Καστίδου Γεωργία Α.Μ. 42
Πετράκης Ιωάννης Α.Μ. 46

Κατανεμημένα Συστήματα
Εργασία 1 (3/3/2003)

Άσκηση 1. Read Section 1.3.1 of Coulouris et. al or Sec 11.1 of Tanenbaum et al. (on the web). Web can be considered as an example of a distributed system. At what extend does the web satisfies the four goals of distributed systems that were set in the lecture? In particular:

Goal 1: connecting users and resources
What are the types of resources shared? How does web address the two problems of

Goal 2: For each type of transparency, explain the transparency degree offered by the web.

Goal 3: To what extent is web an open system? Briefly mention the protocols used by the web. Give an example of a web policy and a web mechanism (other than caching).

Goal 4: Is web scalable (in each of the three dimensions)? How is scalability at each dimension achieved, that is, which are the scalability techniques used along each

Λύση:

Goal 1

Κάθε εξυπηρέτης του web διατηρεί συλλογή από έγγραφα, όπου κάθε έγγραφο αποθηκεύεται σε ένα αρχείο. Όταν δέχεται αιτήσεις για κάποιο έγγραφο μεταφέρει το έγγραφο αυτό στον πελάτη, ο οποίος έκανε την αίτηση. Επομένως οι διαμοιραζόμενοι πόροι είναι έγγραφα. Τα έγγραφα αυτά μπορεί να είναι HTML (Hypertext Metadata Language), XML (Extensive Markup Language), κάποιο script, καθώς επίσης και έγγραφα σε μορφή PDF, JPEG, GIF, MP3, MPEG κτλ..

Ένα βασικό πρόβλημα στην επικοινωνία μεταξύ κόμβων σε κάθε κατανεμημένο σύστημα είναι η ασφάλεια και η ακεραιότητα της μεταδιδόμενης πληροφορίας. Με τον όρο ασφάλεια εννοούμε την προστασία της μεταδιδόμενης πληροφορίας από μη πιστοποιημένους παραλήπτες, δηλαδή παραλήπτες που δεν θα έπρεπε να έχουν πρόσβαση σε αυτήν την πληροφορία. Με τον όρο ακεραιότητα εννοούμε την αυτούσια μεταφορά της πληροφορίας. Ο τρόπος με τον οποίο εξασφαλίζεται στο web η ασφαλή

και ακέραια μεταφορά δεδομένων είναι με την εγκατάσταση ασφαλούς καναλιού μεταξύ του εξυπηρέτη και του πελάτη. Αυτό πραγματοποιείται με τη χρήση του SSL (Secure Socket Layer) ή του TSL (Transport Layer Security), τα οποία δεν είναι τίποτα άλλο παρά πρωτόκολλα που εξασφαλίζουν την ασφαλή επικοινωνία και την πιστοποίηση των κόμβων που παίρνουν μέρος σε αυτήν.

΄Ενα άλλο πρόβλημα είναι η ανεπιθύμητη επικοινωνία, όπως για παράδειγμα η παραλαβή ανεπιθύμητων e-mail. Ο βασικότερος τρόπος προστασίας που προσφέρεται είναι η χρήση φίλτρου, το οποίο ελέγχει τα μηνύματα και επιτρέπει τη μετάδοση μόνο εκείνων που πρέπει.

Goal 2

Διαφάνεια πρόσβασης (access transparency).
Διαφάνεια πρόσβασης πετυχαίνεται με τη χρήση μιας ειδικής γλώσσας (HTML) η οποία είναι κατανοητή από τους browsers, ώστε να εμφανίσουν κάθε σελίδα στην οθόνη οποιουδήποτε πελάτη ανεξάρτητα από τη μηχανή ή το λειτουργικό σύστημα που χρησιμοποιεί.

