ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Λειτουργικά Συστήματα

Ενότητα # 7: Ασφάλεια

Διδάσκων: Γεώργιος Ξυλωμένος

Τμήμα: Πληροφορικής

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «**Ανοικτά Ακαδημαϊκά Μαθήματα στο Οικονομικό Πανεπιστήμιο Αθηνών**» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

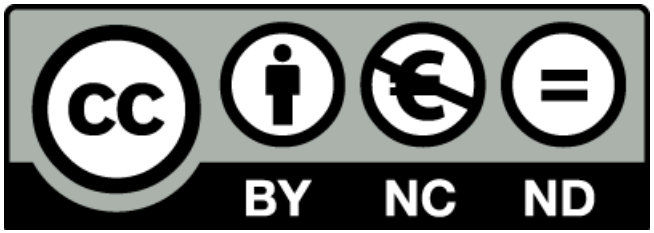
ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Οι εικόνες προέρχονται από το βιβλίο «Σύγχρονα Λειτουργικά Συστήματα», A.S. Tanenbaum, 3η έκδοση, 2009, Εκδόσεις Κλειδάριθμος.



Σκοποί ενότητας

- Κατανόηση της έννοιας της ασφάλειας στα ΛΣ και εισαγωγή στην κρυπτογραφία
- Εξοικείωση με τους βασικούς μηχανισμούς προστασίας και πιστοποίησης ταυτότητας
- Κατανόηση των βασικών κινδύνων ασφάλειας, όπως εσωτερικές επιθέσεις, αξιοποίηση σφαλμάτων κώδικα και κακόβουλο λογισμικό
- Εισαγωγή στους βασικούς τρόπους άμυνας απέναντι στις επιθέσεις στα ΛΣ

Περιεχόμενα ενότητας

- Περιβάλλον ασφάλειας
- Αρχές κρυπτογραφίας
- Μηχανισμοί προστασίας
- Πιστοποίηση ταυτότητας
- Εσωτερικές επιθέσεις
- Αξιοποίηση σφαλμάτων κώδικα
- Κακόβουλο λογισμικό
- Τρόποι άμυνας

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Περιβάλλον ασφάλειας

Μάθημα: Λειτουργικά Συστήματα, **Ενότητα # 7:** Ασφάλεια

Διδάσκων: Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Εισαγωγή

- Το πρόβλημα της ασφάλειας
  - Αποθήκευση εμπιστευτικών πληροφοριών
  - Τα πολύπλοκα ΛΣ δυσχεραίνουν την προστασία
  - Ακόμη μεγαλύτερο πρόβλημα είναι η δικτύωση
- Το πρόβλημα σταδιακά μεταβάλλεται
  - Παλιότερα: ανάμεσα στους χρήστες μιας μηχανής
  - Ενδιάμεσα: κάθε χρήστης στη δική του μηχανή
  - Τώρα: κάθε χρήστης βλέπει μηχανές των άλλων

Το περιβάλλον ασφάλειας

- Η ασφάλεια έχει πολλές όψεις
  - Λειτουργικό σύστημα
  - Δίκτυο υπολογιστών
  - Πληροφοριακό σύστημα
- Ορολογία
  - Ασφάλεια (security)
    - Τα δεδομένα δεν θα διαβαστούν ή τροποποιηθούν
  - Μηχανισμοί προστασίας (protection mechanisms)
    - Μέθοδοι επίτευξης των στόχων της ασφάλειας

Απειλές (1 από 2)

- Ένα σύστημα έχει κάποιους στόχους ασφάλειας
- Κάθε στόχος αντιμετωπίζει κάποιες απειλές
- Εμπιστευτικότητα δεδομένων (confidentiality)
  - Απόκρυψη μυστικών δεδομένων
  - Ο ιδιοκτήτης τους καθορίζει ποιος θα τα δει
- Ακεραιότητα δεδομένων (integrity)
  - Τροποποίηση μόνο με άδεια του ιδιοκτήτη τους
  - Αλλαγή, προσθήκη και διαγραφή δεδομένων

Απειλές (2 από 2)

- Διαθεσιμότητα συστήματος (availability)
  - Το σύστημα δεν αχρηστεύεται από τρίτους
    - Επιθέσεις άρνησης εξυπηρέτησης (denial of service)
    - Συνήθως εναντίον ομάδων και όχι χρηστών
- Αποκλεισμός εισβολέων (intruder exclusion)
  - Αποφυγή κατάληψης του συστήματος
    - Προσπάθειες μετατροπής του σε ζόμπι
    - Παράδειγμα: συμμετοχή σε spam botnet

Δύσκολα θέματα

- Προσωπικό απόρρητο (privacy)
  - Τι πρόσβαση έχει ο Χ στα δεδομένα μου;
  - Όπου Χ = κράτος/εργοδότης/οικογένεια
- Εγείρονται μη τεχνικά ζητήματα
  - Νομικά
  - Πολιτικά
  - Ηθικά

Εισβολείς (1 από 2)

- Εισβολείς (intruders) ή εχθροί (adversaries)
  - Παθητικοί (passive): διαβάζουν δεδομένα
  - Ενεργητικοί (active): τροποποιούν δεδομένα
- Απλοί χρήστες χωρίς τεχνικές γνώσεις
  - Διαβάζουν απροστάτευτα δεδομένα
  - Κίνητρό τους η περιέργεια
- Εσωτερικοί χρήστες με τεχνικές γνώσεις
  - Διαχειριστές, φοιτητές, προγραμματιστές
  - Αντιμετωπίζουν την προστασία ως πρόκληση

Εισβολείς (2 από 2)

- Επαγγελματίες με στόχο το κέρδος
  - Εκμεταλλεύονται κενά του συστήματος
- Εμπορικοί ή στρατιωτικοί κατάσκοποι
  - Χρηματοδότηση και προχωρημένα μέσα
- Ιοί και παρόμοιο λογισμικό
  - Προσπαθούν να κάνουν ζημιές
  - Δεν έχουν συγκεκριμένο στόχο
  - Μπορεί να έρχονται από οπουδήποτε

Απώλεια δεδομένων

- Απώλεια δεδομένων από ατύχημα
  - Δεν υπάρχουν μόνο κακόβουλοι εισβολείς
  - Θεομηνίες: πλημμύρες, σεισμοί, φωτιές
  - Σφάλματα υλικού/λογισμικού: αλλοίωση
  - Ανθρώπινα λάθη: διαγραφή αρχείων κατά λάθος
  - Αντιμετωπίζονται με αντίγραφα ασφαλείας
    - Κατά προτίμηση μακριά από το σύστημα
  - Πιο σοβαρό πρόβλημα από τις παραβιάσεις!

