



Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών: Υπηρεσία WWW / Πρωτόκολλο HTTP / Ρύθμιση Apache Web Server

Δρ. Απόστολος Γιάμας

Διδάσκων (407/80)

gkamas@uop.gr

Διαφάνεια 1

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Εισαγωγή στην ορολογία της υπηρεσίας WWW



- Μια ιστοσελίδα (web page) ή έγγραφο (document) αποτελείται από αντικείμενα (objects)
- Είδη αντικειμένων: αρχεία HTML, εικόνες JPEG, αρχεία audio, μικροεφαρμογές (applets) java ...
- Κάθε αντικείμενο διευθυνσιοδοτείται με ένα URL (Uniform Resource Locator)
- Παράδειγμα URL: <http://www.uop.gr/departments/index.html>
- Μία ιστοσελίδα αποτελείται από ένα αρχείο HTML βάσης το οποίο περιλαμβάνει αρκετά αντικείμενα που αναφέρονται με το URL τους.

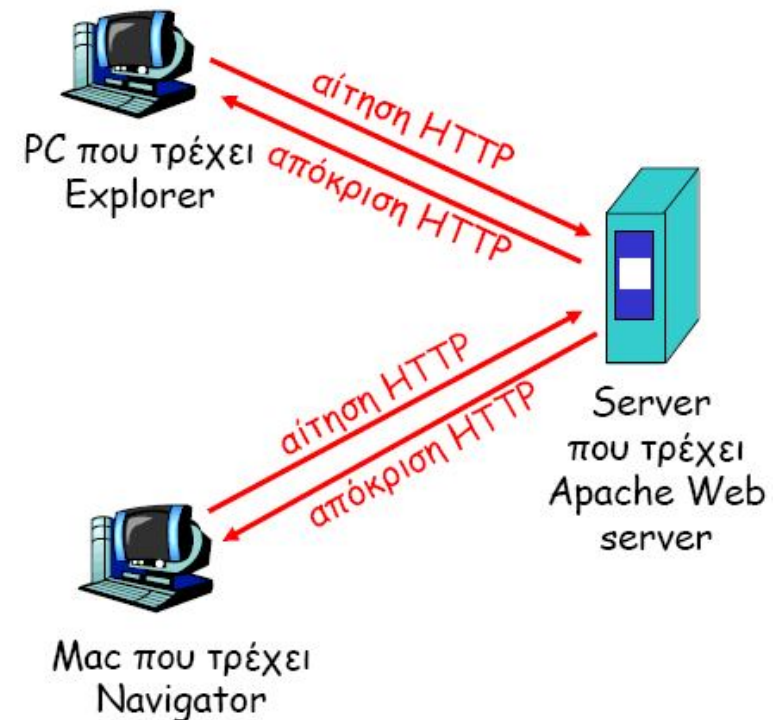
Διαφάνεια 2

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



Γενική επισκόπηση του HTTP

- HTTP: HyperText Transfer Protocol (Πρωτόκολλο Μεταφοράς Υπερκειμένου)
- Πρωτόκολλο επιπέδου εφαρμογής του Web
- Μοντέλο client/server
- Client: Web browser ζητεί, λαμβάνει, απεικονίζει αντικείμενα
- Server: Web server "στεγάζει" αντικείμενα, στέλνει αντικείμενα αποκρινόμενος σε αιτήσεις
- HTTP 1.0: RFC 1945
- HTTP 1.1: RFC 2616



Διαφάνεια 3

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Γενική επισκόπηση του HTTP



- Χρησιμοποιεί TCP:
 - Ο client ξεκινά μία σύνδεση TCP με τον server, θύρα 80
 - Ο server αποδέχεται τη σύνδεση TCP από τον client
 - Ανταλλάσσονται μηνύματα HTTP μεταξύ HTTP client (browser) και HTTP server (Web server)
 - Τερματίζει η σύνδεση TCP
- Το HTTP είναι "stateless": Ο server δε διατηρεί καμία πληροφορία σχετικά με προηγούμενες αιτήσεις
- Η επιλογή "stateless" πρωτοκόλλου στο HTTP έγινε για λόγους μειωμένης πολυπλοκότητας και αυξημένης ταχύτητας. Τα πρωτόκολλα τα οποία διατηρούν καταστάσεις είναι πολύπλοκα γιατί πρέπει να διατηρούν «ιστορικό» για κάθε σύνδεση και σε περίπτωση κατάρρευσής ενός client ή ενός server απαιτείται συγχρονισμός των καταστάσεων.

Συνδέσεις HTTP



- HTTP με μη παραμένουσες συνδέσεις (non-persistent HTTP)
 - Ένα μόνο αντικείμενο μπορεί να σταλεί μέσω μίας σύνδεσης TCP
 - Η έκδοση HTTP/1.0 χρησιμοποιεί μη παραμένουσες συνδέσεις HTTP
- HTTP με παραμένουσες συνδέσεις (persistent HTTP)
 - Πολλαπλά αντικείμενα μπορούν να σταλούν μέσω της ίδιας σύνδεσης TCP
 - Η έκδοση HTTP/1.1 χρησιμοποιεί εκ προεπιλογής (default) παραμένουσες συνδέσεις HTTP

Διαφάνεια 5

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Non-persistent HTTP



- Έστω ότι ένας χρήστης ζητά την ιστοσελίδα με URL www.uop.gr/departments/index.html το οποίο περιέχει κείμενο και αναφορές σε 10 εικόνες jpeg
 1. Ο HTTP client εκκινεί μια σύνδεση HTTP προς τον HTTP server (διεργασία) που τρέχει στον host www.uop.gr στην θύρα 80
 2. Ο HTTP server στον host www.uop.gr περιμένει συνδέσεις TCP στη θύρα 80, αποδέχεται την σύνδεση και ενημερώνει τον client
 3. Ο HTTP client στέλνει μήνυμα αίτησης (που περιέχει το URL) στο socket της σύνδεσης TCP. Το μήνυμα υποδεικνύει ότι ο client θέλει το αντικείμενο `departments/index.html`
 4. Ο HTTP server λαμβάνει το μήνυμα αίτησης, δημιουργεί ένα μήνυμα απόκρισης που περιέχει το αιτούμενο αντικείμενο και στέλνει το μήνυμα στο socket του

Διαφάνεια 6

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Non-persistent HTTP



5. Ο HTTP server κλείνει τη σύνδεση TCP
6. Ο HTTP client λαμβάνει το μήνυμα απόκρισης που περιέχει το αρχείο html. Εξετάζοντας το αρχείο HTML, βρίσκει αναφορές σε 10 αντικείμενα jpeg
7. Τα βήματα 1-4 επαναλαμβάνονται για καθένα από τα 10 αντικείμενα jpeg