Διαφάνεια τοποθέτησης (location transparency).
Το web προσφέρει διαφάνεια τοποθέτησης με τη χρήση ονομάτων τα οποία ακολουθούν τη μορφή URL για την αναφορά σε σελίδες (όπως για παράδειγμα www.ttttt.com). Το σχήμα αυτό για την απόδοση ονομάτων αποκρύπτει την τοποθεσία της σελίδας.

Διαφάνεια μετανάστευσης (migration transparency).
Το URL προσφέρει και διαφάνεια μετανάστευσης αφού δεν δίνει πληροφορία για το αν κάποιο έγγραφο μετακινήθηκε πρόσφατα στην τωρινή του θέση ή ήταν πάντα εκεί.

Διαφάνεια ομοιοτυπίας (replication transparency).
Επίσης το web έχει διαφάνεια ομοιοτυπίας γιατί δεν δίνεται πληροφορία για το που βρίσκονται τα διαφορετικά αντίτυπα ενός εγγράφου, αλλά προσπελαύνονται όλα με τη χρήση ίδιου URL.

Διαφάνεια αποτυχίας (failure transparency).
Όσον αφορά τη διαφάνεια αποτυχίας, αυτή προσφέρεται σε πολύ μικρό βαθμό αφού αν αναζητηθεί κάποιος πόρος ο οποίος προσωρινά δεν μπορεί να προσφερθεί, τότε ο web browser είτε εμφανίζει άμεσα μήνυμα αποτυχίας, είτε αργεί πάρα πολύ να αποκριθεί συνεχίζοντας την προσπάθεια σύνδεσης με τον πόρο.

Διαφάνεια ταυτοχρονισμού (concurrency transparency).
Το web προσφέρει διαφάνεια ταυτοχρονισμού όσον αφορά δεδομένα, τα οποία είναι μόνο προς ανάγνωση. Στην περίπτωση αυτή ο εξυπηρέτης μπορεί να παρέχει στον πελάτη τα δεδομένο που έχει ζητήσει χωρίς ο πελάτης να γνωρίζει σε πόσους ταυτόχρονα παρέχει υπηρεσίες ο εξυπηρέτης. Η μόνη περίπτωση να «καταλάβει» ένας πελάτης ότι μπορεί ο εξυπηρέτης να παρέχει ταυτόχρονα σε περισσότερους κόμβους

πληροφορία είναι αν παρατηρήσει καθυστέρηση στις απαντήσεις του εξυπηρέτη, ωστόσο όμως δεν μπορεί να είναι σίγουρος ότι η καθυστέρηση οφείλειται σε φόρτο εργασίας του εξυπηρέτη λόγω ταυτόχρονης εξυπηρέτησης άλλων πελατών.
Όσον αφορά δεδομένα τα οποία βρίσκονται σε έναν εξυπηρέτη και μπορεί να τροποποιθούν από τους πελάτες, αν ο εξυπηρέτης ακολουθεί κάποια πολιτική κελιδώματος των δεδομένων για να αποτρέψει ταυτόχρονη μετατροπή τους από διάφορους πελάτες τότε στην περίπτωση που ένας πελάτης προσπαθεί να προσπελάσει ένα άρχειο που είναι κλειδωμένο διότι εκείνη τη στιγμή τροποποιείται από κάποιον άλλο πελάτη, ο εξυπηρέτης του επιστρέφει μήνυμα ειδοποιώντας τον ότι θα πρέπει να περιμένει.