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Αρχές κρυπτογραφίας

Μάθημα: Λειτουργικά Συστήματα, **Ενότητα # 7:** Ασφάλεια

Διδάσκων: Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής

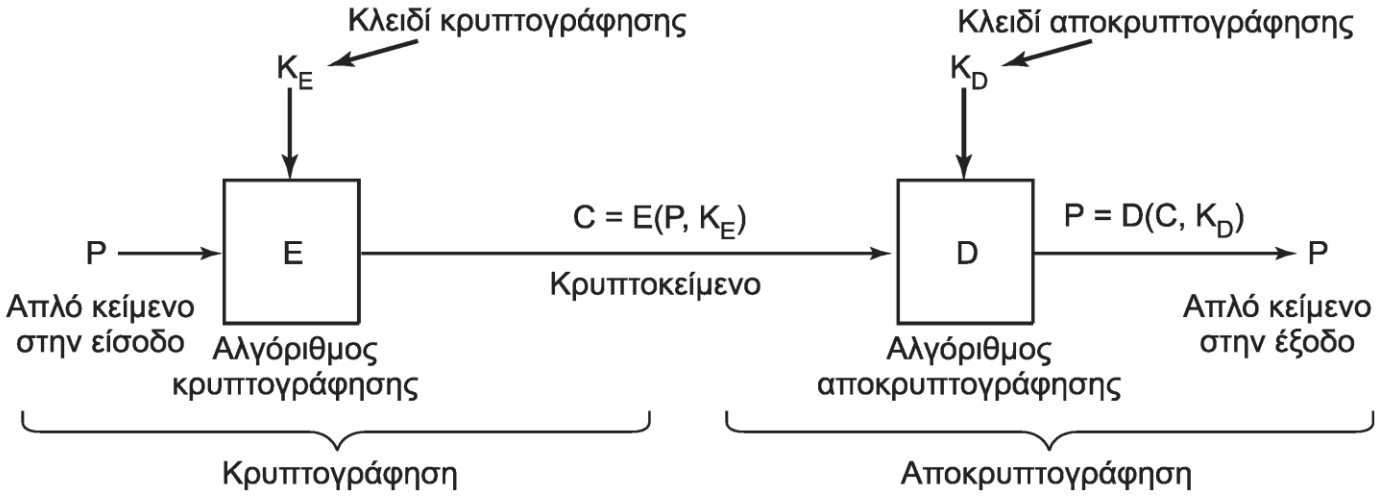
Βασικές αρχές (1 από 3)

- Κρυπτογραφία: απόκρυψη δεδομένων
  - Ξεκινάμε με το απλό κείμενο (P, plaintext)
  - Εφαρμόζουμε έναν μετασχηματισμό
  - Παράγουμε το κρυπτοκείμενο (C, ciphertext)
  - Για τους τρίτους είναι ακατανόητο
- Αλγόριθμοι κρυπτογράφησης: P->C
- Αλγόριθμοι αποκρυπτογράφησης: C->P

Βασικές αρχές (2 από 3)

- Αρχή του Kerchoff (Kerchoff's principle)
  - Οι αλγόριθμοι κρυπτογραφίας είναι δημόσιοι
  - Τα κλειδιά (keys) τους είναι μυστικά
  - Τελικά οι αλγόριθμοι θα μαθευτούν
  - Τα κλειδιά όμως αλλάζουν τακτικά
- Εναλλακτικά: ασφάλεια μέσω κάλυψης
  - Κρατάμε μυστικά τα πάντα
  - Ρεαλιστικά, κάποτε θα διαρρεύσουν

Βασικές αρχές (3 από 3)



- Δομή συστημάτων κρυπτογραφίας
  - Έστω P το απλό κείμενο και C το κρυπτοκείμενο
  - Έστω E/D οι αλγόριθμοι (απο)κρυπτογράφησης
  - Έστω K_E/K_D τα κλειδιά (απο)κρυπτογράφησης
  - $C=E(P,K_E)$ και $P=D(C,K_D)$

Μυστικού κλειδιού (1 από 2)

- Κρυπτογραφία μυστικού κλειδιού
- Μονοαλφαβητική υποκατάσταση
  - Κάθε γράμμα αντικαθίσταται (1 προς 1)
  - Απλό κείμενο:
    `ABCDEFGHIJKLMNOPQRSTUVWXYZ`
  - Κρυπτοκείμενο:
    `QWERTYUIOPASDFGHJKLZXCVBNM`
  - Το κλειδί είναι η συμβολοσειρά αντικατάστασης
  - Παρόμοια κρυπτογράφηση / αποκρυπτογράφηση

Μυστικού κλειδιού (2 από 2)

- Μονοαλφαβητική υποκατάσταση
  - Απλές επιθέσεις για αποκρυπτογράφηση
    - Αξιοποίηση στατιστικών χαρακτήρων
    - Προσθήκη στατιστικών διγραμμάτων
- Κρυπτογραφία συμμετρικού κλειδιού
  - Τα δύο κλειδιά είναι τα ίδια
    - Ή το ένα παράγεται εύκολα από το άλλο
  - Σύνηθες στην κρυπτογραφία μυστικού κλειδιού

Δημόσιου κλειδιού (1 από 2)

- Κρυπτογραφία δημόσιου κλειδιού
- Τα μυστικά κλειδιά έχουν πρόβλημα διανομής
  - Το κλειδί δεν πρέπει ναι διαρρεύσει
- Αντιμετώπιση με ζεύγη κλειδιών
  - Το ιδιωτικό διατηρείται μυστικό στον παραλήπτη
  - Το δημόσιο διανέμεται ανοιχτά σε όλους
  - Το δημόσιο χρησιμοποιείται για κρυπτογράφηση
  - Το ιδιωτικό χρησιμοποιείται για αποκρυπτογράφηση

Δημόσιου κλειδιού (2 από 2)

- Κρυπτογραφία δημόσιου κλειδιού
  - Τα κλειδιά δημιουργούνται μαζί
  - Η συσχέτισή τους είναι εξαιρετικά δύσκολη
    - Δεν βγαίνει εύκολα το ιδιωτικό από το δημόσιο
  - Βασίζεται σε δύσκολα αντιστρέψιμες πράξεις
    - 314159265358979 στο τετράγωνο
    - Ρίζα 3912571506419387090594828508241
    - Αντίστροφες αλλά όχι αντιστρέψιμες πράξεις

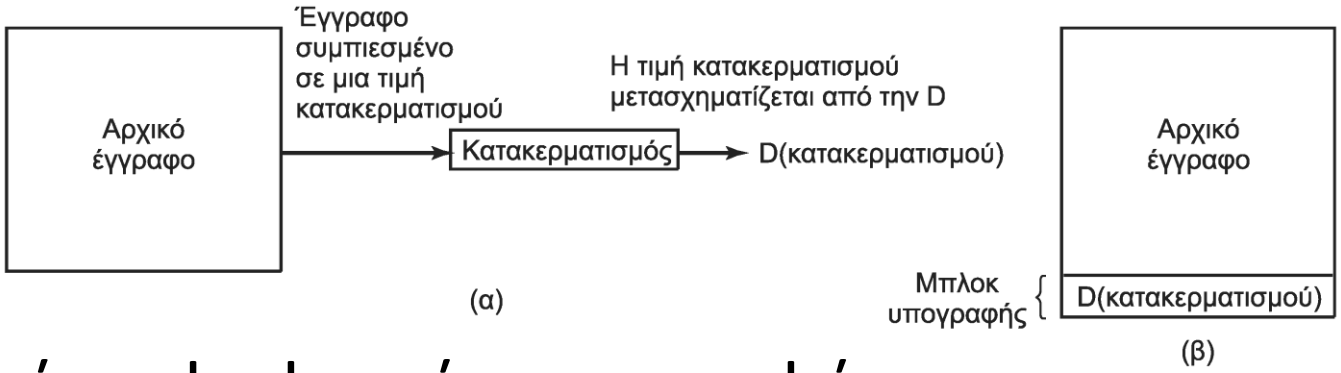
Μονόδρομες συναρτήσεις

- Μία συνάρτηση y=f(x) είναι μονόδρομη όταν
  - Ο υπολογισμός του y από το x είναι απλός
  - Ο υπολογισμός του x από το y δεν είναι
    - Αντίθετα, είναι εξαιρετικά δύσκολος
- Κρυπτογραφική συνάρτηση κατακερματισμού
  - Μονόδρομη συνάρτηση με πρόσθετες ιδιότητες
  - Δεν είναι συνάρτηση κρυπτογραφησης
    - Δεν υπάρχει συνάρτηση αποκρυπτογράφησης

