιος θελήσει να συνδεθεί με τον προσωπικό του λογαριασμό μεταφοράς αρχείων και να αποστείλει ένα αρχείο, χωρίς την χρήση εξειδικευμένων εργαλείων γραφικού περιβάλλοντος π.χ., WinSCP, FileZilla, κτλ.

```
# Θα πρέπει να ανοίξουμε 2 sockets: 1 για τη σύνδεση ελέγχου και 1 για τη σύνδεση
δεδομένων
# Άνοιξε ένα τερματικό και εκτέλεσε την ακόλουθη εντολή, η οποία ανοίγει ένα tcp socket
(ένα κανάλι επικοινωνίας) με τον εξυπηρετητή μεταφοράς αρχείων που είναι εγκατεστημένος
στο VPS σας, στην θύρα 21 (σύνδεση ελέγχου), για ανάγνωση/γραφή. Με την εντολή αυτή
δημιουργείται ο χειριστής του socket #3 (socket descriptor) μέσω του οποίου θα γίνεται η
αποστολή και λήψη δεδομένων προς και από το socket.
exec 3<>/dev/tcp/<myVPS>.in.cs.ucy.ac.cy/21

# Αποστείλε το username και password του λογαριασμού για να αποκτήσεις πρόσβαση
πάνω στα μηνύματα ηλεκτρονικού ταχυδρομείου. Αν χρησιμοποιήσουμε ανώνυμη
πρόσβαση τότε το username θα είναι anonymous ή ftp και το password μπορεί να'ναι
κενό.
echo USER anonymous >&3
echo PASS >&3

# Μετά από κάθε εντολή μπορούμε να εκτυπώνουμε την απάντηση του server στο
STDOUT.
read line <&3
echo $line

# Για να ανεβάσουμε αρχεία στο server πρέπει να ανοίξουμε και ένα άλλο socket. Πιο κάτω
υλοποιείται η παθητική λειτουργία (passive mode) για το άνοιγμα της σύνδεσης δεδομένων.
Προτού ανοίξει το νέο socket, ο client στέλνοντας την εντολή PASV απαιτεί από το server
να του στείλει το IP και το port τα οποίο θα χρησιμοποιήσει ο client για να αρχικοποιήσει
τη σύνδεση TCP.
echo PASV > &3
# Η απάντηση του server έχει την πιο κάτω μορφή:
(h1,h2,h3,h4,p1,p2)
# Τα h1-h4 είναι τα δεκαδικά ψηφία της διεύθυνσης IP του FTP server και το p1 και p2
προσδιορίζουν τον αριθμό του port με 2 δεκαδικά ψηφία. Τα 2 δεκαδικά αυτά ψηφία θα
πρέπει να μετατραπούν σε 2 8-ψήφιους δυαδικούς αριθμούς. Ο 16-ψήφιος δυαδικός αριθμός
```

θα πρέπει να μετατραπεί σε δεκαδικό για να βρούμε τον αριθμό port. Για παράδειγμα: (10,16,15,109,215,199)

Εύρεση IP address = 10.16.15.109

Εύρεση port number: 215₁₀ = 11010111₂, 199₁₀ = 11000111₂

Ενώνω τους 2 δυαδικούς αριθμούς: 1101011111000111₂ = 55239₁₀

```
exec 5<>/dev/tcp/<myVPS>.in.cs.ucy.ac.cy/55239
```

Μέσω του socket #5 αποστέλλεται το αρχείο και από το socket #3 λαμβάνονται στατιστικά στοιχεία της επικοινωνίας. Όταν αποσταλεί ολόκληρο το αρχείο, ο client κλείνει το socket #5.

Κλείσε το output redirection για το socket

```
exec 5>&-
```

Κλείσε το input redirection για το socket

```
exec 5<&-
```

Στη συνέχεια μπορούμε να ανεβάζουμε τα αρχεία ένα-ένα στο server. Κάθε φορά που αποστέλλεται ένα αρχείο θα πρέπει να κλείνει η σύνδεση από το client.