Διαφάνεια μετακίνησης (relocation transparency).
Το web προσφέρει διαφάνεια μετακίνησης στην περίπτωση που θεωρήσουμε πως η πληροφορία που θέλουμε να προσπελάσουμε βρίσκεται σε έναν φορητό υπολογιστή (εξυπηρέτης), ο οποίος μετακινείται και συνδέεται στο δίκτυο μέσω ασύρματου δικτύου. Στην περίπτωση αυτή ανεξάρτητα από την κάθε φορά θέση του εξυπηρέτη έχουμε πρόσβαση στην πληροφορία που θέλουμε χωρίς να χρειάζεται να γνωρίζουμε τις «κινήσεις» του εξυπηρέτη.
Ωστόσο στην περίπτωση που η πληροφορία που θέλουμε να προσπελάσουμε βρίσκεται σε κάποιον εξυπηρέτη το όνομα του οποίου περιέχει γεωγραφική πληροφορία, τότε η μετακίνηση του μπορεί να έχει ως αποτέλεσμα την αλλαγή του ονόματος του εξυπηρέτη. Στην περίπτωση αυτή η μετακίνηση του γίνεται αντιληπτή από τους πελάτες.

Διαφάνεια (persistent transparency).
Το web χαρακτηρίζεται από persistent διαφάνεια και αυτό γιατί όταν ένας πελάτης ζητήσει κάποιο αρχείο από ένα εξυπηρέτη, ο πελάτης δεν είναι σε θέση να γνωρίζει αν το αρχείο βρισκόταν σε κάποιο δίσκο ή στην μνήμη του εξυπηρέτη.

Goal 3.

Το web ανήκει στην κατηγορία των open κατανεμημένων συστημάτων γιατί προσφέρει υπηρεσίες σύμφωνα με καθορισμένους κανόνες, οι οποίοι περιγράφουν τη σύνταξη και το αποτέλεσμα εκτέλεσης (δηλαδή τι ακριβώς κάνουν) των παραπάνω υπηρεσιών, επιτρέποντας έτσι την ανάπτυξη μηχανισμών για την επικοινωνία των διάφορων εφαρμογών. Οι κανόνες αυτοί είναι γνωστοί με τον όρο πρωτόκολλα.

Στο web έχει οριστεί μια πληθώρα πρωτοκόλλων, τα οποία περιγράφουν τους κανόνες που πρέπει οι διάφορες εφαρμογές να ακολουθούν, ώστε να είνα όλες συμβατές μεταξύ τους. Τα κυριότερα από αυτά είναι:

- HTTP: Hypertext Transfer Protocol

Το http δεν είναι τίποτα άλλο παρά ένα απλό πελάτη – εξυπηρέτη πρωτόκολλο, όπου ο πελάτης στέλνει μια αίτηση στον εξυπηρέτη και περιμένει για την απάντηση. Το βασικό χαρακτηριστικό του http είναι πως δεν χρησιμοποιεί ανοιχτές συνδέσεις και δεν απαιτεί

από τον εξυπηρέτη να διατηρεί πληροφορίες στον πελάτη. Ένα επιπλέον χαρακτηριστικό του http είναι η διαπραγμάτευση της αναπαράστασης των δεδομένων, η οποία επιτρέπει τα διάφορα συστήματα να σχεδιάζονται ανεξάρτητα από την ανάπτυξη νέων αναπαραστάσεων.

- FTP: File Transfer Protocol

Το πρωτόκολλο μεταφοράς δεδομένων που χρησιμοποιείται κυρίως για την πρόσβαση σε κοινόχρηστη πληροφορία είναι το "Anonymous FTP". Το πρωτόκολλο αυτό είναι ένα πρωτόκολλο μεταφοράς δεδομένων στο web, το οποίο όμως δεν εξασφαλίζει την ασφαλή μεταφορά τους.

- SSL - Secure Sockets Layer , TSL - Transport Layer Security

Το SSL είναι ένα πρωτόκολλο που αναπτύχθηκε από την Netscape για την ιδιωτική μετάδοση αρχείων κειμένου μέσω του web. Το SSL λειτουργεί με τη χρήση δημοσίου κλειδιού για την κρυπτογράφηση δεδομένων τα οποία μεταδίδονται μέσω της SSL σύνδεσης.