Ψηφιακές υπογραφές (1 από 3)

- Σύνδεση αρχείου με ένα κλειδί
  - Συνήθως, το ιδιωτικό κλειδί του δημιουργού
  - Περνάμε το αρχείο από μονόδρομη συνάρτηση
    - Κρυπτογραφική συνάρτηση κατακερματισμού
  - Παράγεται συμβολοσειρά σταθερού μήκους
  - Κρυπτογράφησή της με το ιδιωτικό κλειδί
    - Το οποίο γνωρίζει μόνο ο δημιουργός
  - Το αποτέλεσμα είναι η ψηφιακή υπογραφή

Ψηφιακές υπογραφές (2 από 3)



- Χρήση ψηφιακής υπογραφής
  - Η υπογραφή επισυνάπτεται στο αρχείο
  - Έστω ότι παραλαμβάνεται αρχείο και υπογραφή
    - Με το δημόσιο κλειδί αποκρυπτογραφείται η υπογραφή
    - Ελέγχεται αν ταιριάζει η τιμή κατακερματισμού
  - Προσοχή: υποθέτει συμμετρία στις συναρτήσεις
    - E(D(x))=x πέραν του D(E(x))=x

Ψηφιακές υπογραφές (3 από 3)

- Πιστοποιητικά (certificates)
  - Πώς γνωρίζουμε το δημόσιο κλειδί κάποιου;
    - Πώς ξέρουμε ότι δεν έχει αλλαχτεί στη διαδρομή;
  - Μπορεί κάποιος έμπιστος να το υπογράψει ψηφιακά
    - Αρκεί να γνωρίζουμε το κλειδί του έμπιστου
- Αρχή πιστοποίησης (certification authority)
  - Οργανισμός που υπογράφει κλειδιά άλλων
  - Το δημόσιο κλειδί του περιέχεται παντού
  - Εναλλακτικά, υποδομή δημόσιου κλειδιού (PKI)

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Μηχανισμοί προστασίας

Μάθημα: Λειτουργικά Συστήματα, **Ενότητα # 7:** Ασφάλεια

Διδάσκων: Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

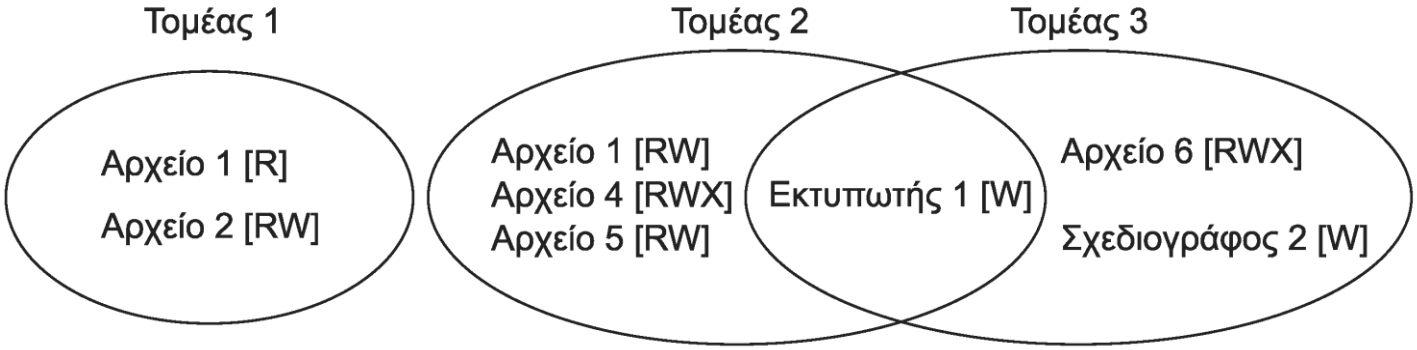
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Τομείς προστασίας (1 από 5)

- Το ΛΣ επιβλέπει πολλά αντικείμενα
- Σε κάθε αντικείμενο εφαρμόζονται μέθοδοι
- Πώς μπορεί να οριστεί ποιος μπορεί να κάνει τι;
- Τομέας: σύνολο (αντικείμενο, δικαιώματα)
- Δικαίωμα: άδεια εκτέλεσης λειτουργίας
  - Ο τομέας αναφέρεται σε χρήστη ή ομάδα χρηστών
- Αρχή ελάχιστης εξουσίας
  - Ο τομέας διαθέτει τα ελάχιστα δυνατά δικαιώματα

Τομείς προστασίας (2 από 5)



- Παράδειγμα με τρεις τομείς (1, 2 και 3)
  - Τα δικαιώματα είναι Read, Write και Execute
- Μία διεργασία εκτελείται πάντα σε κάποιον τομέα
  - Συγκεκριμένα δικαιώματα σε συγκεκριμένα αντικείμενα
- Η διεργασία μπορεί να αλλάζει τομείς
  - Το πώς γίνεται αυτό εξαρτάται από το σύστημα

Τομείς προστασίας (3 από 5)

- Παράδειγμα: UNIX
  - Ο τομέας μίας διεργασίας ορίζεται από τα UID/GID
  - Το κέλυφος παίρνει τα UID/GID του χρήστη
    - Περιέχονται στο αρχείο /etc/passwd
  - Κάθε αντικείμενο αντιπροσωπεύεται από ένα αρχείο
    - Κάθε αρχείο έχει δικαιώματα για UID, GID και όλους
  - Δύο διεργασίες με ίδια UID/GID είναι στον ίδιο τομέα
  - Οι κλήσεις συστήματος αλλάζουν τομέα
    - Η διεργασία εκτελείται στο τμήμα πυρήνα
  - Η εκτέλεση προγράμματος SETUID/SETGID αλλάζει τομέα

Τομείς προστασίας (4 από 5)

Τομέας	Αντικείμενο: Αρχείο 1	Αρχείο 2	Αρχείο 3	Αρχείο 4	Αρχείο 5	Αρχείο 6	Εκτυπωτής 1	Σχεδιογράφος 2
1	Ανάγνωση	Ανάγνωση Εγγραφή						
2			Ανάγνωση	Ανάγνωση Εγγραφή Εκτέλεση	Ανάγνωση Εγγραφή		Εγγραφή	
3						Ανάγνωση Εγγραφή Εκτέλεση	Εγγραφή	Εγγραφή

