

Δίκτυα Υπολογιστών

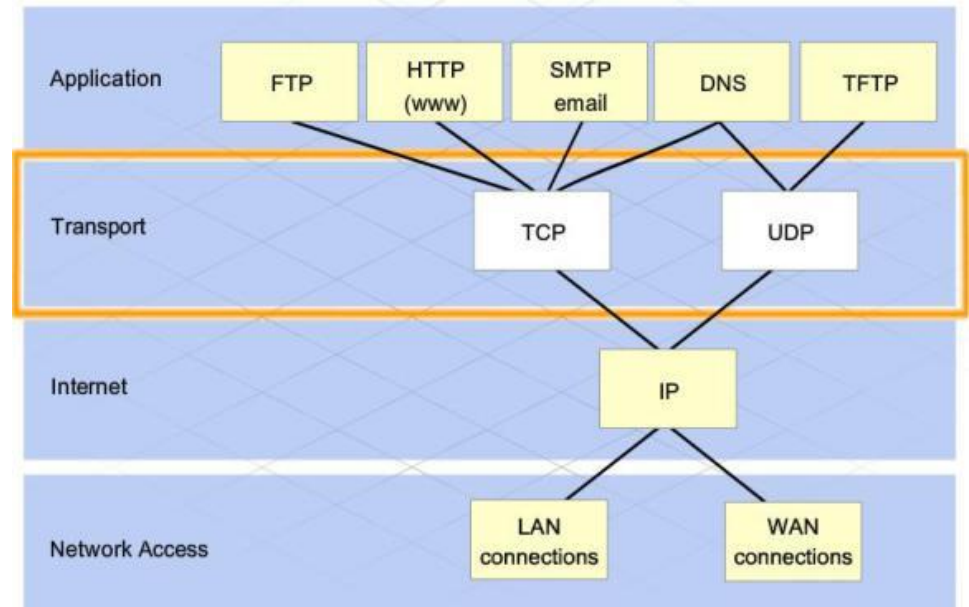


*Το επίπεδο μεταφοράς
(transport layer)*

Κ. Βασιλάκης

Περίγραμμα – ενότητες που εξετάζονται

- Επισκόπηση του επιπέδου μεταφοράς
- Υπηρεσίες επιπέδου μεταφοράς
- Επίπεδο μεταφοράς έναντι επιπέδου δικτύου
- Πολύπλεξη και αποπολύπλεξη
- Το πρωτόκολλο UDP
- Το πρωτόκολλο TCP

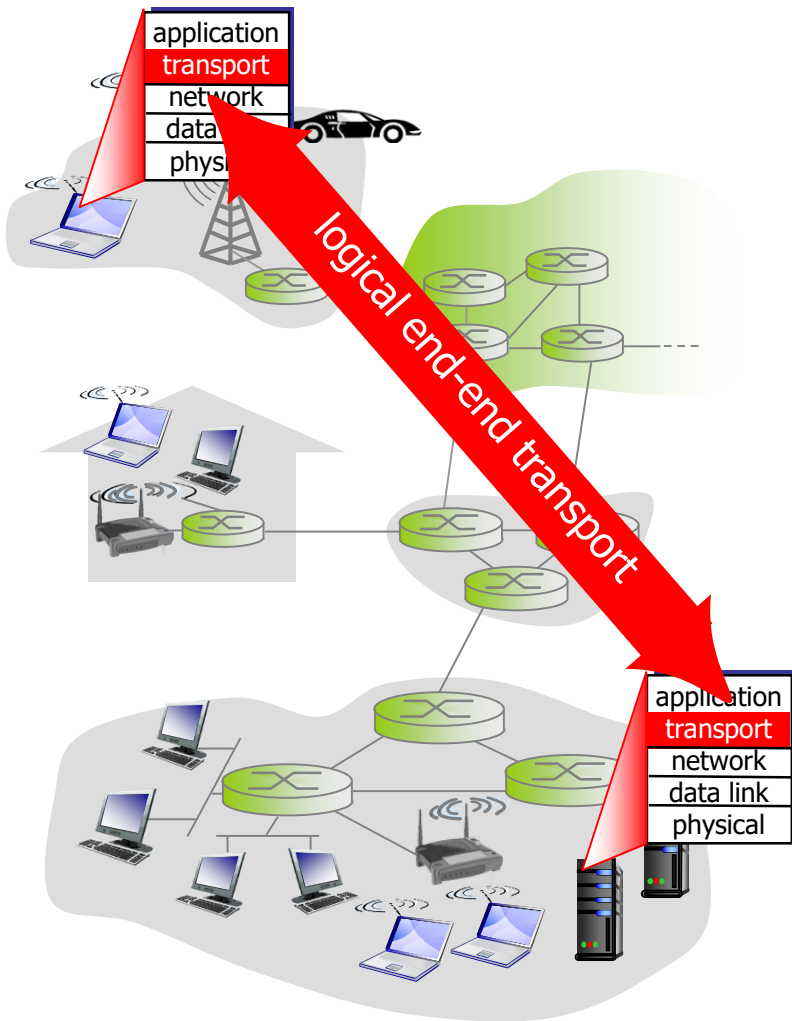


Επισκόπηση του επίπεδου μεταφοράς

- Αναλαμβάνει την ανταλλαγή μηνυμάτων σε μια κατανεμημένη εφαρμογή και φροντίζει για την επιτυχημένη *ποιοτική και ποσοτική* μεταφορά τους από τον κόμβο αποστολής στο κόμβο προορισμού (άκρο-σε-άκρο).
- Μεριμνά για *κατάτμηση* των μηνυμάτων του επιπέδου εφαρμογής σε πακέτα (στη αποστολή) και την επανένωση τους (στη παραλαβή).
- Προωθεί τα πακέτα με την σωστή σειρά στα κατώτερα επίπεδα.
- Κατά την παραλαβή των πακέτων γίνεται έλεγχος για την αριτιότητα τους και για την ορθή ανασύνθεση τους, πριν προωθηθούν στο ανώτερο επίπεδο (εφαρμογής).
- Τα πακέτα που δημιουργούνται σε αυτό το επίπεδο ονομάζονται:
τμήματα (segments).



Αρχιτεκτονική λογικής επικοινωνίας



- Η μεταφορά μηνυμάτων γίνεται ανεξάρτητα από τα υποκείμενα επίπεδα.
- Τα πρωτόκολλα του επιπέδου μεταφοράς παρέχουν τη δυνατότητα *λογικής επικοινωνίας* (logical communication) μεταξύ διεργασιών εφαρμογών που τρέχουν σε διαφορετικά τερματικά συστήματα (άκρα).
- Οι διεργασίες συμπεριφέρονται σαν να είναι τα hosts που τις φιλοξενούν, απ' ευθείας συνδεδεμένα μεταξύ τους.

Λογική επικοινωνία

Τα hosts που εκτελούν τις διεργασίες φαίνεται να είναι απ' ευθείας συνδεδεμένα, αλλά στη πραγματικότητα μεσολαβεί η υποδομή των υποκείμενων στρωμάτων.



Υπηρεσίες και πρωτόκολλα μεταφοράς

- Τα πρωτόκολλα μεταφοράς «τρέχουν» στα τερματικά συστήματα.
- Ένα πρωτόκολλο μεταφοράς από:
 - την πλευρά του κόμβου αποστολής, τεμαχίζει τα μηνύματα της εφαρμογής σε *τμήματα* (segments) και αφού τους επισυνάψει μια επικεφαλίδα, τα προωθεί στο επίπεδο δικτύου, όπου ενθυλακώνονται σε ένα *δεδομενόγραμμα* (datagram) για να προωθηθούν στο host-παραλήπτη μέσω του δικτύου (από το επίπεδο δικτύου).
 - την πλευρά λήψης, ανασυνθέτει σε μηνύματα τα τμήματα που λαμβάνει από επίπεδο δικτύου και τα προωθεί στο επίπεδο εφαρμογής.
- Οι δρομολογητές που βρίσκονται στη διαδρομή των datagrams, *ΔΕΝ* ασχολούνται με το περιεχόμενο των segments.