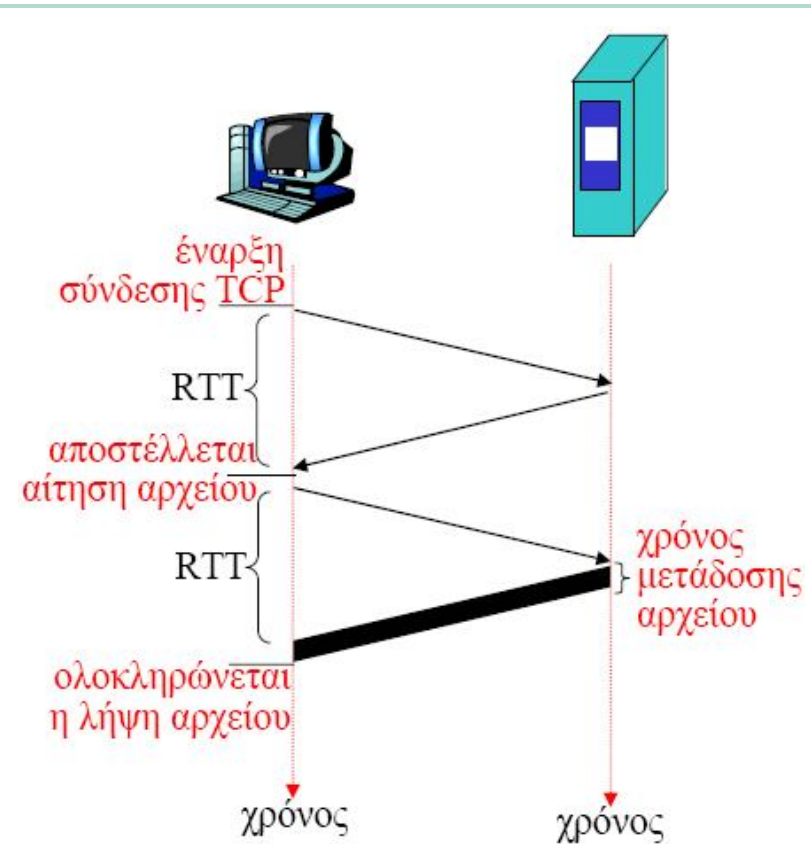
Διαφάνεια 7

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Μοντελοποίηση του χρόνου απόκρισης



- Χρόνος διαδρομής μετά επιστροφής
– Round Trip Time (RTT): Ο χρόνος που χρειάζεται ένα μικρό πακέτο για να ταξιδέψει από τον client στον server
- Χρόνος απόκρισης:
 - Ένα RTT για την έναρξη της σύνδεσης TCP
 - Ένα RTT για την αίτηση HTTP και επιστροφή πρώτων bytes της απόκρισης HTTP
 - Χρόνος μετάδοσης αρχείου
σύνολο = $2RTT + \text{χρόνος μετάδοσης}$



Διαφάνεια 8

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Persistent HTTP



- Μειονεκτήματα non-persistent HTTP:
 - Απαιτεί 2 RTT ανά αντικείμενο
 - Το OS πρέπει να απομείνει σε κάθε σύνδεση TCP πόρους του host (buffers)
 - Οι browsers χρησιμοποιούν συχνά παράλληλες συνδέσεις TCP για τη μεταφορά αντικειμένων
- Persistent HTTP
 - Ο server δεν κλείνει τη σύνδεση αφού στείλει απόκριση
 - Επόμενα μηνύματα HTTP μεταξύ του ίδιου ζεύγους client/server στέλλονται μέσω της ίδιας σύνδεσης TCP

Persistent HTTP



- Persistent χωρίς pipelining
 - Ο client στέλνει ένα νέο μήνυμα αίτησης αφού ληφθεί το προηγούμενο μήνυμα απόκρισης
 - Ένα RTT ανά αντικείμενο
- Persistent με pipelining
 - Προεπιλεγμένο στο HTTP/1.1
 - Ο client στέλνει μήνυμα αίτησης αμέσως μόλις αντιμετωπίσει ένα αναφερόμενο αντικείμενο
 - Ελάχιστη καθυστέρηση: ένα RTT για όλα τα αντικείμενα

Διαφάνεια 10

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



Μήνυμα αίτησης HTTP

- Δύο είδη μηνυμάτων HTTP: αίτηση (request) και απόκριση (response)
- Μήνυμα αίτησης HTTP: ASCII (αναγνώσιμη μορφή)

γραμμή αίτησης (request line)
(εντολές GET, POST, HEAD)

```
GET /somedir/page.html HTTP/1.1
Host: www.someschool.edu
User-agent: Mozilla/4.0
Connection: close
Accept-language: fr

```

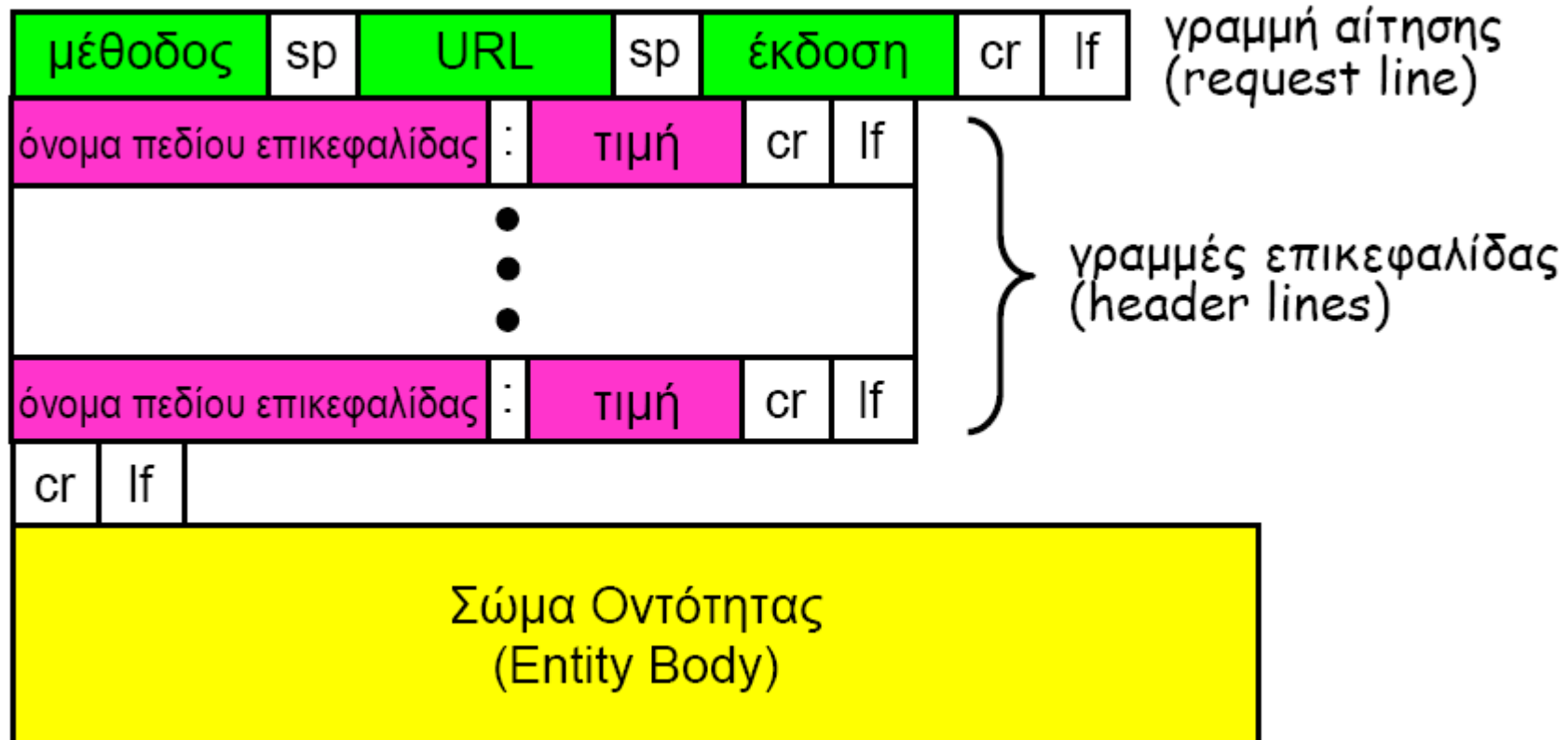
γραμμές επικεφαλίδας (header lines)