- https: Secure Hypertext Transfer Protocol

Το https είναι ένα πρωτόκολλο επικοινωνίας σχεδιασμένο για τη μεταφορά κρυπτογραφημένων πληροφοριών μεταξύ των υπολογιστών του web. Το https είναι ουσιαστικά το πρωτόκολλο http με χρήση του SSL. Το SSL είναι ένα πρωτόκολλο κρυπτογράφησης, το οποίο συμπεριλαμβάνεται σε έναν Web εξυπηρέτη που χρησιμοποιεί το https.

- IMAP, Version4rev1: Internet Message Access Protocol

Το πρωτόκολλο IMAP4rev1 επιτρέπει σε έναν πελάτη να αποκτήσει πρόσβαση και να διαχειριστεί την ηλεκτρονική αλληλογραφία που βρίσκεται αποθηκευμένη σε κάποιον εξυπηρέτη. Συγκεκριμένα το IMAP4rev1 επιτρέπει τη διαχείριση ενός καταλόγου μηνυμάτων που βρίσκεται σε κάποιον απομακρυσμένο κόμβο, οι οποίοι αποκαλούνται «γραμματοκιβώτια» (mailboxes), με τρόπο ισοδύναμο της διαχείρισης τοπικών γραμματοκιβωτίων. Επιπλέον το πρωτόκολλο αυτό παρέχει τη δυνατότητα σε πελάτες που ήταν εκτός δικτύου να επανασυγχρονιστούν με τον εξυπηρέτη.

- SMTP: Simple Mail Transfer Protocol

Το SMTP είναι ένα απλό ASCII πρωτόκολλο αποστολής ηλεκτρονικής αλληλογραφίας μηνυμάτων ανάμεσα σε απομακρυσμένους κόμβους. Ύστερα από την εγκατάσταση της TCP επικοινωνίας στο port 25, ο αποστολέας, ο οποίος είναι ο πελάτης, περιμένει να απαντήσει πρώτος ο παραλήπτης, ο οποίος λειτουργεί σαν εξυπηρέτης,. Ο εξυπηρέτης αρχίζει στέλνοντας μία γραμμή κειμένου, με την οποία ενημερώνει τον αποστολέα για την ταυτότητα του καθώς επίσης αν είναι έτοιμος να δεχτεί μηνύματα. Αν ο παραλήπτη δεν είναι ετοιμος να δεχτεί κάποιο μήνυμα, ο αποστολές ακυρώνει την επικοινωνία και προσπαθεί αργότερα.

- TCP: Transmission Control Protocol

Το TCP είναι ένα πρωτόκολλο ελέγχου αποστολής δεδομένων το οποίο αποτελείται από ένα σύνο κανόνων και το οποίο χρησιμοποιείται για την αποστολή δεδομένων με μορφή μηνυμάτων μεταξύ υπολογιστών. Κάθε μήνυμα που μεταδίδεται ανάλογα με το μέγεθος του και με το μέγεθος των πακέτων που επιτρέπει κάθε router από όπου περνάει προκειμένου να φτάσει στον προορισμό του σπάει σε πακέτα. Το TCP ουσιαστικά είναι υπεύθυνο να μπορεί να εντοπίζει τα πακέτα αυτά.

- MIME

Στο RFC 822 ορίζεται ένα πρωτόκολλο αναπαράστασης μηνυμάτων, το οποίο καθορίζει με μεγάλη λεπτομέρεια τις επικεφαλίδες των μηνυμάτων αλλα αφήνει το περιεχόμενο του μηνύματος σαν απλό ASCII κείμενο. Το MIME επαναρίζει τη μορφή του περιεχομένου του μηνύματις ώστε να επιτρέπεται το περιεχόμενο ενός μηνύματος να είναι multi-part textual και non-textual χωρίς να υπάρχει κίνδυνος απώλειας πληροφορίας.

- Telnet

Το πρωτόκολλο αυτό είναι ένα πρωτόκολλο για πρόσβαση σε απομακρυσμένους κόμβους, το οποίο όμως δεν εξασφαλίζει ούτε την πιστοποίηση των δύο άκρων της επικοινωνίας αλλά ούτε και την ασφαλή – κρυπτογραφημένη μετάδοση της επικοινωνίας.