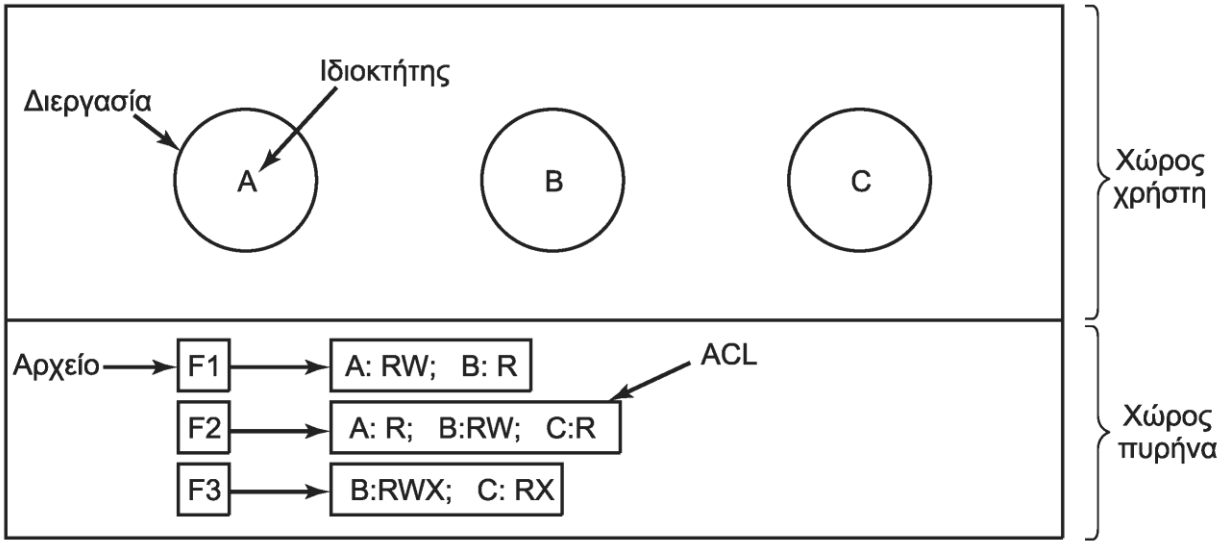
- Πώς παρακολουθεί το σύστημα τα αντικείμενα;
- Μητρώο προστασίας (protection matrix)
  - Οι γραμμές αντιστοιχούν σε τομείς
  - Οι στήλες αντιστοιχούν σε αντικείμενα
  - Σε κάθε κελί περιέχονται τα δικαιώματα

Τομείς προστασίας (5 από 5)

	Αρχείο 1	Αρχείο 2	Αρχείο 3	Αρχείο 4	Αρχείο 5	Αρχείο 6	Εκτυπωτής 1	Σχεδιογράφος 2	Τομέας 1	Τομέας 2	Τομέας 3
Τομέας 1	Ανάγνωση	Ανάγνωση Εγγραφή								Enter	
2			Ανάγνωση	Ανάγνωση Εγγραφή Εκτέλεση	Ανάγνωση Εγγραφή		Εγγραφή				
3						Ανάγνωση Εγγραφή Εκτέλεση	Εγγραφή	Εγγραφή			

- Μοντελοποίηση της εναλλαγής τομέων
  - Στις στήλες έχουμε και τομείς ως αντικείμενα
  - Το δικαίωμα enter σημαίνει αλλαγή τομέα
  - Στην πράξη δεν αποθηκεύουμε το μητρώο
    - Είναι πολύ μεγάλο αλλά πολύ αραιό
    - Αποθήκευση ανά γραμμές ή ανά στήλες

Λίστες ελέγχου πρόσβασης (1 από 4)



- Λίστα δικαιωμάτων που αντιστοιχούν σε αντικείμενο
  - Ουσιαστικά, μια στήλη του μητρώου προστασίας
  - Κάθε στοιχείο δίνει τομέα και δικαιώματα
    - Γενικά: καταστροφή ή αντιγραφή αντικειμένου
    - Ειδικά: προσάρτηση μηνύματος, εκτέλεση αρχείου

Λίστες ελέγχου πρόσβασης (2 από 4)

Αρχείο	Λίστα ελέγχου πρόσβασης
Password	georgia, sysadm: RW
Pigeon_data	bill, pigfan: RW; georgia, pigfan: RW; ...

- Σε τι αντιστοιχούν οι τομείς;
- Τουλάχιστον σε μεμονωμένους χρήστες
  - Λέγονται και υποκείμενα (subjects)
  - Εναλλακτικά, κύριοι (principals)
- Συνήθως έχουμε και ομάδες (groups) χρηστών
  - Ένας χρήστης μπορεί να ανήκει σε διάφορες ομάδες

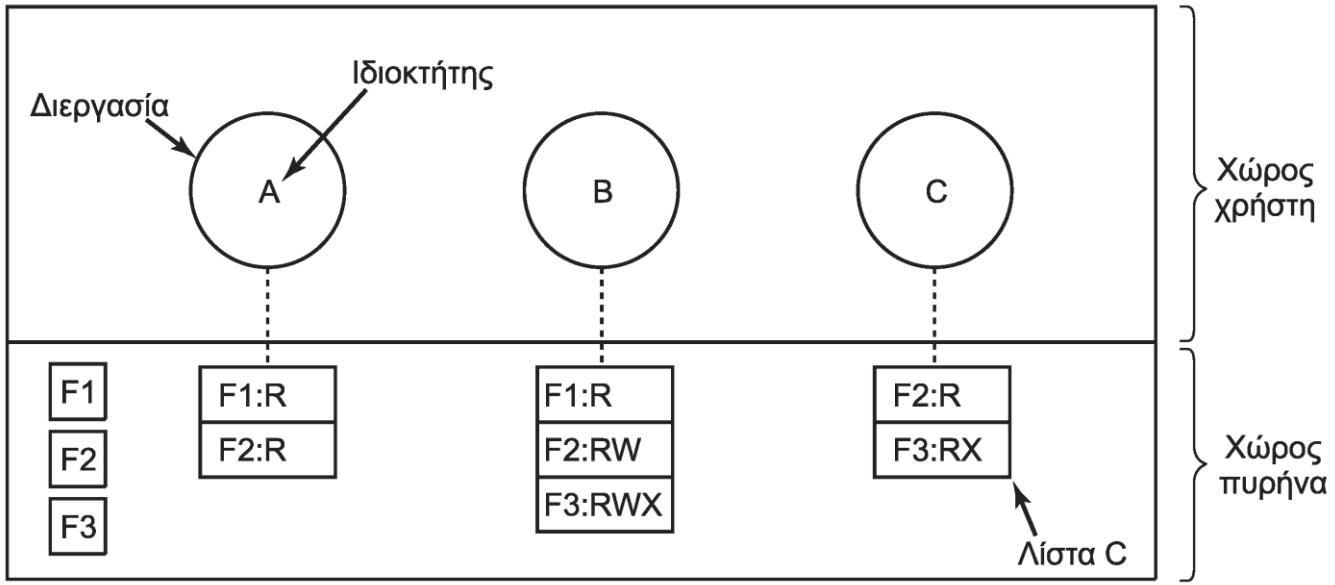
Λίστες ελέγχου πρόσβασης (3 από 4)

- Δικαιώματα ανά UID/GID
  - Ο συνδυασμός UID/GID είναι ένας ρόλος (role)
    - Παράδειγμα: η georgia είναι και sysadm και pigfan
  - Τα δικαιώματα εξαρτώνται από τρέχουσα ομάδα
- Δικαιώματα ανεξαρτήτως ομάδας
  - Παράδειγμα: tana,*: RW