Διαδίκτυο: 2 πρωτόκολλα στο επίπεδο μεταφοράς

- Συνήθως υπάρχουν περισσότερα από ένα πρωτόκολλα μεταφοράς διαθέσιμα στις εφαρμογές.
- Στο διαδίκτυο υπάρχουν 2 πρωτόκολλα μεταφοράς:
 - *TCP* (Transmission Control Protocol)
που παρέχει *συνδεδεσμένη* υπηρεσία στις εφαρμογές με εγγυημένη παράδοση μηνυμάτων, έλεγχο ροής και έλεγχο συμφόρησης (αξιόπιστο).
 - *UDP* (User Datagram Protocol)
που παρέχει *ασυνδεδεσμένη* υπηρεσία, χωρίς αξιοπιστία, χωρίς έλεγχο ροής και χωρίς έλεγχο συμφόρησης.
- Κάθε ένα από αυτά προσφέρει διαφορετικές υπηρεσίες.
- Το ποιο πρωτόκολλο επιλέγεται κάθε φορά, ορίζεται στην εφαρμογή (προγραμματιστής).



Επίπεδο μεταφοράς έναντι επιπέδου δικτύου

- Επίπεδο δικτύου: λογική επικοινωνία *μεταξύ υπολογιστών*
- Επίπεδο μεταφοράς: λογική επικοινωνία *μεταξύ διεργασιών*
- Το επίπεδο μεταφοράς στηρίζεται στο επίπεδο του δικτύου και συμπληρώνει τις υπηρεσίες του.
- Παράδειγμα του βιβλίου: 12 παιδιά μιας οικογένειας στέλνουν γράμματα σε 12 παιδιά μιας άλλης οικογένειας και ένα παιδί σε κάθε οικογένεια αναλαμβάνει το μοίρασμα της αλληλογραφίας.
 - διεργασίες = παιδιά
 - μηνύματα εφαρμογών = γράμματα σε φακέλους
 - υπολογιστές = σπίτια
 - πρωτόκολλο μεταφοράς = το παιδί που αναλαμβάνει το μοίρασμα της αλληλογραφίας σε κάθε σπίτι (λογική επικοινωνία μεταξύ παιδιών)
 - πρωτόκολλο επιπέδου δικτύου = ταχυδρομική υπηρεσία (λογική επικοινωνία μεταξύ οικιών).



Στο επίπεδο δικτύου

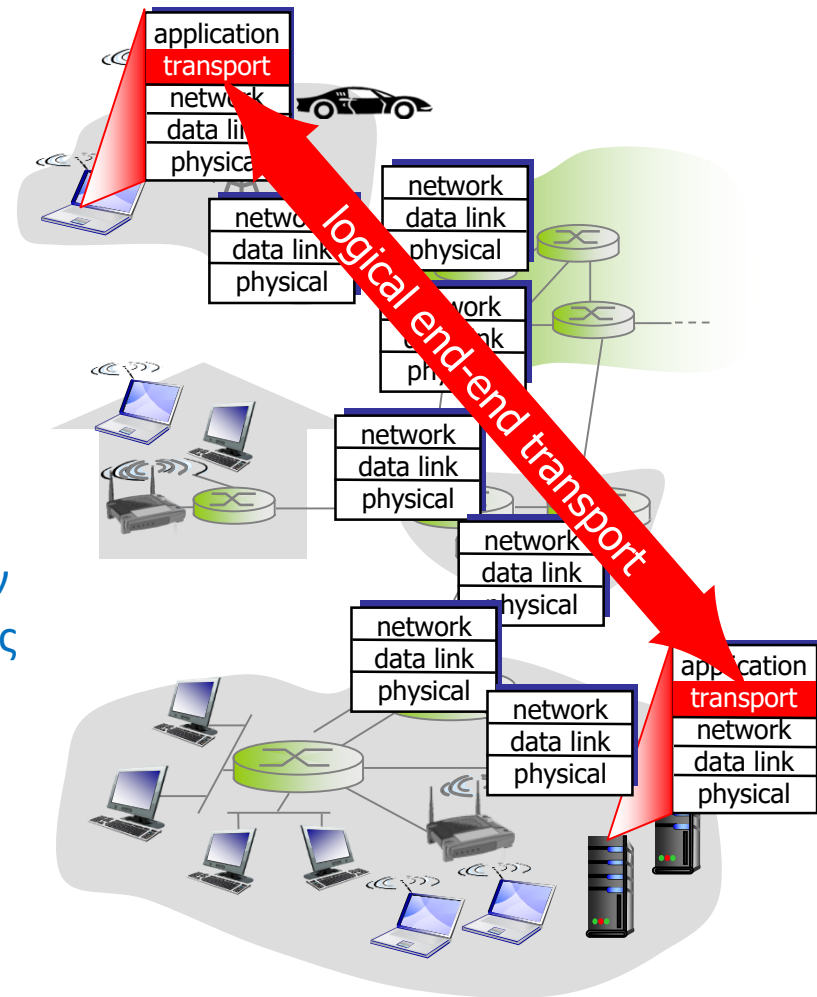
- Επίπεδο δικτύου: ένα πρωτόκολλο, το *IP* (Internet Protocol).
- Κάθε host (υπολογιστής) έχει τουλάχιστον μια διεύθυνση επιπέδου δικτύου (*IP address*).
- Μοντέλο υπηρεσίας επιπέδου δικτύου: *προσπάθεια βέλτιστης παράδοσης* (best effort delivery).
- Πρόκειται για μοντέλο *αναξιόπιστης υπηρεσίας*. Δεν εγγυάται:
 - παράδοση segment (τμήματος),
 - παράδοση των segments με την σωστή σειρά και
 - ακεραιότητα δεδομένων στα segments.
- Τα πρωτόκολλα επιπέδου μεταφοράς (UDP & TCP) κάνουν επέκταση της παράδοσης από host σε host (IP) σε παράδοση από διεργασία σε διεργασία κάθε εφαρμογής.
- Αυτή η επέκταση καλείται *πολύπλεξη-αποπολύπλεξη* (multiplexing-demultiplexing) επιπέδου μεταφοράς.



Πρωτόκολλα επιπέδου μεταφοράς του Διαδικτύου

- TCP: αξιόπιστο, σε ορθή σειρά μεταφορά των τμημάτων με:
 - έλεγχο συμφόρησης (congestion control),
 - έλεγχο ροής (flow control),
 - εγκαθίδρυση σύνδεσης και με
 - έλεγχο σφαλμάτων.
- UDP: μη αξιόπιστο, με παράδοση εκτός σειράς.

Γίνεται απλός έλεγχος σφαλμάτων, όμως δεν υπάρχει καμία άλλη επέκταση της “βέλτιστης προσπάθειας” (best effort) του IP.
- Υπηρεσίες που δεν είναι διαθέσιμες:
 - εγγυήσεις ως προς την καθυστέρηση και
 - εγγυήσεις ως προς το εύρος ζώνης.



Αξιόπιστη μεταφορά (reliable data transfer)

- Τα δεδομένα παραδίδονται, από τη διεργασία αποστολής στη διεργασία λήψης, σωστά και με τη ορθή σειρά.
- Επιτυγχάνεται με:
 - *Έλεγχος ροής* (flow control).

Απαίτηση για επιβεβαίωση λήψης (*acknowledgment*) κάθε τμήματος πακέτου από την άλλη μεριά, πριν σταλθεί το επόμενο τμήμα (δεν υπερφορτώνεται ο παραλήπτης).
 - *Επαναμεταδόσεις*.
 - *Ταυτοποιήσεις* τμημάτων (αριθμοί ακολουθίας).
 - *Χρονομετρήσεις*.
- Ένα πρωτόκολλο επιπέδου μεταφοράς μπορεί να παρέχει αξιόπιστη μεταφορά δεδομένων μεταξύ εφαρμογών, ακόμα και αν το υποκείμενο επίπεδο δικτύου *δεν είναι* αξιόπιστο.
- Αξιόπιστη μεταφορά μπορεί να παρασχεθεί και από πρωτόκολλα άλλων επιπέδων.