Παράδειγμα web policy:

Καθορισμός δικαιωμάτων εκτέλεσης και πρόσβασης κινητών προγραμμάτων (π.χ. κινητών πρακτόρων) σε κάθε κόμβο του web.

Παράδειγμα web mechanism:

Μεταφορά προγραμμάτων (applets) από τον εξυπηρέτη στον πελάτη και εκτέλεση τους.

Goal 4.

Το web έχει ικανότητα κλιμάκωσης στο μέγεθος του (1η διάσταση), το οποίο πετυχαίνεται με την κατανομή των τεράστιων ποσοτήτων από έγγραφα σε πολλούς web εξυπηρέτες καθώς επίσης και με την δημιουργία αντιγράφων. Η κατανομή και η χρήση αντιγράφων των δεδομένων έχει σαν σκοπό την αποφυγή της συμφόρησης. Σε αυτό επίσης βοηθάει και η διαφάνεια ομοιοτυπίας του web, αφού μπορεί να υπάρχουν αντίγραφα του ίδιου πόρου σε διαφορετικούς κόμβους και να προωθούνται οι αιτήσεις των χρηστών ανάλογα με το φόρτο των κόμβων αυτών.

Επίσης το web προσφέρει γεωγραφική κλιμάκωση (2η διάσταση), η οποία σχετίζεται με τη διαφάνεια ομοιοτυπίας που παρέχεται. Μπορεί δηλαδή να τοποθετηθεί ένα αντίγραφο σε κάθε περιοχή, ώστε να αποφεύγονται μεγάλες καθυστερήσεις στην επικοινωνία και τη

μεταφορά δεδομένων λόγω της μεγάλης γεωγραφικής απόστασης. Ακόμη, η ύπαρξη πολλών web servers δίνει τη δυνατότητα επιλογής του κοντινότερου δυνατού για την εξυπηρέτηση του χρήστη. Δεν είναι δηλαδή αναγκασμένος να απευθυνθεί σε ένα μοναδικό εξυπηρέτη που μπορεί να βρίσκεται πολύ μακριά.

Τέλος μέχρι κάποιο βαθμό παρέχεται και administratively scaling η οποία βασίζεται στην χρήση των πρωτοκόλλων SSL και TSL τα οποία εξασφαλίζουν ακεραιότητα των μηνυμάτων και πιστοποίηση των άκρων κατά την επικοινωνία.

Άσκηση 2: *Read the following paper: Jerome H. Saltzer, David P. Reed, and David D. Clark, "End-To-End Arguments In System Design", ACM Transactions on Computer Systems, Vol. 2, No. 4, Nov 1984,p. 277-288 (online copy at the course webpage).*
(a) Write a short summary of the paper including the main points made by the authors.
(b) Give arguments against the end-to-end argument.
(c) Provide examples of distributed systems that violate the end-to-end argument.

Λύση:

a) Στην παραπάνων εργασία παρουσιάζεται μια μέθοδος σχεδίασης δικτυκών συστημάτων, η οποία μελετάει το επίπεδο στο οποίο θα πρέπει να υλοποιηθούν οι διάφορες συναρτήσεις του συστήματος, ενώ επιπλέον συγκρίνει την παραπάνω μέθοδο με τη μέθοδο υλοποίησης η οποία θέλει οι συναρτήσεις να υλοποιούνται σε χαμηλότερα επίπεδα. Η μέθοδος που παρουσιάζεται στην παραπάνω εργασία ονομάζεται "end - to - end argument" και η βασική της ιδέα προκύπτει από την παρατήρηση πως σε πολλές περιπτώσεις η χρήση συναρτήσεων σε χαμηλότερα επίπεδα δεν είναι αποδοτική.