κενή γραμμή

Χαρακτήρες επαναφοράς (carriage return),
τροφοδότησης γραμμής (line feed)
υποδεικνύουν το τέλος του μηνύματος



Μήνυμα αίτησης HTTP: Γενική μορφή



Διαφάνεια 12

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Συμπλήρωση φόρμας και μεταφορά περιεχομένου στον server (uploading form input)



— Μέθοδος POST:

- Η ιστοσελίδα συχνά περιλαμβάνει φόρμα προς συμπλήρωση
- Το περιεχόμενο των πεδίων της φόρμας «φορτώνεται» στον server με το entity body

— Μέθοδος URL:

- Χρησιμοποιεί μέθοδο GET
- Το περιεχόμενο των πεδίων της φόρμας προσαρτάται στο πεδίο URL της request line
- Παράδειγμα: www.uop.gr/peoplesearch?person=gkamas

Είδη μεθόδων



- HTTP/1.0
 - GET
 - POST
 - HEAD: ζήτα από τον server να μην περιλάβει το ζητούμενο αντικείμενο στην απόκριση
- HTTP/1.1
 - GET, POST, HEAD
 - PUT: φορτώνει αρχείο που περιέχεται στο entity body στο path που καθορίζεται στο πεδίο URL
 - DELETE: διαγράφει το αρχείο που καθορίζεται στο πεδίο URL



Μήνυμα απόκρισης HTTP

γραμμή κατάστασης (status line)
(πρωτόκολλο, κώδικας κατάστασης, φράση κατάστασης)

HTTP/1.1 200 OK

γραμμές επικεφαλίδας
(header lines)

Connection: close

Date: Thu, 06 Aug 1998 12:00:15 GMT

Server: Apache/1.3.0 (Unix)

Last-Modified: Mon, 22 Jun 1998 ...

Content-Length: 6821

Content-Type: text/html

δεδομένα, π.χ.

data data data data data ...

Κώδικες κατάστασης απόκρισης HTTP



- Στην πρώτη γραμμή (status line) του μηνύματος απόκρισης από τον server στον client
- Μερικά παραδείγματα κωδικών:
 - 200 OK: επιτυχής αίτηση, το αντικείμενο που έχει ζητηθεί έπεται στο ίδιο μήνυμα
 - 301 Moved Permanently: το αντικείμενο που έχει ζητηθεί μεταφέρθηκε, νέο URL καθορίζεται παρακάτω στο ίδιο μήνυμα (επικεφαλίδα Location:)
 - 400 Bad Request: το μήνυμα αίτησης δεν ήταν κατανοητό από server
 - 404 Not Found: Το ζητούμενο έγγραφο δε βρέθηκε σε αυτό τον server
 - 505 HTTP Version Not Supported

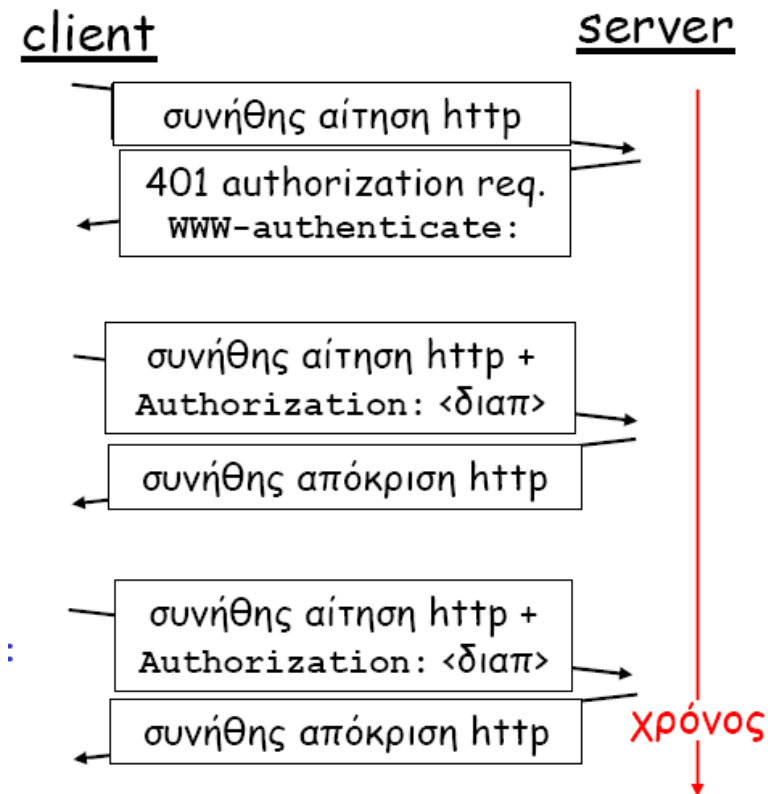
Διαφάνεια 16

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



Εξουσιοδότηση (authorization)