Στο τέλος του προγράμματος θα πρέπει να κλείσετε το input/output redirection του socket, για να απελευθερώσετε τον File Handler #3.

```
# Κλείσε το output redirection για το socket
```

```
exec 3>&-
```

```
# Κλείσε το input redirection για το socket
```

```
exec 3<&-
```

4. Ζητούμενα Άσκησης

Στην άσκηση αυτή καλείστε να υλοποιήσετε δύο εντολές (commands). Η πρώτη εντολή, όταν εκτελείται θα «ανεβάζει» στον anonymous λογαριασμό του VPS σας ένα αριθμό αρχείων (χρησιμοποιήστε το επισυναπτόμενο ZIP αρχείο) που θα είναι αποθηκευμένα τοπικά σε ένα κατάλογο (directory). Με τη δεύτερη εντολή κατεβάζετε μέσω του προγράμματος σας τα δεδομένα και τα αναλύετε όπως περιγράφεται πιο κάτω.

A) Υποβολή αρχείων

Πρότυπο εντολής: **./ftppupload filedir ftpserver username**

όπου filedir είναι ο κατάλογος που θα βρίσκονται αποθηκευμένα τα μηνύματα, ftpserver είναι ο εξυπηρετητής μεταφοράς αρχείων και username το όνομα χρήστη για το ftp account. Η εντολή θα πρέπει να απαιτεί από τον χρήστη τον κωδικό του (password).

Στο απομακρυσμένο σύστημα αρχείων του ftp server θα πρέπει να δημιουργηθεί η ίδια ιεραρχία που υπάρχει στον κατάλογο filedir.

B) Ανάκτηση-Ανάλυση Αρχείων

Η νέα εντολή που θα δημιουργήσετε θα εκτελεί κάποιες λειτουργίες πάνω στα αρχεία που βρίσκονται στον FTP server. Κάποιες από τις λειτουργίες αυτές προϋποθέτουν την ανάκτηση κάποιων δεδομένων από το server και την αποθήκευσή τους τοπικά, κάποιες άλλες θα γίνονται πάνω στα απομακρυσμένα αρχεία. Οι διαφορετικές λειτουργίες της εντολής θα ορίζονται από τις διαφορετικές παραμέτρους όπως περιγράφεται πιο κάτω.

Η νέα εντολή θα εκτελείται όπως φαίνεται παρακάτω:

Πρότυπο εντολής: **./ftpanalyze [options] ftpserver username**

Η εντολή αυτή θα ενώνεται με τον ftpserver και θα αποκτά πρόσβαση στο λογαριασμό του χρήστη με το δοθέν username. Η εντολή θα πρέπει να απαιτεί από τον χρήστη να δώσει το

password του. Στην συνέχεια, ανάλογα με τα options θα γίνεται ανάκτηση-ανάλυση των αρχείων είτε τοπικά είτε απομακρυσμένα.

Options:

(α) show-dir

Η επιλογή αυτή θα παρουσιάζει στην οθόνη τα περιεχόμενα (αρχεία και καταλόγους) του home directory του user (/) στον απομακρυσμένο FTP server. Κανένα αρχείο δε θα κατεβαίνει τοπικά.

(β) show-file <path/filename>

Η επιλογή αυτή θα παρουσιάζει στην οθόνη το περιεχόμενο του αρχείου που δίδεται από το δεύτερο όρισμα (σχετική διαδρομή αρχείου στον απομακρυσμένο FTP server). Το εν λόγω αρχείο μπορεί να κατεβαίνει τοπικά και να αποθηκεύεται προσωρινά (μέχρι το πέρας της εντολής και μετά να διαγράφεται) στον κατάλογο /tmp/\$USER.

(γ) find-string <path/> <string>

Η επιλογή αυτή θα κοιτάζει μέσα σε όλα τα αρχεία που βρίσκονται στον κατάλογο που ορίζεται από το path (σχετική διαδρομή καταλόγου στον απομακρυσμένο FTP server) και θα παρουσιάζει στην οθόνη τις γραμμές των αρχείων που περιέχουν το string (δεύτερο όρισμα). Τα αρχεία μπορούν να κατεβαίνουν τοπικά και να αποθηκεύονται προσωρινά στον κατάλογο /tmp/\$USER.