Λίστες ελέγχου πρόσβασης (4 από 4)

- Επιλεκτικός αποκλεισμός ή αρνητικά προνόμια
  - Παράδειγμα: virgil,*:none; *,*:RW
  - Ισχύει η πρώτη καταχώριση της ACL που ταιριάζει
- Τα προνόμια μπορεί να δίνονται ανά GID ή UID
  - UNIX: δικαιώματα για UID, για GID, για όλους
- Πότε ελέγχεται η ACL;
  - Συνήθως κατά το «άνοιγμα» του αντικειμένου
  - Ο αποκλεισμός ισχύει αφού «κλείσει» το αντικείμενο

Δυνατότητες (1 από 3)



- Σε κάθε διεργασία αντιστοιχεί λίστα δικαιωμάτων
  - Ουσιαστικά, μία γραμμή του μητρώου προστασίας
  - Κάθε στοιχείο δίνει αντικείμενο και δικαιώματα
- Οι λίστες δυνατοτήτων είναι και αυτές αντικείμενα

Δυνατότητες (2 από 3)

- Προστασία δυνατοτήτων
  - Οι δυνατότητες σχετίζονται με μια διεργασία
  - Η διεργασία δεν πρέπει να τις αλλάζει!
- Αρχιτεκτονικές με ετικέτες (tagged architectures)
  - Κάθε λέξη της μνήμης έχει μία ετικέτα (tag)
  - Αν είναι 1, η λέξη αλλάζει μόνο από το ΛΣ
- Διατήρηση δυνατοτήτων μέσα στο λειτουργικό
  - Οι διεργασίες αναφέρονται σε θέσεις στη λίστα

Δυνατότητες (3 από 3)

Διακομιστής	Αντικείμενο	Δικαιώματα	f(Αντικείμενα,Δικαιώματα,Έλεγχος)

- Κρυπτογράφηση των δυνατοτήτων
  - Στη δημιουργία παράγεται τυχαίο πεδίο ελέγχου
    - Ο δημιουργός κρατάει τοπικά το πεδίο ελέγχου
  - Ο ιδιοκτήτης λαμβάνει ένα ειδικό αντικείμενο
    - Περιέχει κρυπτογραφική σύνοψη των στοιχείων
  - Ο χρήστης στέλνει το αντικείμενο για χρήση
    - Ο διακομιστής ελέγχει αν τα δικαιώματα είναι ίδια
    - Αν όχι, τότε η κρυπτογραφική σύνοψη δεν ταιριάζει

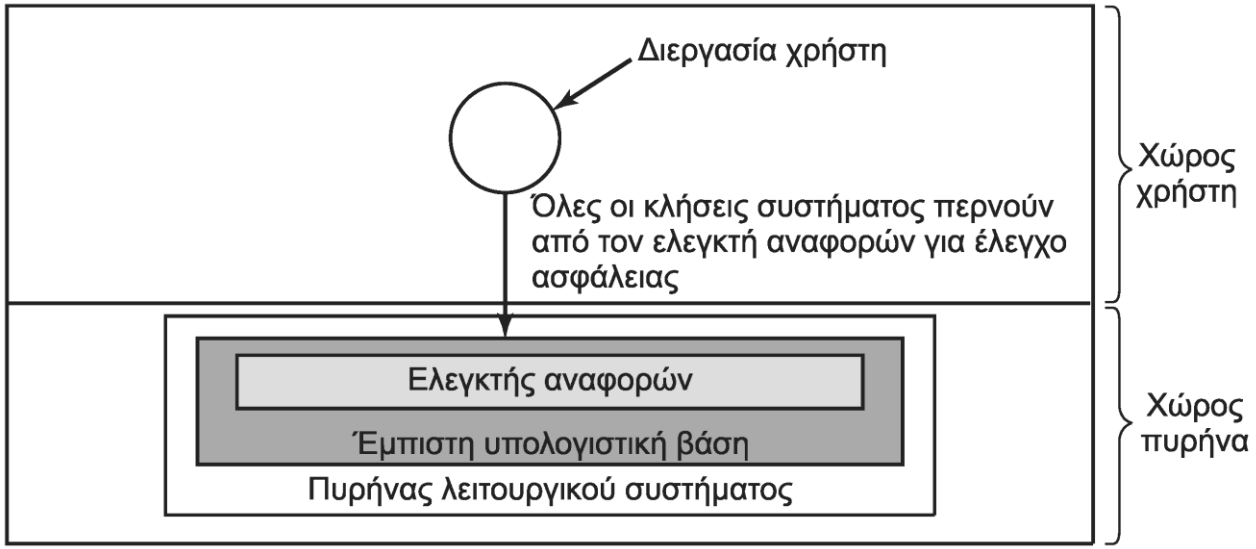
Δυνατότητες (4 από 4)

- Γενικά δικαιώματα για όλα τα αντικείμενα
  - Αντιγραφή ή διαγραφή δυνατότητας / αντικειμένου
- Ανάκληση δικαιωμάτων πρόσβασης
  - Χρήση έμμεσων αντικειμένων για τις διεργασίες
  - Τα έμμεσα δείχνουν στα πραγματικά
  - Η διαγραφή του έμμεσου ανακαλεί τα πάντα
- Δυνατότητες ή λίστες ελέγχου πρόσβασης;
  - Οι δυνατότητες επιτρέπουν πιο γρήγορο έλεγχο
  - Οι λίστες ελέγχου επιτρέπουν επιλεκτική ανάκληση

Έμπιστα συστήματα

- Γιατί τα σύγχρονα ΛΣ έχουν προβλήματα ασφάλειας;
  - Ο τρόπος κατασκευής ασφαλών ΛΣ είναι γνωστός
- Συμβατότητα: ένα νέο ΛΣ θέλει νέες εφαρμογές
  - Δεν θα εκτελεί τα υπάρχοντα προγράμματα
- Πολυπλοκότητα: ένα ασφαλές ΛΣ θα είναι απλό
  - Δεν θα δέχεται ενεργό περιεχόμενο στο e-mail
  - Δεν θα δέχεται ενεργό περιεχόμενο στις ιστοσελίδες
  - Ζητάνε πραγματικά οι χρήστες αυτές τις δυνατότητες;
    - Ή μήπως προστίθενται από τις εταιρείες για λόγους μαρκετινγκ;

Έμπιστη υπολογιστική βάση



- Ορισμένοι οργανισμοί χρειάζονται ασφαλή συστήματα
- Έμπιστη υπολογιστική βάση (trusted computing base)
  - Μικρό τμήμα υλικού και λογισμικού του συστήματος
    - Υλικό και συσκευές που επηρεάζουν την ασφάλεια
    - Πυρήνας και προγράμματα με προνόμια υπερχρήστη
    - Ο ελεγκτής αναφορών (reference monitor) ελέγχει τις κλήσεις

Τυπικά μοντέλα (1 από 3)