Σε πολλές εφαρμογές η υλοποίηση όλων των συναρτήσεων σε χαμηλό επίπεδο δεν είναι δυνατή λόγω της φύσης του προβλήματος που επιλύουν ενώ παράλληλα σε πολλές περιπτώσεις κάποιες από τις συναρτήσεις, οι οποίες είναι απαραίτητο να υλοποιηθούν στο επίπεδο εφαρμογής, επικαλύπτουν συναρτήσεις, οι οποίες θα μπορούσαν να υλοποιηθούν αποδοτικά σε χαμηλότερο επίπεδο.

Επίσης στην περίπτωση των εφαρμογών όπου είναι δυνατή η εξολοκλήρου υλοποίηση του συνόλου των συναρτήσεων σε χαμηλό επίπεδο δεν είναι δεδομένο πως το σύστημα θα είναι πιο αποδοτικό από ότι στην περίπτωση που οι συναρτήσεις αυτές υλοποιούνταν στο επιπέδο εφαρμογής. Αυτό συμβαίνει είτε γιατί οι συναρτήσεις που βρίσκονται σε χαμηλό επίπεδο δεν έχουν πρόσβαση σε όλη την πληροφορία της εφαρμογής, με αποτέλεσμα ενδεχομένως να μην είναι αποδοτικές, είτε γιατι «αλλάζοντας» βασικές συναρτήσεις το χαμηλό επίπεδο αναγκάζουμε όλες τις εφαρμογές, οι οποίες τροποποήθηκαν με τέτοιο τρόπο, ώστε να εξυπηρετούν κάποια συγκεκριμένη εφαρμογή.

Η σύγκριση των δύο μεθόδων γίνεται σε ένα σύνολο γνωστών προβλημάτων που παρουσιάζονται σε εφαρμογές όπου απαιτείται η ασφαλής μετάδοσης δεδομένων, όπως κρυπτογράφηση, ανίχνευση διπλότυπων μηνυμάτων, σειριακή παραλαβή μηνυμάτων, εγγυημένη παράδοση μηνυμάτων, ανίχνευση κατάρρευσης κόμβων κα. Ενώ επιπλέον γίνεται λόγος και για διάφορα συστήματα, τα οποία χρησιμοποιούν την end – to – end argument μέθοδο.

Τέλος η μέθοδος end – to – end argument όπως τονίζουν και οι συγγραφείς δεν είναι κάποιος απόλυτος κανόνας απλά είναι μια στρατηγική σχεδίασης εφαρμογών και προτοκόλων, η οποία πρέπει να προσαρμόζεται κάθε φορά στα ιδιαίτερα χαρακτηριστικά των εφαρμογών ή προτοκόλων.

b) Μειονεκτήματα end – to end argument

Ένα από τα βασικά μειονεκτήματα του end – to – end argument είναι η ανάθεση πολλών λειτουργιών στις εφαρμογές, με αποτέλεσμα να αυξάνεται ο φόρτος των εργασιών του χρήστη και κατεπέκτασιν να αυξάνεται και ο χρόνος απόκρισης της εφαρμογής.

‘Ενα άλλο μειωνέκτημα είναι πως στην περίπτωση που οι εφαρμογές έχουν κοινές συναρτήσεις η εφαρμογή του end – to – end argument απαιτεί την τοποθέτηση των συναρτήσεων (και πιθανώς τον επαναπρογραμματισμό τους) σε κάθε εφαρμογή.

Τέλος, η ανάγκη για επικοινωνία μεταξύ των εφαρμογών και η τοποθέτηση των συναρτήσεων σε κάθε μια από αυτές απαιτεί την ύπαρξη συμβατότητας μεταξύ των συναρτήσεων. Επομένως απαιτείται ο ορισμός προτύπων για την επικοινωνία μεταξύ των εφαρμογών.

c) Ένα παράδειγμα κατανεμημένου συστήματος, το οποίο παραβιάζει το end to end argument είναι το internet.