(δ) show-dir-R <path/>

Η επιλογή αυτή θα δουλεύει όπως η εντολή "ls -R <path/>", η οποία παρουσιάζει αναδρομικά όλα τα αρχεία που βρίσκονται μέσα (σε οποιοδήποτε βάθος) στον κατάλογο που ορίζεται από το path. Κανένα αρχείο δε χρειάζεται να κατεβαίνει τοπικά.

(ε) show-urls

Η επιλογή αυτή θα κάνει εξαγωγή όλων των συνδέσμων (links) από όλα τα αρχεία που βρίσκονται μέσα στο λογαριασμό του χρήστη και θα τα παρουσιάζει στην οθόνη. Όλα τα αρχεία μπορούν να κατεβούν τοπικά και να αποθηκευτούν προσωρινά στον κατάλογο /tmp/\$USER.

(ζ) analyze-html

Η επιλογή αυτή δημιουργεί ένα φίλτρο το οποίο επεξεργάζεται κάθε ανακτημένη ιστοσελίδα και εξάγει όλες τις λέξεις. Σημειώστε ότι στις λέξεις αυτές, δεν περιλαμβάνονται τα HTML tags, και οι ειδικοί χαρακτήρες HTML, όπως αυτά περιγράφονται πιο κάτω:

• HTML TAG

Οτιδήποτε περιλαμβάνεται μεταξύ των συμβολών < >.

π.χ., <html> , <td bgcolor="red" width="100%">

Σε αυτή την κατηγορία περιλαμβάνονται και τα HTML σχόλια. Ένα σχόλιο ξεκινά με <!-- και τερματίζει με -->

<!-- This is a

Multiline comment

Var[0];

This should be ignored by your lexicon analyzer

-->

• HTML Ειδικοί Χαρακτήρες

Οτιδήποτε περιλαμβάνεται μεταξύ & και ;

π.χ., **&**, ** **, **Á**; Οι ειδικοί χαρακτήρες ΔΕΝ πρέπει να περιλαμβάνονται στο λεξικό σας.

Το σύστημα πρέπει να δημιουργεί ένα λεξικό στο οποίο να εμφανίζει η συχνότητα εμφάνισης της κάθε λέξης που υπάρχει στο ανακτημένο σύνολο σελίδων:

```
$head -6 /tmp/dzeina/lexicon.txt
1942 the
1717 of
1543 and
1047 to
762 a
732 for
```

- Το σύστημα επαναλαμβάνει την δημιουργία του λεξικού 2 φορές την μέρα και αποθηκεύει το λεξικό με κατάληξη την ημερομηνία και ώρα ανάκτησης.
- Οι λέξεις μέσα στο λεξικό δεν κάνουν διάκριση μεταξύ πεζών και κεφαλαίων γραμμάτων.

Γενικοί Περιορισμοί για την εντολή ftpanalyze:

- Το σύστημα ανακτά αρχεία που περιέχουν κείμενο (με επέκταση .txt, .html), και εικόνες (.jpeg, .jpg, .bmp, .gif). Οι εικόνες θεωρούνται δυαδικά αρχεία (binary files). Τα υπόλοιπα αρχεία αγνοούνται.
- Όλες οι εντολές πρέπει να εκτελούνται απομακρυσμένα (μέσω FTP εντολών). Για εντολές που προϋποθέτουν την ανάκτηση αρχείων (π.χ., β, γ και ε) δύναται να χρησιμοποιηθεί ως χώρος προσωρινής αποθήκευσης το tmp/\$USER.

IV. Γενικοί Κανόνες

1. Το σύστημα δεν αφήνει ποτέ άχρηστα και μεταβατικά αρχεία στον δίσκο, ούτε ανοικτά socket descriptors, ανεξάρτητα εάν διακοπεί η λειτουργία του προγράμματος από το κλείσιμο του κελύφους.
2. Το σύστημα πρέπει να χρησιμοποιεί τεχνικές δομημένου προγραμματισμού με την χρήση συναρτήσεων.
3. Το σύστημα πρέπει να ελαχιστοποιεί την χρήση πόρων του συστήματος (αρχεία, μνήμης, κτλ).
4. Το σύστημα πρέπει να μειώνει όσο το δυνατό περισσότερο τον χρόνο διεκπεραίωσης της ανάκτησης και επεξεργασίας των δεδομένων.

Καλή Επιτυχία !