- Τα μητρώα προστασίας αλλάζουν δυναμικά
  - Προσθαφαίρεση αντικειμένων, χρηστών, δικαιωμάτων
- Μοντελοποίηση προστασίας με 6 λειτουργίες
  - Create object, delete object
  - Create domain, delete domain
  - Create right, delete right
- Οι λειτουργίες συνδυάζονται σε διαταγές προστασίας
  - Οι εφαρμογές εκτελούν διαταγές, όχι λειτουργίες
  - Παράδειγμα: δημιουργία νέου αρχείου
    - Συνδυάζει create object και create right

Τυπικά μοντέλα (2 από 3)

Αντικείμενα

	Compiler	Mailbox 7	Secret
Eric	Ανάγνωση Εκτέλεση		
Henry	Ανάγνωση Εκτέλεση	Ανάγνωση Εγγραφή	
Robert	Ανάγνωση Εκτέλεση		Ανάγνωση Εγγραφή

(α)

Αντικείμενα

	Compiler	Mailbox 7	Secret
Eric	Ανάγνωση Εκτέλεση		
Henry	Ανάγνωση Εκτέλεση	Ανάγνωση Εγγραφή	
Robert	Ανάγνωση Εκτέλεση	Ανάγνωση	Ανάγνωση Εγγραφή

(β)

- Το μητρώο λέει τι μπορεί να κάνει μία διεργασία
  - Αντιστοιχεί αυτό στην πολιτική του συστήματος;
  - Μήπως η διεργασία τροποποίησε το μητρώο;
- Παράδειγμα: αρχικό και πειραγμένο μητρώο
  - Ο Robert πρόσθεσε Ανάγνωση στο Mailbox7

Τυπικά μοντέλα (3 από 3)

- Έστω ένα ΛΣ με κάποιες διαταγές προστασίας
- Έστω μια εξουσιοδοτημένη κατάσταση μητρώου
- Μπορούμε να αποδείξουμε ότι δεν θα πάμε σε μη εξουσιοδοτημένη κατάσταση;
  - Ρωτάμε αν οι διαταγές προστασίας είναι επαρκείς
  - Αν όχι, το σύστημα δεν είναι θεωρητικά ασφαλές
  - Το γενικό πρόβλημα είναι απροσδιόριστο
  - Μπορεί να βρεθεί απόδειξη για συγκεκριμένο ΛΣ

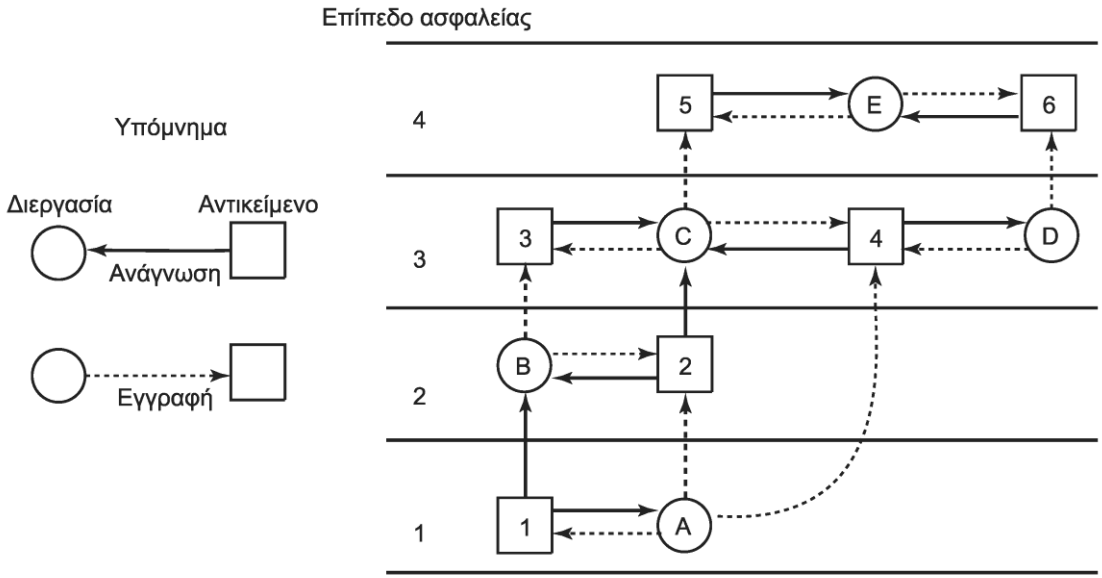
Πολυεπίπεδη ασφάλεια (1 από 3)

- Μοντέλα ασφάλειας
  - Επιλεκτικός έλεγχος πρόσβασης (discretionary)
    - Ο χρήστης αποφασίζει ποιος θα βλέπει τι
    - Συνηθισμένος στα συστήματα γενικής χρήσης
  - Υποχρεωτικός έλεγχος πρόσβασης (mandatory)
    - Επιβολή κανόνων πρόσβασης από το σύστημα
    - Συνηθισμένος σε ευαίσθητα συστήματα

Πολυεπίπεδη ασφάλεια (2 από 4)

- Μοντέλο Bell-La Padula
  - Σχεδιάστηκε για στρατιωτικά συστήματα
  - Διατεταγμένα επίπεδα διεργασιών / αντικειμένων
  - Απλή ιδιότητα ασφάλειας
    - Μία διεργασία στο επίπεδο k μπορεί να διαβάσει <=k
  - Ιδιότητα *
    - Μια διεργασία στο επίπεδο k μπορεί να γράψει >=k
  - Οι διεργασίες διαβάζουν κάτω & γράφουν πάνω
    - Μπορώ να στείλω στον ανώτερο
    - Μπορώ να λάβω από τον κατώτερο

Πολυεπίπεδη ασφάλεια (3 από 4)

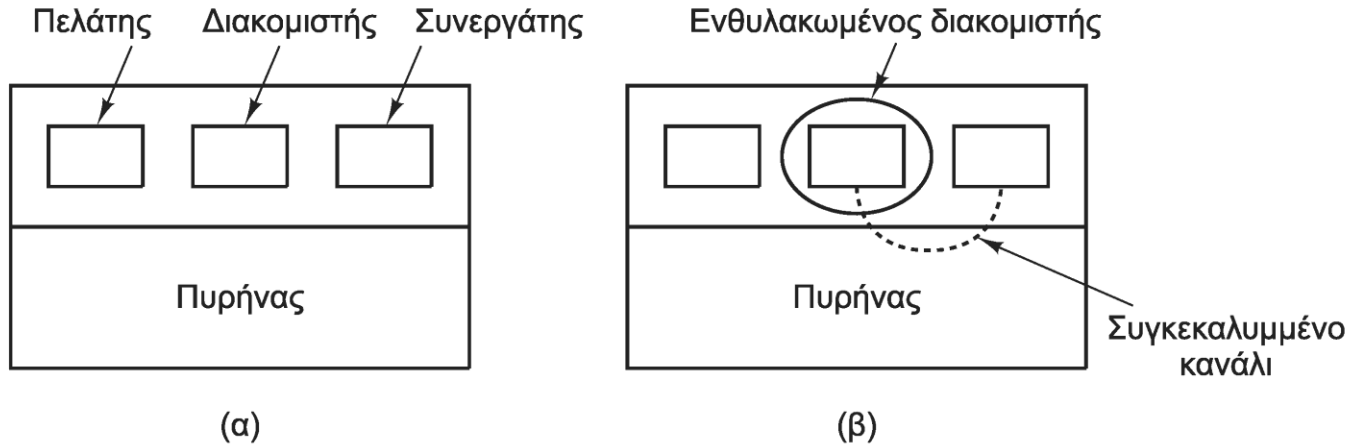


- Παράδειγμα με 4 επίπεδα ασφάλειας
  - Τα συμπαγή βέλη (ανάγνωση) δεν πάνε κάτω
    - Διαβάζουμε χαμηλότερα αντικείμενα
  - Τα διακεκομμένα βέλη (εγγραφή) δεν πάνε κάτω
    - Γράφουμε υψηλότερα αντικείμενα