- Εξουσιοδότηση: ελέγχει πρόσβαση στο περιεχόμενο του server
- Χρησιμοποιεί γραμμές επικεφαλίδας & κώδικες κατάστασης
- Διαπιστευτήρια εξουσιοδότησης: συνήθως όνομα, password
- **Stateless:** ο client πρέπει να παρουσιάζει εξουσιοδότηση σε κάθε μήνυμα αίτησης
 - **Authorization:** γραμμή επικεφαλίδας σε κάθε μήνυμα αίτησης
 - Εάν λείπει η γραμμή επικεφαλίδας : **Authorization:** Ο server αρνείται πρόσβαση, στέλνει γραμμή επικεφαλίδας WWW-authenticate: στο μήνυμα response



Διαφάνεια 17

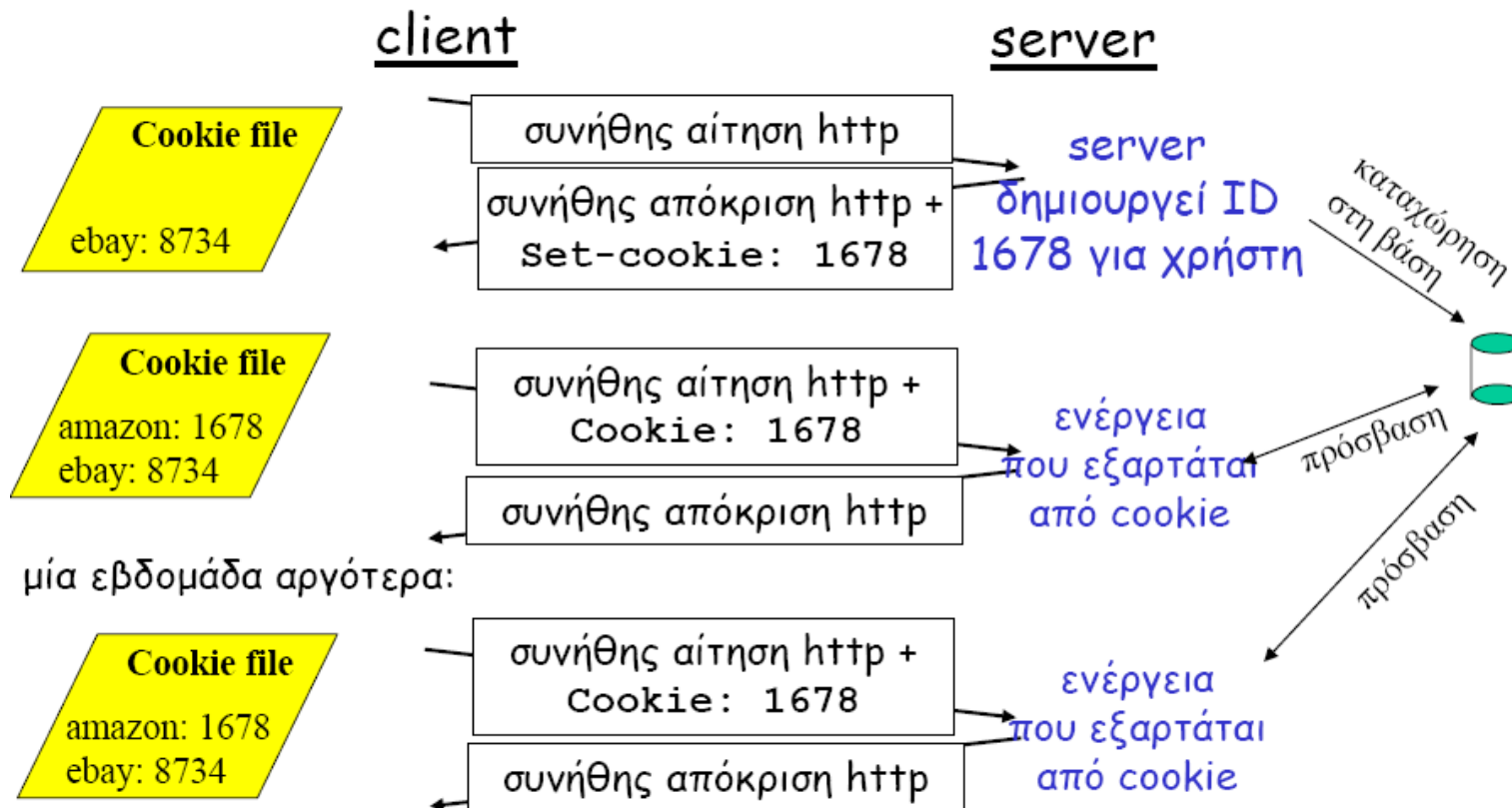
Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Cookies: διατήρηση «κατάστασης (state)»



- Πολλά μεγάλα Web sites χρησιμοποιούν cookies
- Τέσσερα στοιχεία:
 - Γραμμή επικεφαλίδας Set-cookie: στο μήνυμα απόκρισης HTTP
 - Γραμμή επικεφαλίδας Cookie: στο μήνυμα αίτησης HTTP
 - Διατηρείται αρχείο με cookies στον host του χρήστη το οποίο διαχειρίζεται ο browser
 - Διατηρείται βάση δεδομένων στο Web Site
- Παράδειγμα:
 - Ένας χρήστης χρησιμοποιεί πάντα το ίδιο PC για να συνδεθεί στο Διαδίκτυο
 - Επισκέπτεται ένα συγκεκριμένο e-commerce site για πρώτη φορά
 - Όταν η αρχική αίτηση HTTP φθάσει στο site, το site δημιουργεί ένα μοναδικό ID και επίσης μια καταχώρηση στη βάση δεδομένων για το ID αυτό

Cookies: διατήρηση «κατάστασης (state)»



Διαφάνεια 19

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Cookies συνέχεια



- Τα cookies μπορούν να χρησιμοποιηθούν για:
 - Εξουσιοδότηση
 - Shopping carts
 - Συστάσεις
- Cookies και απόρρητο:
 - Τα cookies επιτρέπουν στα sites να συλλέγουν πληροφορίες για τους χρήστες
 - Ενδέχεται ο χρήστης να παρέχει όνομα και e-mail στα sites.