Έλεγχος συμφόρησης

- Έλεγχος συμφόρησης (congestion control). Ρύθμιση της κίνησης αποστολής για να μην υπερφορτωθεί το διαδίκτυο.
 - Απαγορεύεται σε οποιαδήποτε σύνδεση να «πλημμυρήσει» με υπερβολική κίνηση τις ζεύξεις και τους δρομολογητές,
 - Χρησιμοποιούνται αλγόριθμοι που έχουν ως σκοπό, είτε να αποφύγουν εξ αρχής τη συμφόρηση, είτε να ανταποκριθούν σε αυτή.
- Πρόκειται για μια υπηρεσία για όλο το διαδίκτυο.
- Χωρίς έλεγχο συμφόρησης ένα δίκτυο μπορεί εύκολα να «φρακάρει» και λίγα ή καθόλου δεδομένα να μη μεταφέρονται.
- Ένα πρωτόκολλο επιπέδου μεταφοράς δεν μπορεί να αγνοήσει τι συμβαίνει στο διαδίκτυο μεταξύ των δύο συνδεδεμένων άκρων (και όμως το UDP το αγνοεί!).
- Πρέπει να είναι αρκετά "έξυπνο", ώστε να αντιληφθεί και να χειριστεί κατάλληλα μια συμφόρηση στο δίκτυο (και περίπλοκο!).



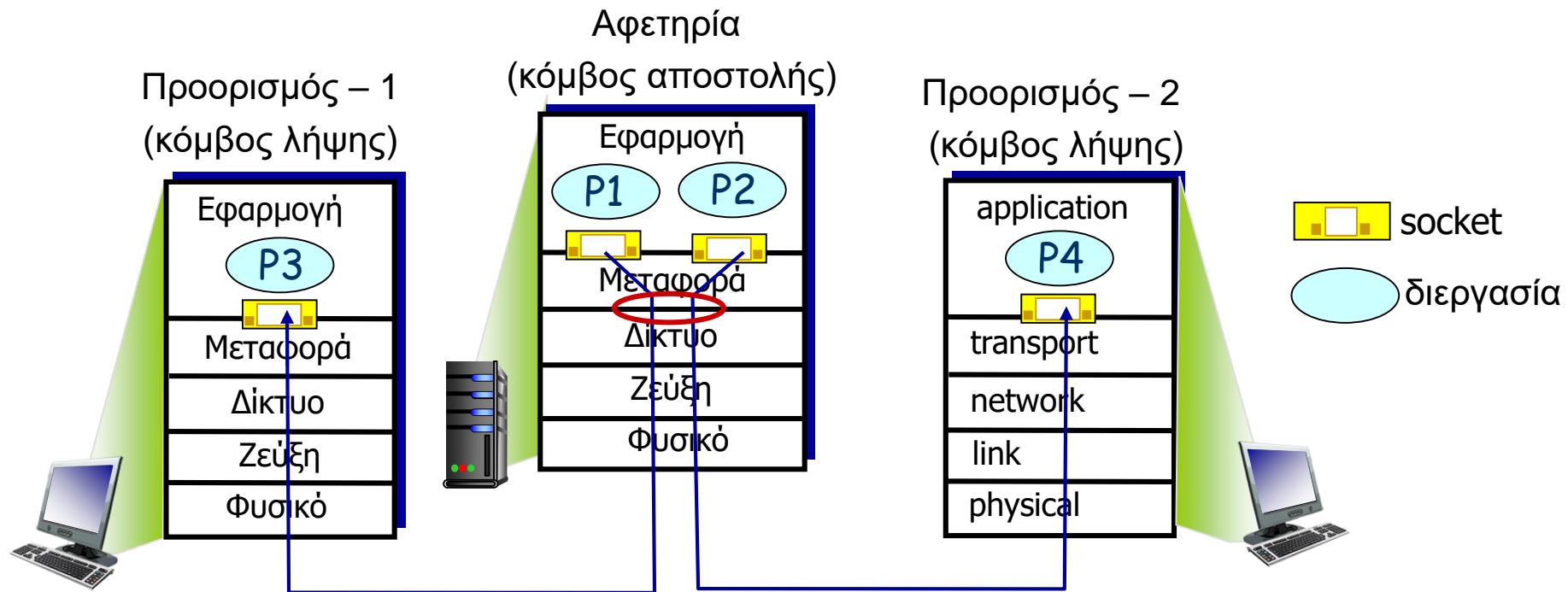
Πολύπλεξη/Αποπολύπλεξη

- Κάθε διεργασία χρησιμοποιεί ένα ή περισσότερα sockets
- Στο επίπεδο μεταφοράς, ένα ζεύγος sockets είναι η μοναδική ταυτοποίηση σύνδεσης μεταξύ 2 εφαρμογών (1 σε κάθε εφαρμογή).
- Στο host αποστολής, το επίπεδο μεταφοράς:
 - λαμβάνει ένα μήνυμα από την εφαρμογή μέσω ενός socket (προορισμού),
 - το χωρίζει σε κομμάτια (πακέτα),
 - ενθυλακώνει κάθε πακέτο του μηνύματος σε ένα τμήμα (segment) και
 - προωθεί κάθε segment στο επίπεδο δικτύου.
- Στο host προορισμού, το επίπεδο μεταφοράς:
 - παραλαμβάνει τα segments από το επίπεδο του δικτύου,
 - σχηματίζει το μήνυμα (ενώνοντας ξανά τα πακέτα του) και
 - διοχετεύει το μήνυμα στη εφαρμογή μέσω ενός socket (αποστολής).
- Προφανώς, σε κάθε host μπορεί να υπάρχουν πολλές εφαρμογές (άρα και πολλά – μοναδικά για κάθε εφαρμογή - sockets).



Πολύπλεξη

- Αφορά στη συλλογή μηνυμάτων που προέρχονται από διαφορετικές διεργασίες εφαρμογών (πολλαπλά sockets) στο host της αποστολής.
- Προσθήκη της κατάλληλης επικεφαλίδας σε κάθε segment που δημιουργείται (και που αργότερα χρησιμοποιείται για αποπολύπλεξη στο host του προορισμού) και παράδοση των segments στο επίπεδο δικτύου.