Πολυεπίπεδη ασφάλεια (4 από 4)

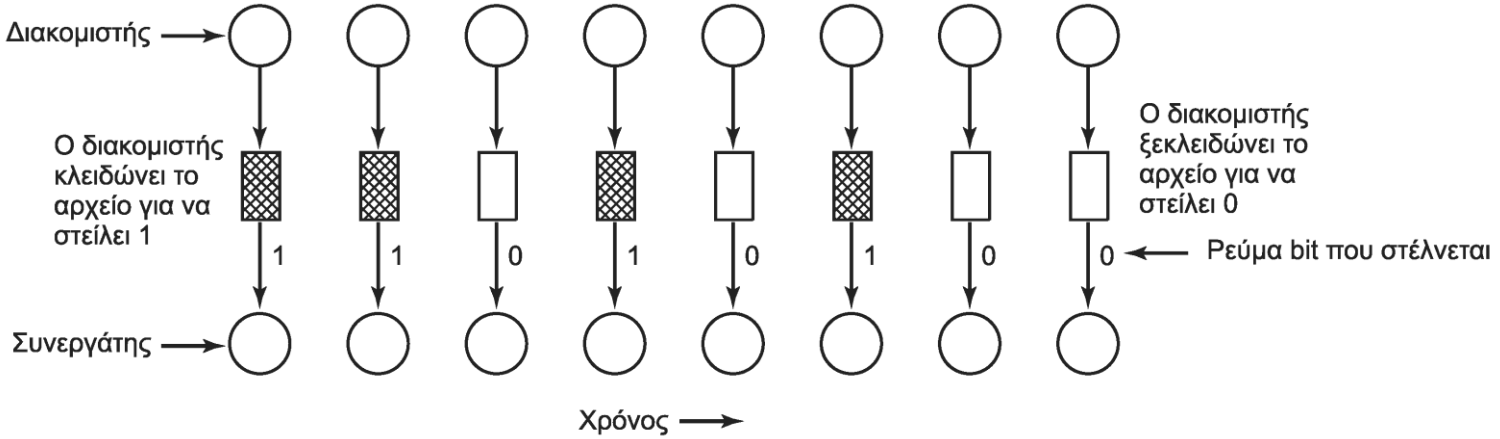
- Μοντέλο Biba
  - Το Bell-La Padula έχει σχεδιαστεί για να κρατάει μυστικά
    - Ενημέρωση από κάτω και αναφορές προς τα πάνω
  - Το Biba εγγυάται την ακεραιότητα των δεδομένων
  - Απλή ιδιότητα ακεραιότητας
    - Μία διεργασία στο επίπεδο k μπορεί να γράψει <=k
  - Ιδιότητα * ακεραιότητας
    - Μια διεργασία στο επίπεδο k μπορεί να διαβάσει >=k
  - Εφαρμογή σε εταιρικό περιβάλλον
    - Ενημέρωση από πάνω και οδηγίες προς τα κάτω

Συγκεκαλυμμένα κανάλια (1 από 5)



- Το πρόβλημα του περιορισμού (containment problem)
  - Ο πελάτης ζητάει από τον διακομιστή κάποια εργασία
  - Για να γίνει η εργασία του αποκαλύπτει πληροφορίες
  - Ο διακομιστής θέλει να τις αποκαλύψει στον συνεργάτη
  - Το μητρώο προστασίας απαγορεύει την επικοινωνία
  - Δυνατότητα επικοινωνίας με συγκεκαλυμμένα κανάλια

Συγκεκαλυμμένα κανάλια (2 από 5)



- Ο διακομιστής σε κάθε διάστημα είναι σε μία κατάσταση
  - Η μία κατάσταση σημαίνει 1 και η άλλη 0
- Παράδειγμα: υπολογίζει πολύ ή υπολογίζει λίγο
  - Ο συνεργάτης στέλνει μηνύματα, βλέπει το χρόνο απόκρισης
- Παράδειγμα: κλειδώνει ή ξεκλειδώνει ένα αρχείο
  - Ο συνεργάτης ελέγχει αν το αρχείο είναι κλειδωμένο

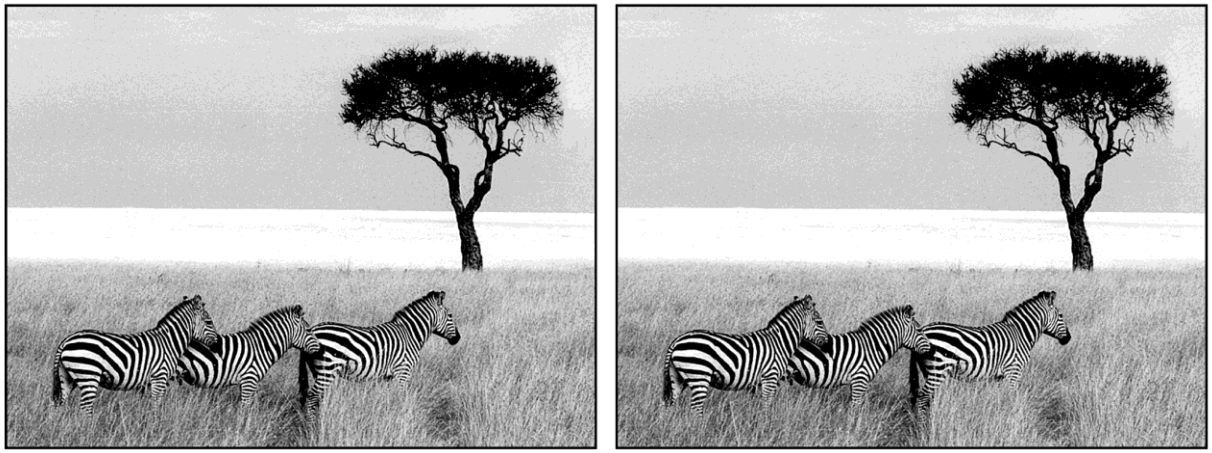
Συγκεκαλυμμένα κανάλια (3 από 5)

- Τα κανάλια αυτά είναι γενικά θορυβώδη
  - Χρήση κώδικα διόρθωσης σφαλμάτων
  - Χρήση επιβεβαιώσεων (αντίστροφη ροή)
- Στεγανογραφία (steganography)
  - Κρύψιμο εμπιστευτικών σε μη εμπιστευτικές
    - Είναι μία μορφή συγκεκαλυμμένου καναλιού
    - Τροποποίηση πληροφοριών με μη εμφανή τρόπο

Συγκεκαλυμμένα κανάλια (4 από 5)