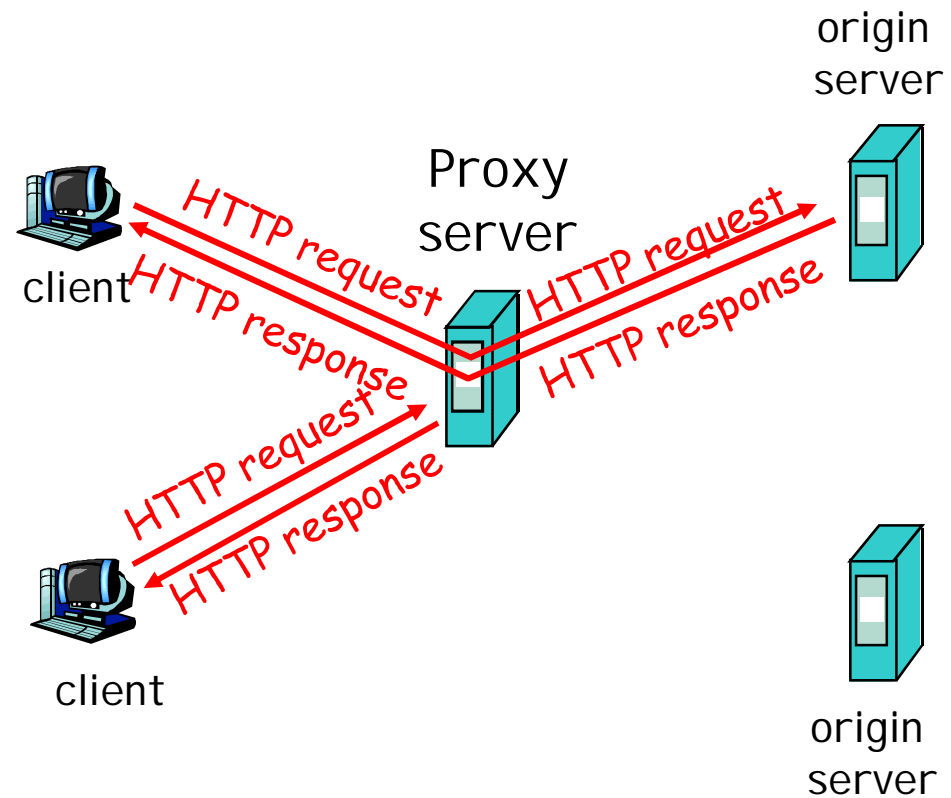
Διαφάνεια 20

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



Web caches (proxy server)

- Στόχος: να ικανοποιηθεί το request του χρήστη χωρίς να χρησιμοποιηθεί ο αρχικός server
- Ο χρήστης ρυθμίζει τον browser ώστε να έχει Web πρόσβαση μέσω cache
- Ο browser στέλνει όλα τα HTTP requests στην cache
 - Εάν το αντικείμενο υπάρχει cache: Η cache επιστρέφει το αντικείμενο
 - Διαφορετικά η cache ζητά το αντικείμενο από τον αρχικό server, και το επιστρέφει στον client



Διαφάνεια 21

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Περισσότερα για το Web caching

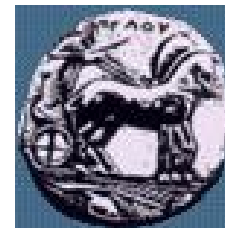


- Η Cache λειτουργεί τόσο ως client όσο και ως server
- Συνήθως η cache εγκαθίστανται από ISP (πανεπιστήμια, εταιρία, τοπικοί ISP)
- Γιατί χρησιμοποιούμε Web cache?
- Μειώνεται ο χρόνος απόκρισης στα client request.
- Μειώνεται η κίνηση στο access link του οργανισμού
- Η χρήση cache επιτρέπει την αποτελεσματικότερη διανομή περιεχομένου στο Διαδίκτυο

Διαφάνεια 22

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Παράδειγμα Caching

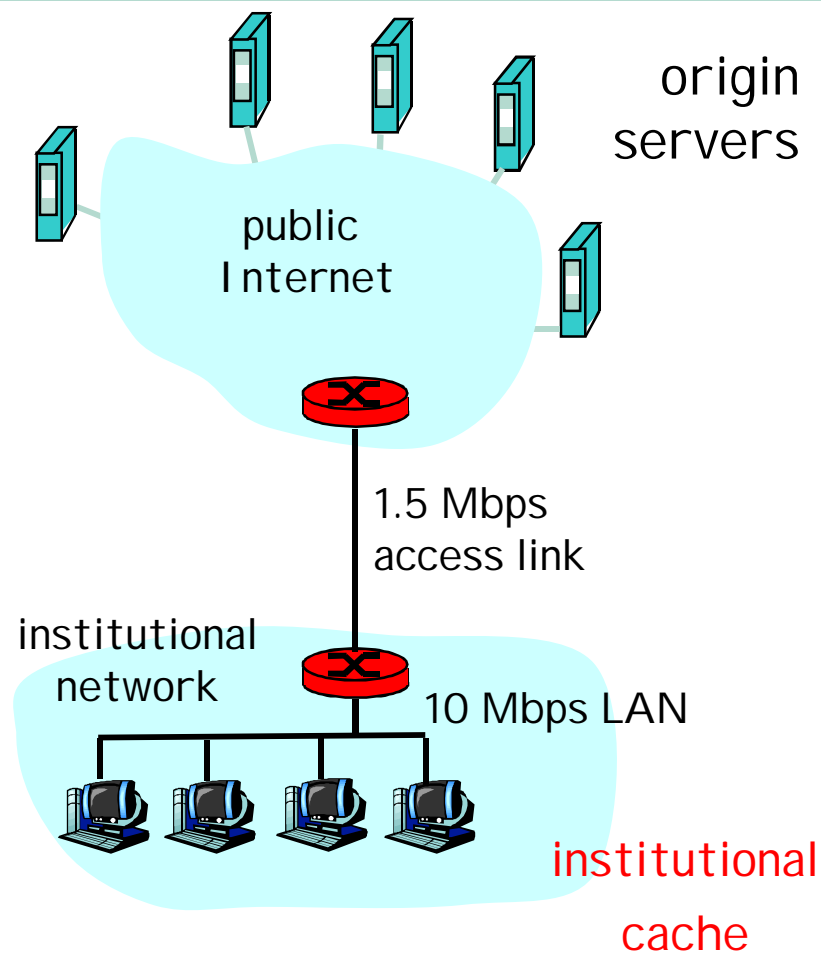


Υποθέσεις

- Μέσω μέγεθος αντικειμένου = 100,000 bits
- Μέσος ρυθμός request από browsers οργανισμού στους αρχικούς servers = 15 req/sec
- Καθυστέρηση από τον δρομολογητή του οργανισμού στους αρχικούς server και αντίστροφα = 2 sec

Συμπεράσματα

- Χρήση LAN = 15%
- Χρήση access link = 100%
- Συνολική καθυστέρηση: total delay = Internet delay + access delay + LAN delay
= 2 sec + milliseconds + milliseconds