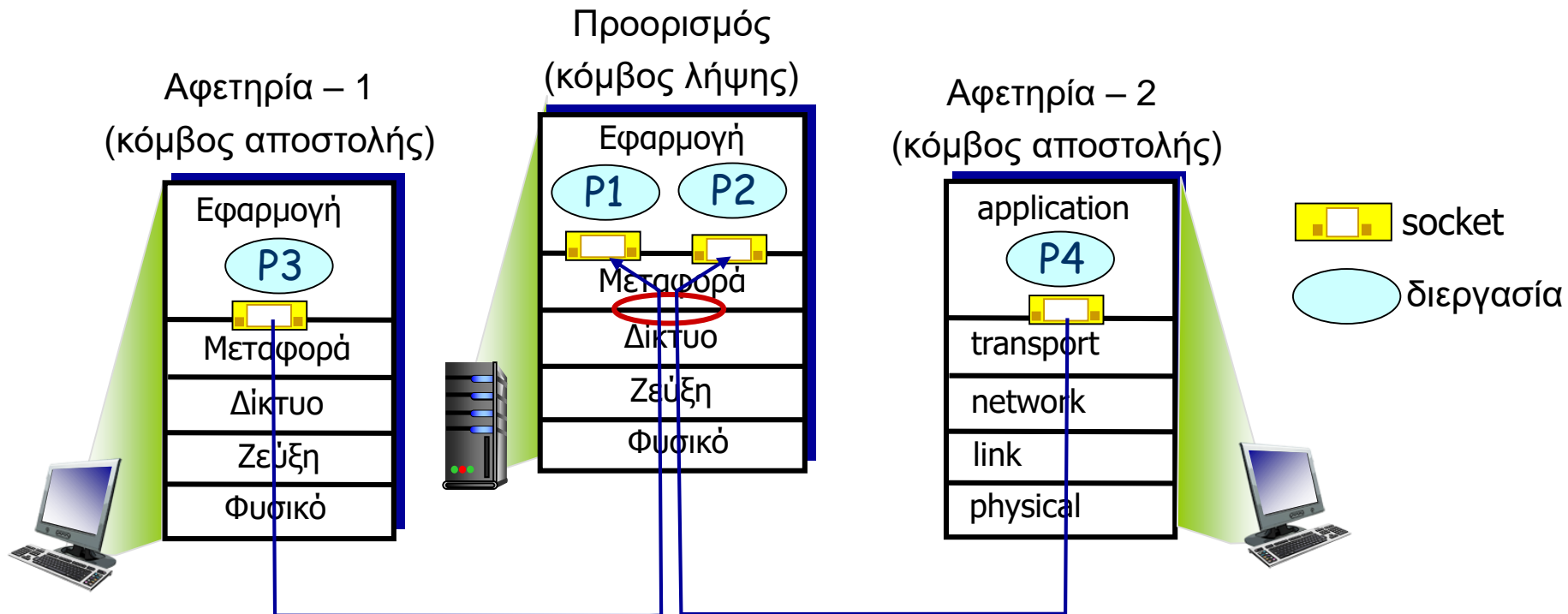
Πώς δουλεύει η πολύπλεξη

- Η πολύπλεξη επιπέδου μεταφοράς χρειάζεται:
 - τα sockets να έχουν μοναδικές ταυτότητες,
 - κάθε segment να έχει ειδικά πεδία που να δηλώνουν τη ταυτότητα του socket (θύρα) στο οποίο πρόκειται να παραδοθεί το segment.
- Στη κεφαλίδα λοιπόν ενός segments του επιπέδου μεταφοράς υπάρχουν 2 πεδία που αφορούν σε:
 - *αριθμό θύρας προέλευσης* (source port field) και σε
 - *αριθμό θύρας προορισμού* (destination port field).
- Κάθε αριθμός θύρας είναι ένας 16-bit αριθμός από 0 έως 65535.
- Οι αριθμοί θύρας από 0 έως 1023 ονομάζονται *πασίγνωστοι αριθμοί θυρών* (well-known port numbers) και είναι δεσμευμένοι από γνωστά πρωτόκολλα.
- Στο επίπεδο μεταφοράς δημιουργείται το segment με τους αριθμούς θυρών προέλευσης και προορισμού και παραδίδεται στο επίπεδο δικτύου για να δημιουργηθεί το δεδομένογραμμα.



Απολύπλεξη

- Αφορά στη συλλογή μηνυμάτων που προέρχονται από το επίπεδο δικτύου στο host λήψης.
- Για κάθε segment που παραλαμβάνεται γίνεται ταυτοποίηση του socket λήψης και κατευθύνεται το segment σε αυτό το socket (παράδοση δεδομένων στο σωστό socket).



Πώς δουλεύει η αποπολύπλεξη

- Ο υπολογιστής προορισμού λαμβάνει IP datagrams (δεδομενογράμματα) από το επίπεδο δικτύου.
 - Κάθε datagram έχει διεύθυνση IP προέλευσης και διεύθυνση IP προορισμού.
 - Κάθε datagram μεταφέρει ένα segment επιπέδου μεταφοράς.
 - Κάθε segment έχει πεδία για αριθμούς θυρών προέλευσης και προορισμού.
- Το επίπεδο δικτύου φροντίζει για την μεταφορά κάθε segment στο σωστό υπολογιστή, αφού είναι γνωστή η IP διεύθυνση του υπολογιστή.
- Το επίπεδο μεταφοράς εξετάζοντας τους αριθμούς θυρών των αφικνούμενων segments, κατευθύνει κάθε segment στο κατάλληλο socket και από εκεί στη αντίστοιχη διεργασία (εφαρμογή), αφού είναι γνωστή η θύρα προορισμού.



Πολύπλεξη/Αποπολύπλεξη (σύνοψη)

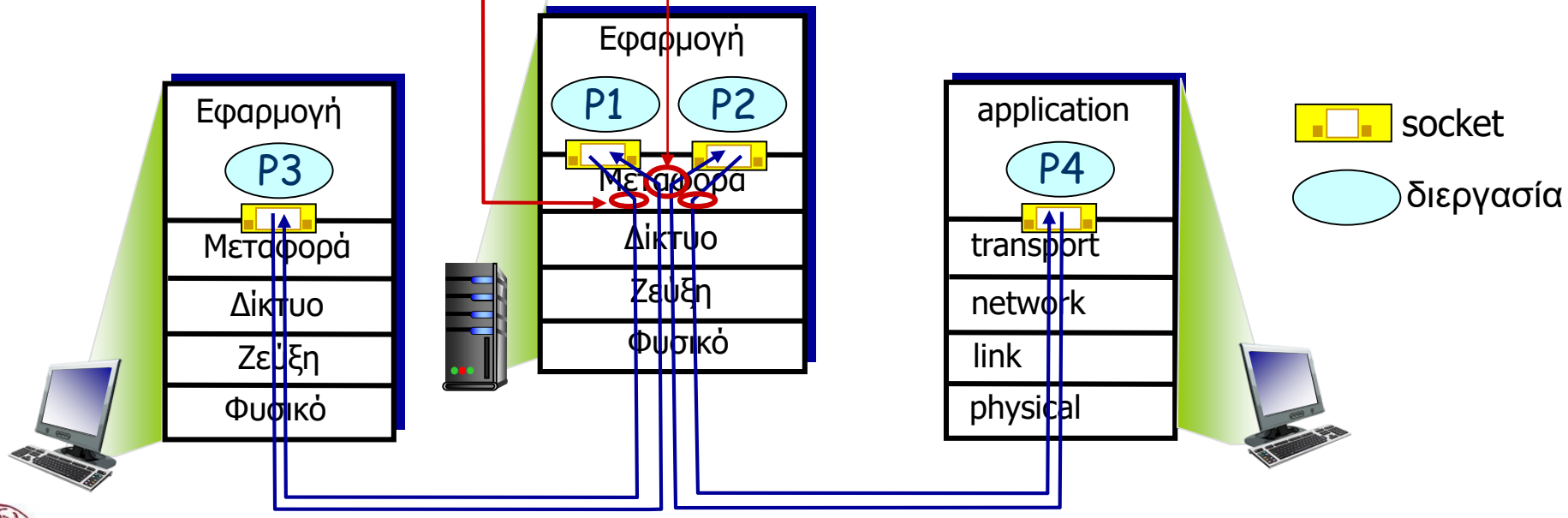
- Κάθε διεργασία χρησιμοποιεί ένα ή περισσότερα sockets
- Μοναδική ταυτοποίηση σύνδεσης: ένα ζεύγος sockets.

Πολύπλεξη:

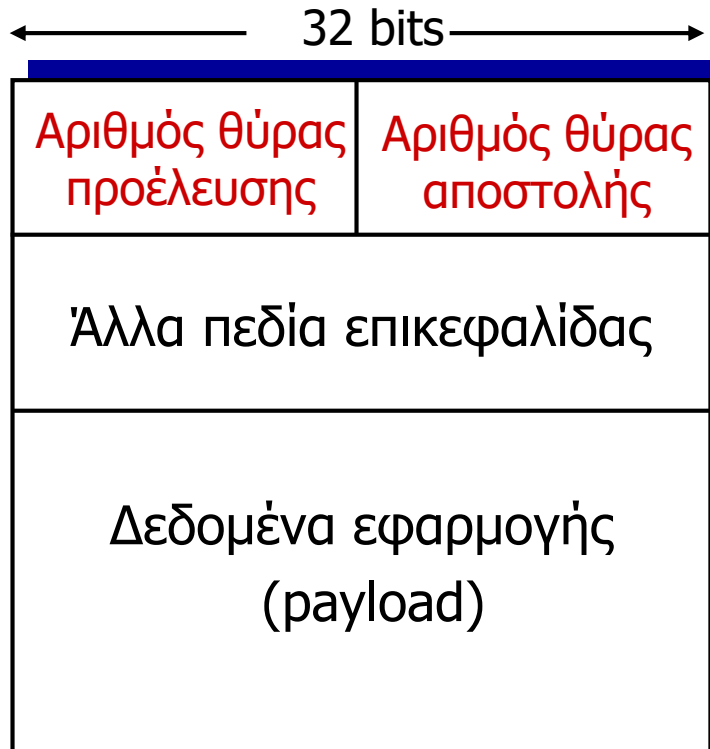
Πολλαπλά sockets, προσθήκη επικεφαλίδας (που αργότερα χρησιμοποιείται για αποπολύπλεξη)

Αποπολύπλεξη:

Παράδοση των τμημάτων που λαμβάνονται, στο σωστό socket (με χρήση της επικεφαλίδας)



Δομή TCP/UDP τμήματος (segment)



Τα πεδία (fields) αριθμών θύρας:

- προορισμού (destination port number) και
- προέλευσης (source port number)

σε ένα segment (τμήμα) επιπέδου μεταφοράς.