Διαφάνεια 23

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Παράδειγμα Caching

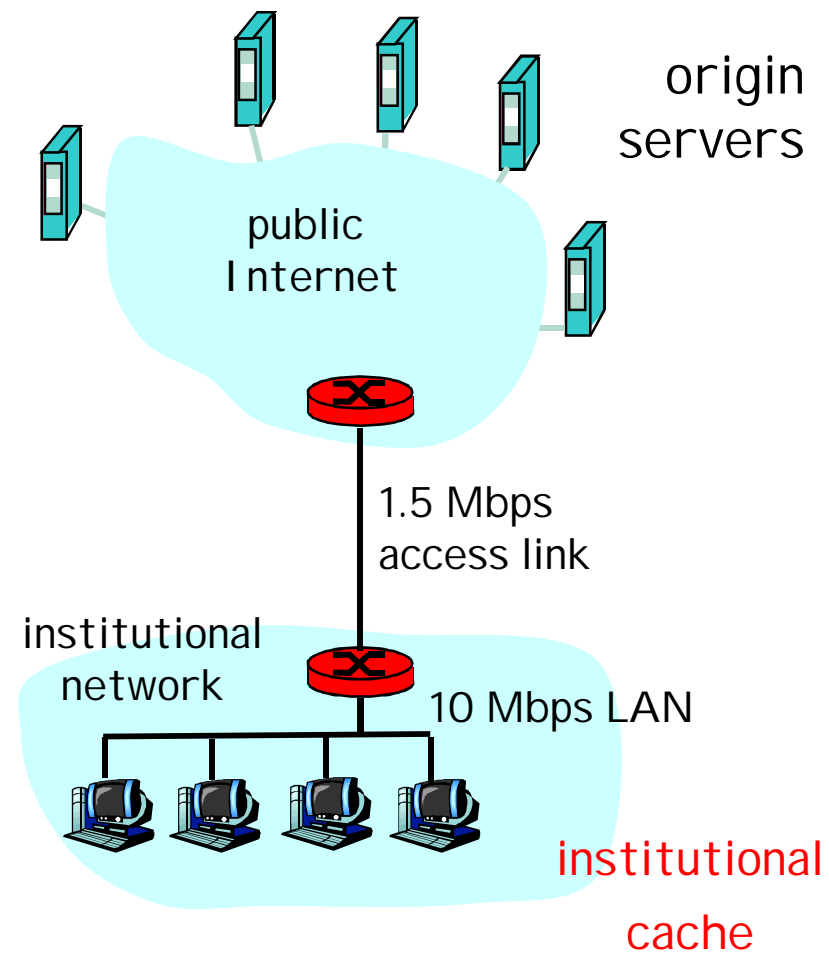


Πιθανή λύση

- Αύξηση bandwidth στο access link στα 10 Mbps

Συμπεράσματα

- Χρήση LAN = 15%
- Χρήση access link = 15%
- Συνολική καθυστέρηση Total delay = Internet delay + access delay + LAN delay
= 2 sec + msec + msec
- Αναβάθμιση η οποία κοστίζει



Διαφάνεια 24

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Παράδειγμα Caching

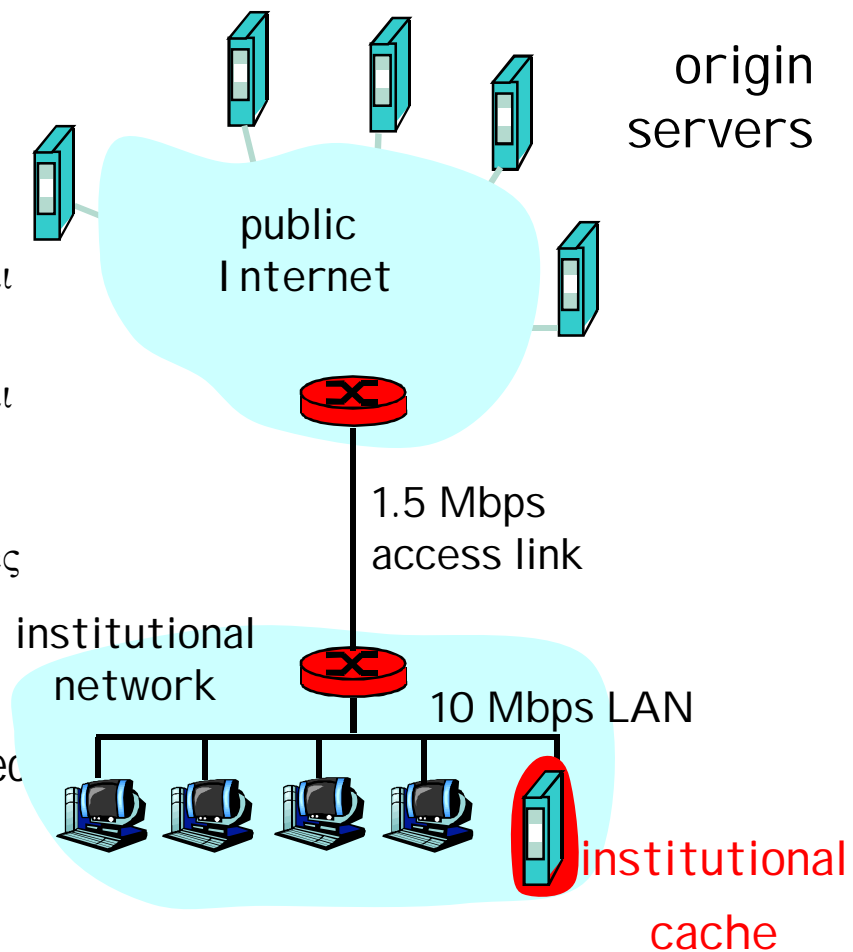


Εγκατάσταση cache

— Υποθέτουμε hit rate is .4

Συμπεράσματα

- 40% των requests θα ικανοποιούνται αμέσως
- 60% των requests θα ικανοποιούνται μέσα από τους αρχικούς server
- Η χρήση του access link μειώθηκε στο 60%, με αποτέλεσμα μικρότερες καθυστερήσεις (say 10 msec)
- Συνολική μέση καθυστέρηση: total avg delay = Internet delay + access delay + LAN delay = $.6 * (2.01) \text{ sec} + \text{milliseconds} < 1.4 \text{ secs}$