Ασυνδεσμική πολύπλεξη (UDP multiplexing)

- Όταν δημιουργείται ένα UDP socket θα πρέπει να εκχωρηθεί σε αυτό ένας αριθμό θύρας.
- Η εκχώρηση από πλευράς πελάτη γίνεται, είτε αυτόματα, είτε από τον προγραμματιστή και δεν πρέπει αυτός ο αριθμός να χρησιμοποιείται από άλλο socket.
- Από πλευράς εξυπηρετητή η εκχώρηση γίνεται από τον προγραμματιστή.
- Η ταυτοποίηση ενός UDP socket επιτυγχάνεται με:
 - την διεύθυνση IP προορισμού και
 - τον αριθμό θύρας προορισμού.
- Όταν στον υπολογιστή αποστολής δημιουργείται ένα datagram στο επίπεδο δικτύου, που ενθυλακώνει ένα segment του επιπέδου μεταφοράς, θα πρέπει να καθορίζονται σε αυτό η IP διεύθυνση και ο αριθμός θύρας του host προορισμού.



Ασυνδεσμική αποπολύπλεξη (UDP demultiplexing)

- Στο υπολογιστή προορισμού από το datagram που παραλαμβάνεται, προκύπτει ένα segment που παραδίδεται στο επίπεδο μεταφοράς.
- Το επίπεδο μεταφοράς στο προορισμό:
 - ελέγχει τον αριθμό θύρας προορισμού στο τμήμα και
 - κατευθύνει το τμήμα UDP στο socket με αυτό τον αριθμό θύρας.
- Αν 2 τμήματα UDP έχουν διαφορετικές IP διευθύνσεις προέλευσης και διαφορετικούς αριθμούς θυρών προέλευσης, αλλά έχουν τις ίδιες IP διευθύνσεις και αριθμούς θυρών προορισμού, τότε παραδίδονται στη ίδια διεργασία μέσω *του ίδιου* socket προορισμού.
- Ο αριθμός θύρας προέλευσης χρησιμοποιείται σαν αριθμός θύρας αποστολής, όταν ο παραλήπτης θέλει να απαντήσει στον αποστολέα.

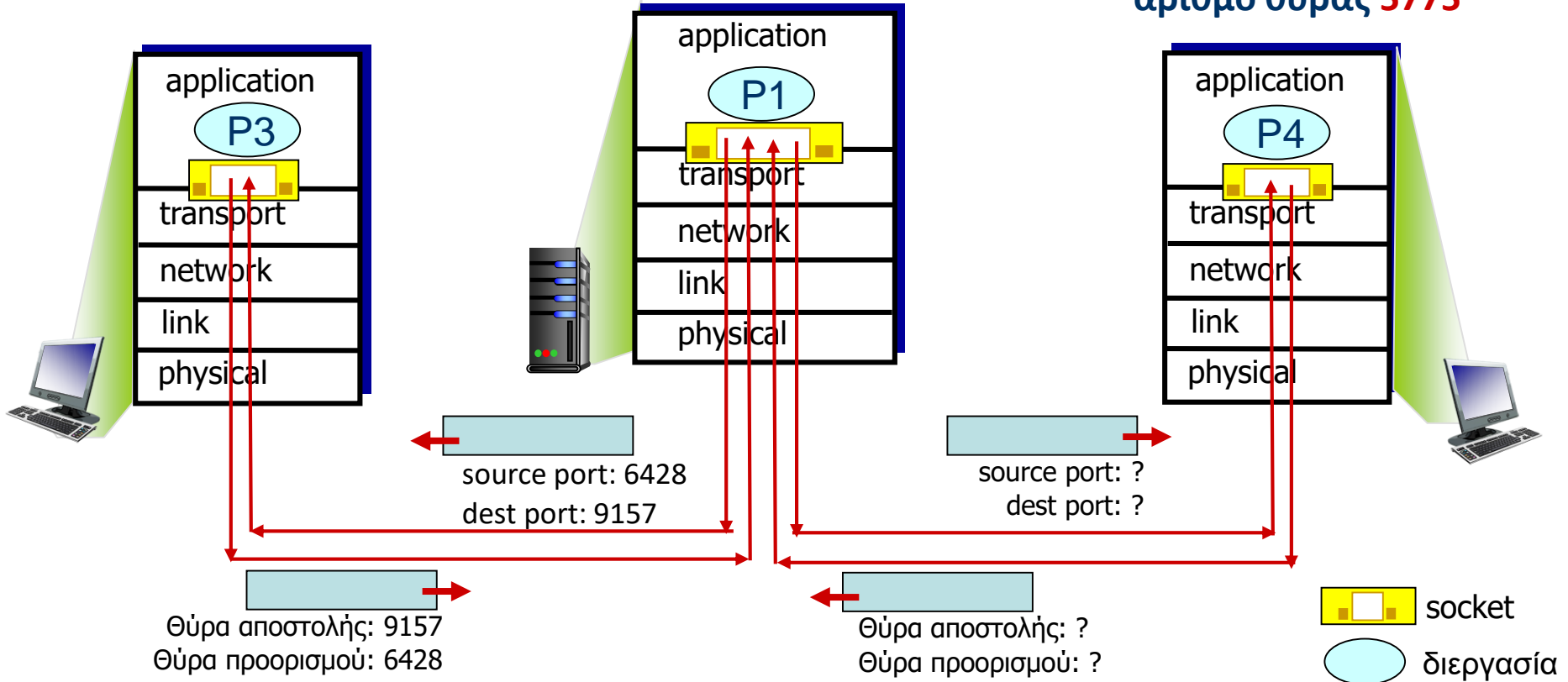


Παράδειγμα ασυνδεσμικής αποπολύπλεξης

Δημιουργία socket με αριθμό θύρας **9157**

Δημιουργία socket με αριθμό θύρας **6428**

Δημιουργία socket με αριθμό θύρας **5775**



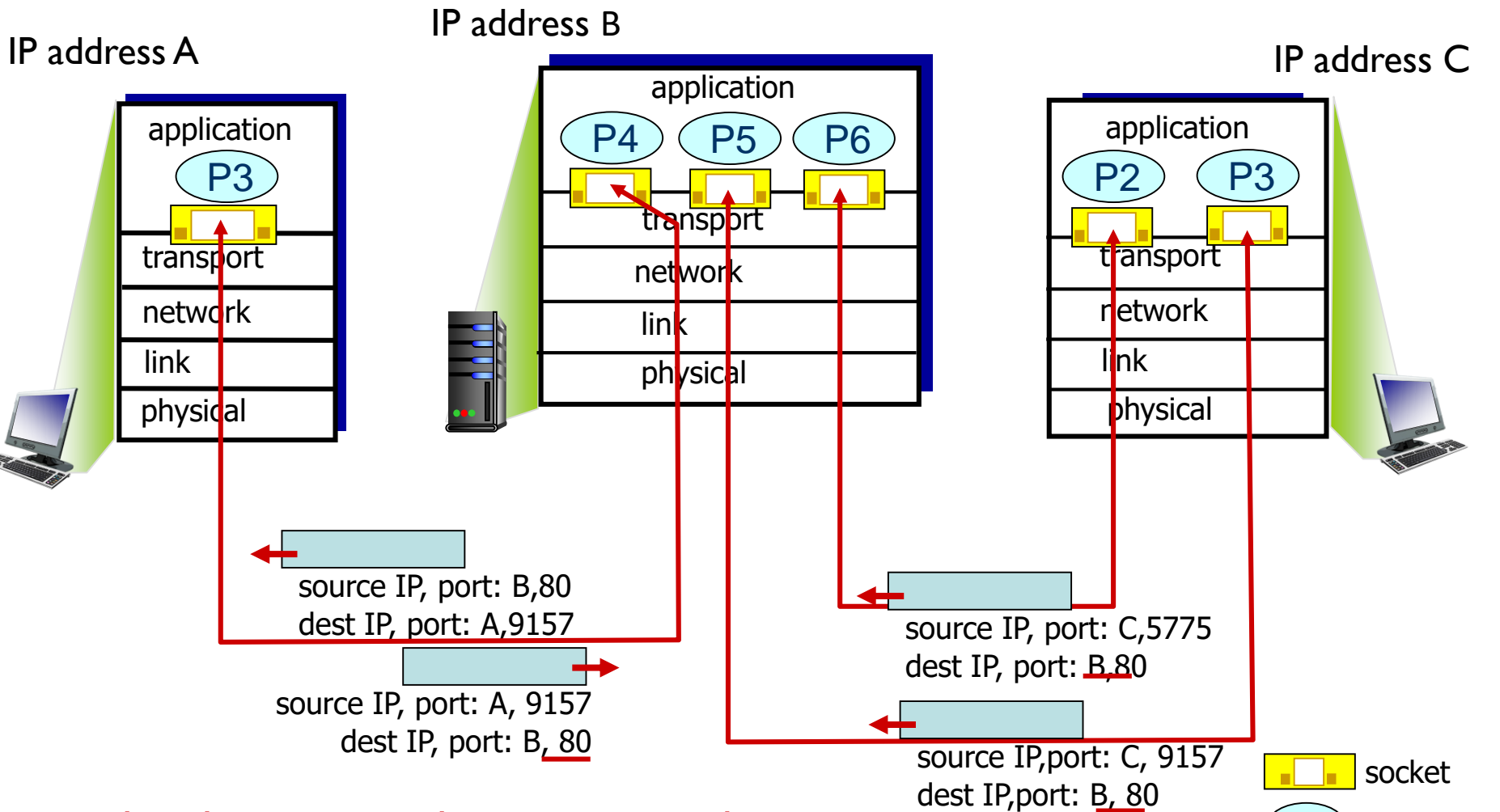
Αντιστροφή των αριθμών θυρών προέλευσης και προορισμού

Συνδεδεσμένη αποπολύπλεξη (TCP)

- Η ταυτοποίησης ενός TCP socket επιτυγχάνεται με την τετράδα:
 - διεύθυνση IP προέλευσης,
 - αριθμός θύρας προέλευσης,
 - διεύθυνση IP προορισμού και
 - αριθμός θύρας προορισμού.
- Ο υπολογιστής λήψης χρησιμοποιεί και τις τέσσερις τιμές για να κατευθύνει το segment στο κατάλληλο socket υποδοχής.
- Δυο διαφορετικά segments αποστολής με διαφορετικές IP διευθύνσεις και διαφορετικούς αριθμούς θυρών κατευθύνονται σε **διαφορετικά sockets**, ακόμα και αν οι IP διευθύνσεις και οι αριθμοί θυρών αποστολής είναι τα ίδια.
- Ένας υπολογιστής εξυπηρετητής μπορεί να υποστηρίξει πολλά ταυτόχρονα TCP sockets:
 - κάθε socket αναγνωρίζεται από τη δική του τετράδα.



Παράδειγμα συνδεσμικής αποπολύπλεξης



Τρία τμήματα με IP διεύθυνση προορισμού: B,
Στη θύρα προορισμού: 80 αποπολυπλέκονται σε διαφορετικά sockets



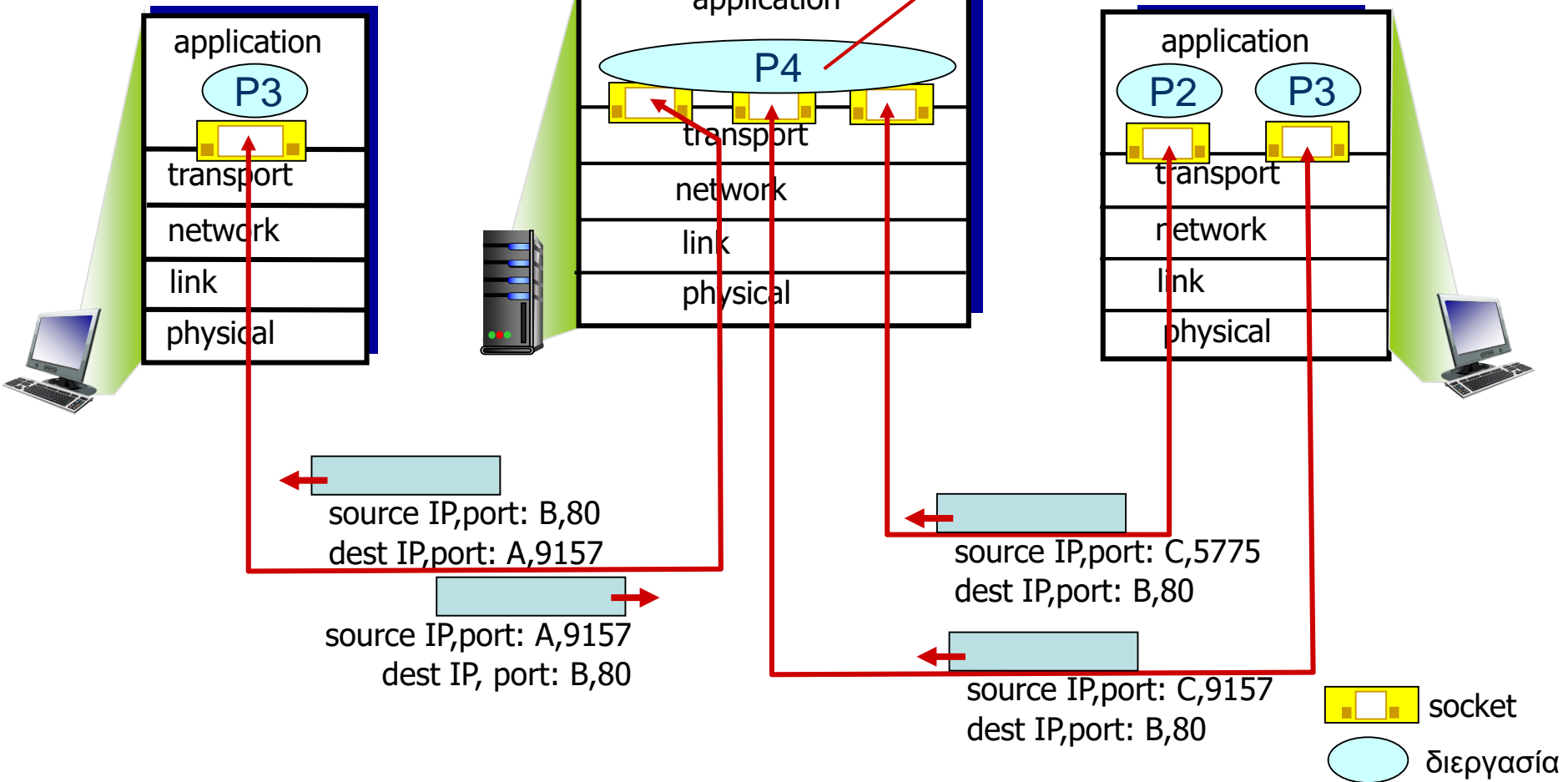
Παράδειγμα συνδεσμικής αποπολύπλεξης

Εξυπηρετητής με νήματα (threaded)

IP address A

IP address B

IP address C



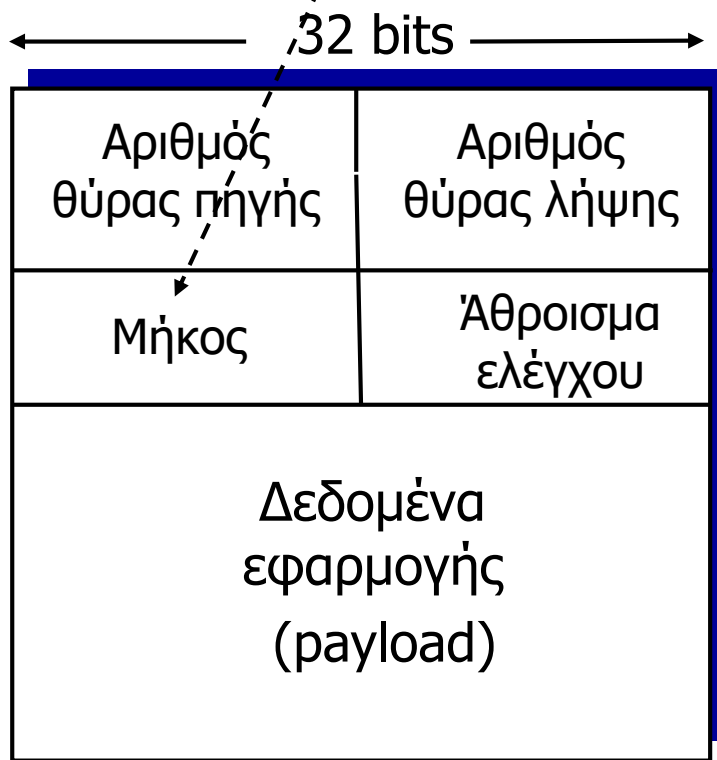
UDP (User Datagram Protocol) [RFC 768]