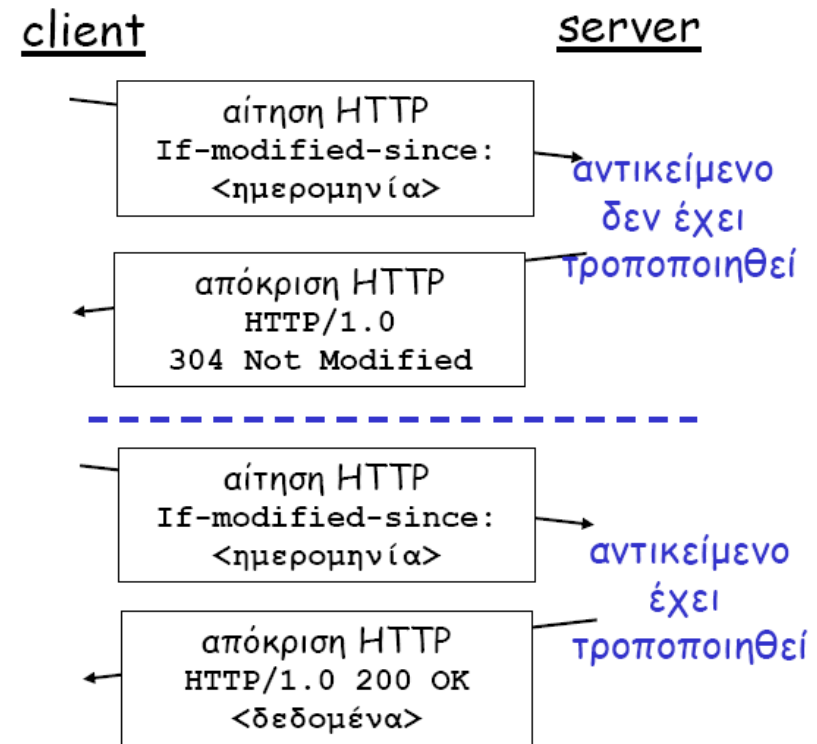
Διαφάνεια 25

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Conditional GET: caching στην πλευρά του client



- Στόχος: να μη σταλεί το αντικείμενο εάν ο client έχει ενημερωμένη έκδοση στην cache
- Client: καθορίζει ημερομηνία αντιγράφου στην cache στην αίτηση HTTP με τη γραμμή επικεφαλίδας If-modified-since: <ημερομηνία>
- Server: απόκριση δεν περιέχει αντικείμενο εάν το αντίγραφο στην cache είναι ενημερωμένο: HTTP/1.0 304 Not Modified



Διαφάνεια 26

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



Τι είναι ένας Web server

- Ένα πρόγραμμα (daemon, httpd) το οποίο:
 - Ανταποκρίνεται σε TCP connection και παρέχει υπηρεσίες σε ένα client
 - Εκτελείτε ανεξάρτητα
- Οι Web servers:
 - Δεν ελέγχουν τον HTML κώδικα
 - Δεν ελέγχουν τα links
- Web site = host + Web server + information (file system)

Διαφάνεια 27

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



Web server Λογισμικό

- Συνήθως διατίθεται ελεύθερα
- Διατίθεται για τις περισσότερες πλατφόρμες:
 - UNIX, Ms Windows, Macintosh, VMS, VM, ...
- Λίστα από διαθέσιμους Web Servers:
 - http://www.yahoo.com/Computers_and_Internet/Software/Internet/World_Wide_Web/Servers/
- Survey από Web Server
 - <http://www.netcraft.com/Survey/>
- Δημοφιλής Web Server:
 - CERN, NCSA (πρώτος web server)
 - Apache, MS IIS, Netscape servers, ...

Apache



- Βρίσκεται υπό συνεχόμενη ανάπτυξη
- Διατίθεται ελεύθερα:
 - Σε πηγαίο κώδικα
 - Σε δυαδικό για πολλές πλατφόρμες
 - Υποστηρίζει HTTP 1.1. από την έκδοση 1.2.
 - Αποτελείται από πολλά **modules** τα οποία επιλέγουμε κατά το installation/compile
- Χρήσιμες διευθύνσεις
 - Apache home: <http://www.apache.org/>
 - <http://www.apacheweek.com/>
 - support via Usenet group(s)





Που τοποθετούμε τους Web Servers?

- Προσεκτική επιλογή host
- Συνήθως δίνεται ένα DNS alias στο επιλεγμένο host (*www.mydomain.mycountry*)
- **ServerRoot**, **DocumentRoot** και **Log files directories** θα πρέπει να τοποθετούνται προσεκτικά με βάση τους κανόνες του daemon και τις απαιτήσεις σε χωρητικότητα δίσκου
- Θα υπάρχουν User Home Pages?
- Καθορισμός CGI rules!

Διαφάνεια 30

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



Δομή directory στον Apache

- Μπορεί να καθοριστεί/αλλαχθεί κατά την εγκατάσταση
- Μερικά σημαντικά directories:
 - cgi-bin/ - CGI scripts directory
 - conf/ - configuration αρχεία για τον httpd server
 - htdocs/ - Ο κύριος κατάλογος για τα αντικείμενα
 - logs/ - Ο κατάλογος με τα log files
 - Άλλοι κατάλογοι (bin/, man/...)



Configuration αρχεία του Apache

- Βρίσκονται στο `conf/` directory:
 - `access.conf` - access configuration
 - `httpd.conf` - server configuration
 - `mime.types` - MIME type
 - `srm.conf` - resource configuration
 - `*.*-dist` - distribution templates
- Από την έκδοση `v.1.3.6`. προτείνεται η χρήση μόνο του κυρίως `configuration file httpd.conf`

Configuration εντολές (directives) στον Apache



- Γενικοί κανόνες:
 - Case insensitive (δεν ισχύει για ονόματα αρχείων και καταλόγων σε linux/unix)
 - Οι γραμμές σχολίων ξεκινάν με #
 - Μια directive ανά γραμμή
 - Κάθε configuration αρχείο αποτελείται από:
 - directive data [data2 ... datan]
 - Επιπλέον κενός χώρος ο οποίο αγνοείται



Configuration αρχείο: httpd.conf

- ServerType standalone
- Port 80
- User nobody
- Group nogroup
- ServerAdmin your_e-mail_address
- ServerRoot /home/httpd/
- ErrorLog /home/httpd/logs/error_log
- TransferLog /home/httpd/logs/access_log
- PidFile /home/httpd/logs/httpd.pid
- Επιπλέον directives:
 - Keep Alive, Spare Servers, Proxy, Cache, Virtual Servers, ...

Configuration αρχείο: httpd.conf (srm.conf)



- DocumentRoot /home/httpd/htdocs/
- UserDir public_html
- DirectoryIndex index.html
- AccessFileName .htaccess
- DefaultType text/plain
- ScriptAlias /cgi-bin/ /home/httpd/cgi-bin/
- Επιπλέον directives:
 - Icons, Language, Handlers, ...