- Το απλούστερο πρωτόκολλο μεταφοράς του διαδικτύου
 - κάνει τα ελάχιστα που μπορεί να κάνει ένα πρωτόκολλο μεταφοράς.
- Υπηρεσία παράδοσης βέλτιστης προσπάθειας (“best effort”)
- Τα τμήματα UDP μπορεί να:
 - χαθούν ή/και να
 - παραδοθούν εκτός σειράς στις εφαρμογές.
- Ασυνδεδεστρεφής (connection-less) υπηρεσία :
 - δεν γίνεται χειραψία μεταξύ των UDP αποστολέα και δέκτη,
 - η διαχείριση κάθε UDP segment γίνεται ανεξάρτητα από τα άλλα.
- Συχνά χρησιμοποιείται για εφαρμογές πολυμέσων συνεχούς ροής (streaming) που έχουν:
 - ανοχές ως προς τις απώλειες και
 - ευαισθησία ως προς το ρυθμό.
- Άλλες γνωστές υπηρεσίες που χρησιμοποιούν UDP: DNS, SMTP



UDP: δομή της κεφαλίδας

Μήκος του τμήματος UDP σε Bytes, συμπεριλαμβανομένης της κεφαλίδας



Δομή τμήματος UDP

Γιατί υπάρχει το UDP;

- Δεν έχει εγκαθίδρυση σύνδεσης (που εισάγει καθυστέρηση).
- Είναι απλό: χωρίς κατάσταση σύνδεσης στον αποστολέα, δέκτη.
- Με μικρή κεφαλίδα τμήματος.
- Χωρίς έλεγχο συμφόρησης: το UDP μπορεί να “εκραγεί” όσο γρήγορα θέλουμε.

Αξιόπιστη μεταφορά πάνω από το UDP:

- προσθήκη αξιοπιστίας στο επίπεδο εφαρμογής
- ανάνηψη από τα λάθη στην εφαρμογή.



UDP: άθροισμα ελέγχου (checksum)

- Σκοπός: ανίχνευση «σφαλμάτων» (π.χ. ανεστραμμένων bits) στο μεταδιδόμενο τμήμα.
- Αποστολέας:
 - χειρίζεται το περιεχόμενο του τμήματος ως ακολουθία 16 bits ακεραίων.
 - *checksum*: συμπλήρωμα ως προς το 1 του αθροίσματος του περιεχομένου (εκφρασμένου σε 16-bit λέξεις)
 - ο αποστολέας τοποθετεί την τιμή του checksum στο πεδίο checksum του τμήματος UDP.
- Δέκτης:
 - υπολογισμός του checksum του λαμβανομένου μηνύματος,
 - έλεγχος αν το υπολογισμένο checksum ισούται με την τιμή του πεδίου checksum:
 - *OXI* – ανίχνευση σφάλματος
 - *NAI* – μη ανίχνευση σφάλματος



Παράδειγμα αθροίσματος ελέγχου

	1	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0
	1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
υπερχείλιση	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1
άθροισμα	1	0	1	1	1	0	1	1	1	0	1	1	1	1	0	0
checksum	0	1	0	0	0	1	0	0	0	1	0	0	0	0	1	1

Κατά την πρόσθεση των αριθμών, το κρατούμενο από την πιο σημαντική θέση πρέπει να προστεθεί στο αποτέλεσμα.

- Ελέγχους για σφάλματα γίνονται και σε άλλα επίπεδα.
- Ο έλεγχος σφαλμάτων στο επίπεδο μεταφοράς ακολουθεί την αρχή από-άκρο-σε-άκρο.

Λειτουργίες κατώτερων επιπέδων μπορεί να είναι πλεονάζουσες ή μικρής αξίας σε σύγκριση με το κόστος παροχής τους σε ψηλότερο επίπεδο.



TCP (Transmission Control Protocol)

- Αξιόπιστη και σε σειρά ροή από bytes:
 - Τα δεδομένα μεταφέρονται σαν μια συνεχή ακολουθία από bytes, χωρίς “όρια μηνυμάτων”.
- Με διοχέτευση (pipeline):
 - Ο έλεγχος συμφόρησης και ροής του TCP καθορίζουν το μέγεθος του παραθύρου (αριθμός μεταδοθέντων, αλλά μη επιβεβαιωμένων πακέτων).
- Πλήρως αμφίδρομη μεταφορά δεδομένων:
 - Δι-κατευθυντική ροή δεδομένων στην ίδια σύνδεση.
 - *MSS*: maximum segment size (μέγιστο μέγεθος τμήματος).
- Συνδεσμική υπηρεσία:
 - Η χειραψία (handshaking) που αφορά σε ανταλλαγή ειδικών αρχικών μηνυμάτων ελέγχου, προετοιμάζει την κατάσταση του αποστολέα και του δέκτη πριν την ανταλλαγή δεδομένων (εγκαθίδρυση σύνδεσης).



TCP σύνδεση: από σημείο σε σημείο

- Μια σύνδεση TCP είναι *πάντα από-σημείο-σε-σημείο (άκρο-σε-άκρο)*.
 - Ανάμεσα σε μια μόνο εφαρμογή-αποστολέα και σε μια μόνο εφαρμογή-παραλήπτη (ανάμεσα σε 2 υπολογιστές).
- Μια σύνδεση TCP δεν είναι μια σύνδεση σε όλο το μήκος της διαδρομής που αποστέλλονται τα τμήματα (segments).
- Η εφαρμογή του πρωτοκόλλου TCP γίνεται μόνο στα τερματικά συστήματα και όχι στους ενδιάμεσους κόμβους.
- Η κατάσταση σύνδεσης βρίσκεται εξ' ολοκλήρου στα δύο άκρα και ΔΕΝ διατηρείται κάποιο κύκλωμα.
- Οι ενδιάμεσοι κόμβοι της διαδρομής δεν είναι ενήμεροι για τις συνδέσεις TCP και δεν οφείλουν να τις διατηρούν (δεν δεσμεύουν πόρους).
- Γενικότερα, οι ενδιάμεσοι κόμβοι (routers, switches) ενδιαφέρονται μέχρι τα 3 χαμηλότερα επίπεδα.