Configuration αρχείο: httpd.conf (access.conf)



- Ορίζει:
 - Τι είδους υπηρεσίες επιτρέπονται
 - Κάτω από ποιες συνθήκες
- `<Directory dir_name> ... directives ... </Directory>`
`<DirectoryMatch regex> ... directives ... </DirectoryMatch>`
`<Files file1 file2 > ... directives ... </Files>`
`<FilesMatch regex> ... directives ... </FilesMatch>`
- Πρέπει να δοθεί ιδιαίτερη προσοχή γιατί μπορεί να υπάρχουν προβλήματα:
 - Λειτουργικά
 - Ασφάλειας



Configuration αρχείο: mime.types

- Παρέχει τα MIME types τα οποία γνωρίζει ο server:
 - format: type/subtype file_extension
 - example: text/html html htm
 - image/gif gif
- Αρχεία με άλλη επέκταση θα δοθούν με το DefaultType
- Προσθέτουμε εγγραφές ανάλογα με τις ανάγκες μας

Διαφάνεια 37

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



Έναρξη και τερματισμός του Apache

- Απλή εκτέλεση του προγράμματος (apachectl start)
- Δυνατότητα έναρξης στο startup
- Επιλογές του apachectl : START, STOP, CONFIGTEST
- Για να σταματήσει ο server
 - kill εντολή (UNIX) (pid βρίσκεται στο αρχείο httpd.pid)
 - apachectl stop

Access control



- Σε δύο επίπεδα:
 - Ανά-server (Global Access Configuration file) - χρησιμοποιώντας directives στο httpd.conf (access.conf)
 - Ανά-directory (Per-directory Access Configuration file) - χρησιμοποιώντας .htaccess αρχεία (μπορεί να αλλάξει το όνομα αυτό με την AccessFileName directive στο httpd.conf (srm.conf))
- Δύο τρόποι:
 - με user/password
 - με host/domain

Πιστοποίηση με User/password



- Πρέπει να δημιουργηθεί ένα αρχείο με το όνομα `.htaccess` στον κατάλογο τον οποίο θέλουμε να προστατεύσουμε

```
AuthUserFile /home/httpd/admin/.htpasswd
AuthGroupFile /dev/null
AuthName ByPassword
AuthType Basic
<Limit GET>
require user username
</Limit>
```





User/password authentication

- Χρησιμοποιώντας την εντολή `htpasswd` μπορεί να δημιουργηθεί το αρχείο με τα `password`:
 - `htpasswd -c /home/httpd/bin/.htpasswd username`
- Δώστε το `password` το οποίο θέλετε
- Μπορεί να προστεθεί `password` στο `.htpasswd` αρχείο με προγραμματισμό στην πλευρά του εξυπηρετητή

Διαφάνεια 41

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

User/password authentication



- Ο server ζητά από τον browser για user/password για να επιτρέψει πρόσβαση
- Το password μεταφέρεται πάνω από το δίκτυα μη κρυπτογραφημένο
- Η μέθοδος αυτή δεν παρέχει υψηλή ασφάλεια όσο αφορά την προστασία του password

Διαφάνεια 42

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



Host/domain authentication

- Χρησιμοποιούμε πάλι `.htaccess` αρχείο με διαφορετική σύνταξη

```
<Limit GET>  
  order deny,allow  
  deny from all  
  allow from hostname/domain  
</Limit>
```





Έλεγχος πρόσβασης

- Υπάρχει δυνατότητα συνδυασμού των μεθόδων πιστοποίησης (και με user/password και με host)
- Για αυξημένη ασφάλεια θα πρέπει να γίνει χρήση του SSL (Secure Socket Layer) module στο Apache

Διαφάνεια 44

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Common Gateway Interface (CGI)



- Οι WWW server έχουν την δυνατότητα να επικοινωνούν με άλλα προγράμματα (CGI scripts)
- Τα CGI μπορούν να γραφούν σε οποιαδήποτε γλώσσα προγραμματισμού (shell script, PERL, C, ...)
- Χρησιμοποιούνται για προγραμματισμό στην πλευρά του εξυπηρετητή

Διαφάνεια 45

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών



CGI



- Ο server πρέπει να ρυθμιστεί ώστε να επιτρέπει την εκτέλεση των CGI:
 - ScriptAlias /cgi-bin/ /home/httpd/cgi-bin/
- Όλα τα αρχεία στο /cgi-bin/ θεωρούνται εκτελέσιμα ανεξάρτητα από το ονομά τους
- Θα πρέπει να λαμβάνονται μέτρα ασφαλείας (πχ έλεγχος των scripts των χρηστών)

Διαφάνεια 46

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών

Virtual servers



- Ένας server μπορεί να «ακούει» σε πολλά hosts names – οπότε έχουμε virtual servers (ίδια port, διαφορετικά hostnames)
- Αυτό ορίζεται στο server configuration (httpd.conf)
<VirtualHost hostname> ... </VirtualHost>
- Κάθε virtual server μπορεί να έχει τελείως διαφορετικό περιεχόμενο, configuration, log και error files
- Εναλλακτικά μπορούμε να τρέξουμε ένα ξεχωριστό server σε διαφορετικό port



Καταγραφή – logs του Apache

- Χρησιμοποιούνται από content providers:
 - usability testing
 - Εντοπισμός προβλημάτων απόδοσης
 - Tuning του server software / μέτρηση απόδοσης (benchmarking)
- Χρησιμοποιούνται από network providers
 - Αξιολόγηση proxies
 - Εντοπισμός προβλημάτων απόδοσης
- Χρησιμοποιούνται από protocol developers
 - Μέτρηση απόδοσης πρωτοκόλλων (DNS, TCP, HTTP)
 - Αξιολόγηση αλλαγών σε μηχανισμούς

Server logs



- Ο servers καταγράφει πληροφορίες για
 - client host,
 - ημερομηνία,
 - client request,
 - status,
 - Αριθμός από bytes τα οποία στάλθηκαν
 - ...
- Τα log αρχεία έχουν καθορισμένη δομή
- Είναι δυνατόν να δημιουργηθούν αναφορές από τα log αρχεία
- Υπάρχει μεγάλος αριθμός από log analyzers (wwwstat, analog,...)

Μερικά αποτελέσματα



- Από διάφορες μελέτες προκύπτει ότι:
 - Ο μέσος όρος του αντικειμένου είναι μικρός (e.g. 8-10 KB)
 - 10% των αντικειμένων καταναλώνουν το 90% της κίνησης
 - Κατά μέσω όρο 3-5 αντικείμενα ανά Web σελίδα (μαζί με το HTML αρχείο)

Διαφάνεια 50

Υλοποίηση Δικτυακών Υποδομών και Υπηρεσιών