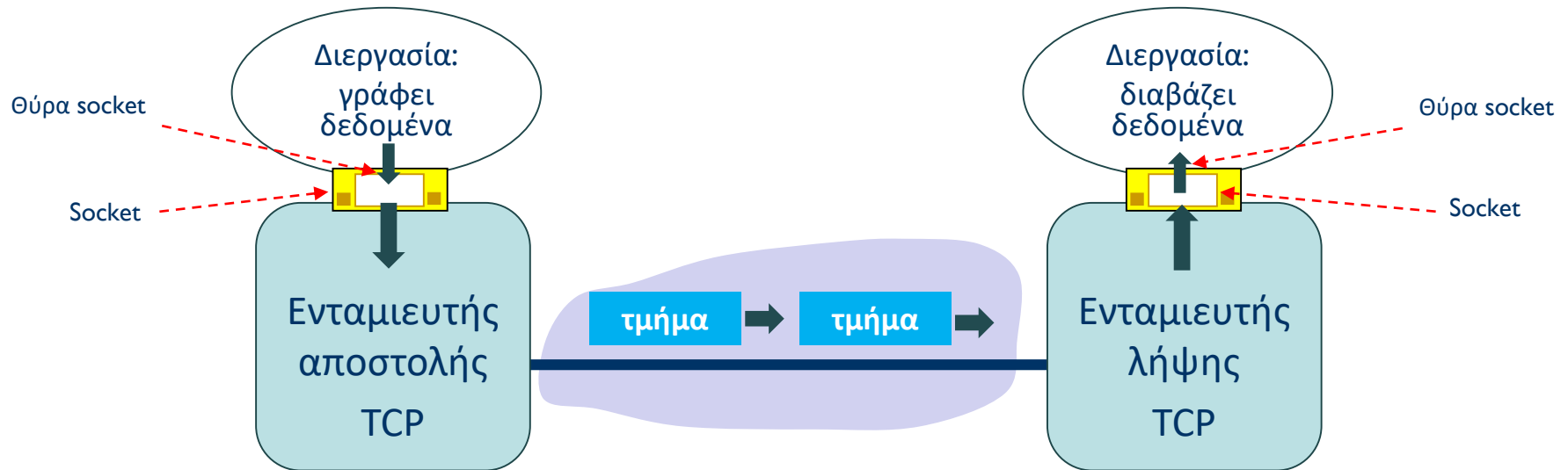
Η εγκαθίδρυση σύνδεσης στο TCP

- Η διεργασία πελάτης που θέλει να ξεκινήσει μια σύνδεση ειδοποιεί το TCP (επίπεδο μεταφοράς):
 - Το TCP από τη πλευρά του πελάτη, δημιουργεί ένα socket στέλνει ένα ειδικό τμήμα TCP στον εξυπηρετητή.
 - Ο εξυπηρετητής αποκρίνεται με ένα δεύτερο ειδικό τμήμα TCP. Κάθε εξυπηρετητής TCP έχει ένα socket υποδοχής και δημιουργεί ένα νέο socket υποδοχής για την συγκεκριμένη διεργασία πελάτη.
 - Ο πελάτης αποκρίνεται πάλι με ένα τρίτο ειδικό τμήμα TCP.
- Τα 2 πρώτα τμήματα δεν περιέχουν *ωφέλιμο φορτίο* (payload: δεδομένα επιπέδου εφαρμογής). Το 3^ο μπορεί να περιέχει ωφέλιμο φορτίο.
- Η διαδικασία εγκαθίδρυσης σύνδεσης ονομάζεται *τρίδρομη χειραψία* (three-way handshaking).
- Μετά τη εγκαθίδρυση σύνδεσης οι 2 διεργασίες μπορούν να ανταλλάσσουν δεδομένα.



Ενταμιευτές αποστολής-λήψης

- Τα δεδομένα που στέλνει ή παραλαμβάνει μια διεργασία αποθηκεύονται προσωρινά σαν τμήματα σε ενταμιευτές που χειρίζεται το TCP.
- Οι ενταμιευτές (πόροι) δεσμεύονται κατά την διάρκεια της χειραψίας.

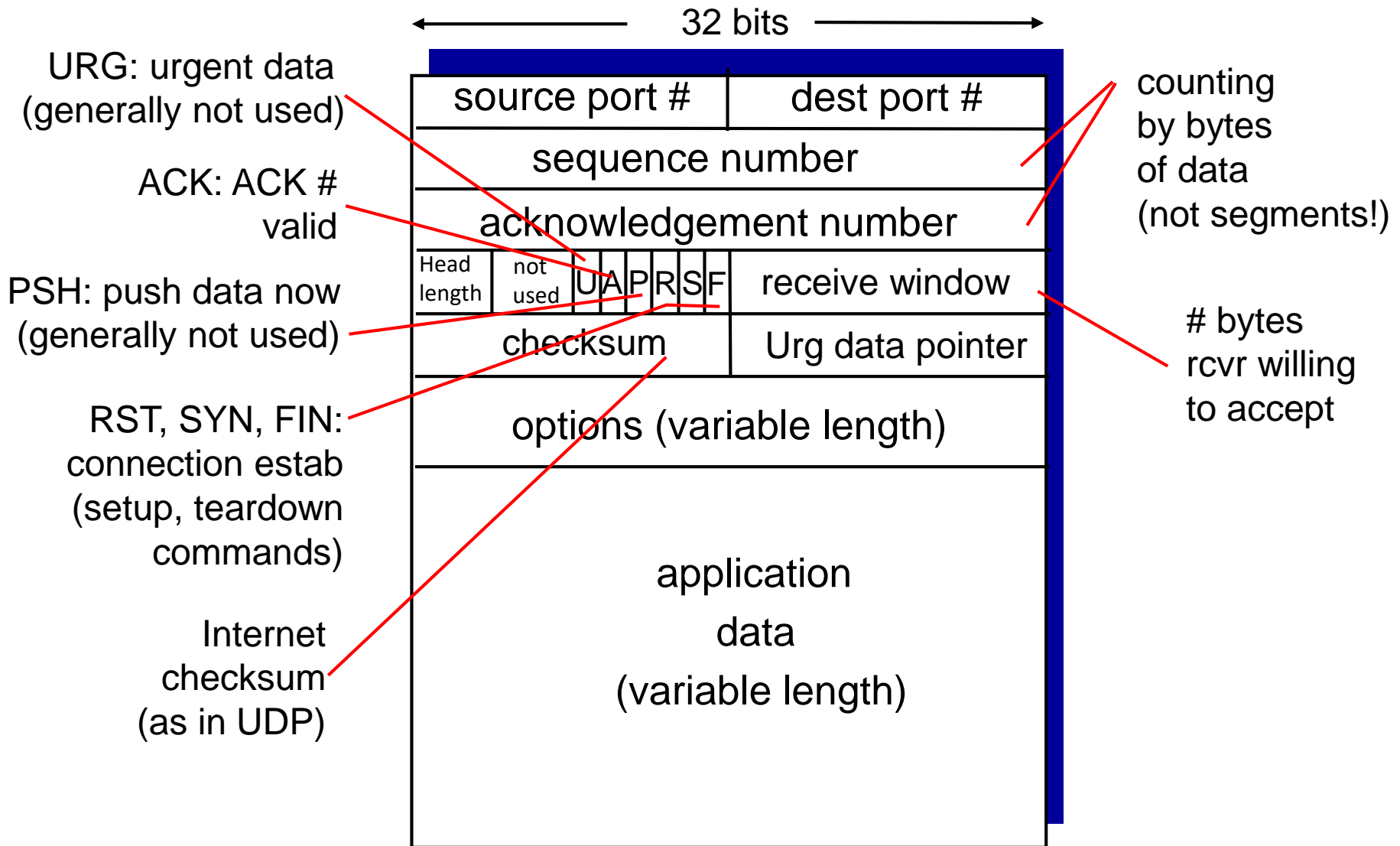


Μέγιστο μέγεθος τμήματος

- Το TCP τεμαχίζει τα δεδομένα του πελάτη (μήνυμα), βάζει μια κεφαλίδα και σχηματίζει ένα τμήμα TCP (segment TCP).
- Η μέγιστη ποσότητα δεδομένων πελάτη που μπορούν ν' απαρτίσουν ένα τμήμα ονομάζεται *μέγιστο μέγεθος τμήματος* (Maximum Segment Size - MSS).
- Το *MSS* τυπικά καθορίζεται από το μήκος του μεγαλύτερου πλαισίου ζεύξης (frame) που μπορεί ν' αποσταλεί από τον τοπικό host (πελάτη).
- Φυσικά, λαμβάνονται υπόψη και οι κεφαλίδες που χρειάζονται για να σχηματιστούν τα datagrams στο επίπεδο δικτύου.



Δομή τμήματος TCP



client state

LISTEN

SYNSENT

ESTAB

choose init seq num, x
send TCP SYN msg

received SYNACK(x)
indicates server is live;
send ACK for SYNACK;
this segment may contain
client-to-server data



server state

LISTEN

SYN RCVD

ESTAB

SYNbit=1, Seq=x

SYNbit=1, Seq=y
ACKbit=1; ACKnum=x+1

ACKbit=1, ACKnum=y+1

choose init seq num, y
send TCP SYNACK
msg, acking SYN

received ACK(y)
indicates client is live





© OmniSecu.com