- Παράδειγμα: εικόνα με 24 bit χρώματος
  - Κάθε εικονοστοιχείο έχει 3 κανάλια χρώματος
  - Κάθε κανάλι έχει 8 bit
  - Χρησιμοποιούμε το λιγότερο σημαντικό bit
    - Αλλάζει ανάλογα με την πληροφορία
  - Ανεπαίσθητες αλλαγές στο χρώμα

Συγκεκαλυμμένα κανάλια (5 από 5)



- Παράδειγμα με πραγματική εικόνα
  - Η αριστερή εικόνα είναι η αρχική
  - Η δεξιά περιέχει και 5 έργα του Σαίξπηρ
- Χρήση για εισαγωγή υδατογραφημάτων (watermarks)
  - Απόκρυψη πληροφοριών για τον ιδιοκτήτη της εικόνας
  - Αν χρησιμοποιηθεί η εικόνα, δείχνουμε το υδατογράφημα

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Πιστοποίηση ταυτότητας

Μάθημα: Λειτουργικά Συστήματα, **Ενότητα # 7:** Ασφάλεια

Διδάσκων: Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής

Κωδικοί πρόσβασης (1 από 11)

- Πώς ξέρουμε ποιος είναι ο χρήστης;
  - Δεν πρέπει να μπορεί να παραστήσει άλλον
  - Χρειάζεται μέθοδος πιστοποίησης ταυτότητας
  - Τρεις γενικές μέθοδοι
    - Κάτι που γνωρίζει ο χρήστης
    - Κάτι που κατέχει ο χρήστης
    - Κάτι που είναι ο χρήστης
  - Σε ορισμένα ΛΣ έχουμε συνδυασμό μεθόδων

Κωδικοί πρόσβασης (2 από 11)

- Χάκερ ή κράκερ;
  - Χάκερ είναι ο επιδέξιος προγραμματιστής
    - Μπορεί να λύσει δύσκολα προβλήματα
  - Κράκερ είναι αυτός που διεισδύει σε συστήματα
    - Σπάει την ασφάλεια των συστημάτων
  - Λέμε και white-hat ή black-hat hacker
    - White-hat είναι ο (μάλλον) καλοπροαίρετος
    - Black-hat είναι ο (σίγουρα) κακοπροαίρετος

Κωδικοί πρόσβασης (3 από 11)

(α)	(β)	(γ)
LOGIN: mitch PASSWORD: FooBar!-7 SUCCESSFUL LOGIN	LOGIN: carol INVALID LOGIN NAME LOGIN:	LOGIN: carol PASSWORD: Idunno INVALID LOGIN LOGIN:

- Πιστοποίηση με χρήση κωδικών πρόσβασης
  - Πληκτρολόγηση ονόματος και κωδικού πρόσβασης
    - Το σύστημα διατηρεί μια λίστα με τέτοια ζεύγη
  - Οι κωδικοί πρόσβασης δεν πρέπει να εμφανίζονται
    - Αλλιώς μπορούν να τους μάθουν όσοι βρίσκονται κοντά
  - Το σύστημα πρέπει να δίνει τις ελάχιστες πληροφορίες
    - Στο (β) ο κράκερ μαθαίνει ότι το όνομα σύνδεσης είναι λάθος
    - Στο (γ) μαθαίνει μόνο ότι ο συνδυασμός δεν είναι σωστός

Κωδικοί πρόσβασης (4 από 11)

- Πώς διεισδύουν οι κράκερ;
  - Τα ονόματα χρήστη είναι συνήθως προφανή
    - Παράδειγμα: όνομα.επώνυμο
  - Αλλά και οι (απλοί) κωδικοί δεν είναι δύσκολοι
    - Οι χρήστες συνήθως χρησιμοποιούν κοινές λέξεις
    - Με ένα μικρό λεξικό ανοίγουν πολλοί λογαριασμοί
  - Το πρόβλημα είναι πιο μεγάλο λόγω δικτύων
    - Ένας χρήστης μπορεί να έχει πολλούς λογαριασμούς
    - Συνήθως χρησιμοποιεί τον ίδιο κωδικό παντού

Κωδικοί πρόσβασης (5 από 11)

- Πώς διεισδύουν οι κράκερ;
  - Τηλεφωνητής πολέμου (war dialer)
    - Σύστημα που καλεί τυχαία τηλέφωνα μέσω μόντεμ
    - Πολλοί υπολογιστές δέχονται κλήσεις χωρίς έλεγχο!
  - Παραλλαγή της μεθόδου για το Διαδίκτυο
    - Δοκιμή τυχαίων διευθύνσεων IP με ping και telnet
  - Αντίμετρα από τα συστήματα
    - Απενεργοποίηση του ping
    - Αποσύνδεση του telnet μετά από μερικές αποτυχίες
      - Η επίθεση θέλει περισσότερο χρόνο
      - Οι υπολογιστές όμως έχουν μεγάλη υπομονή!

Κωδικοί πρόσβασης (6 από 11)

- Πώς διεισδύουν οι κράκερ;
  - Σάρωση θυρών (port scan)
    - Δοκιμή σύνδεσης με διάφορες θύρες UDP και TCP
    - Αποκάλυψη υπηρεσιών που τρέχουν σε ένα σύστημα
    - Στη συνέχεια επίθεση στις υπηρεσίες αυτές
  - Χρήση προεπιλεγμένων κωδικών
    - Σε πολλά ΛΣ υπάρχουν λογαριασμοί ειδικού σκοπού
    - Οι κωδικοί είναι προεπιλεγμένοι από τον κατασκευαστή
    - Πολλοί διαχειριστές δεν τους αλλάζουν ποτέ

Κωδικοί πρόσβασης (7 από 11)

```
LBL> telnet elxsi
ELXSI AT LBL
LOGIN: root
PASSWORD: root
INCORRECT PASSWORD, TRY AGAIN
LOGIN: guest
PASSWORD: guest
INCORRECT PASSWORD, TRY AGAIN
LOGIN: uucp
PASSWORD: uucp
WELCOME TO THE ELXSI COMPUTER AT LBL
```

- Παράδειγμα από τον Clifford Stoll (The cuckoo's egg)
  - Ο κράκερ δοκιμάζει τους τυπικούς λογαριασμούς
- Πρόγραμμα εξέτασης πακέτων (packet sniffer)
  - Εγκαθίσταται από τον κράκερ στο σπασμένο σύστημα
  - Παρακολουθεί το δίκτυο και καταγράφει τους κωδικούς

Κωδικοί πρόσβασης (8 από 11)

- Τα σεναριόπαιδα (script kiddies)
  - Αδαείς χρήστες που χρησιμοποιούν έτοιμα σενάρια
    - Αξιοποίηση είτε γνωστών προβλημάτων είτε επαναλήψεων
- Ασφάλεια κωδικών πρόσβασης στο UNIX
  - Η διατήρηση κωδικών σε αρχείο είναι επικίνδυνη
  - Στο UNIX αποθηκεύονται κρυπτογραφημένοι
    - Χρήση μονόδρομης συνάρτησης στον κωδικό
    - Σε κάθε σύνδεση ο κωδικός κρυπτογραφείται και συγκρίνεται
  - Ο εισβολέας μπορεί όμως να κάνει κι αυτός το ίδιο
    - Κρυπτογράφηση βάσης των κοινών λέξεων και σύγκριση

Κωδικοί πρόσβασης (9 από 11)

Bobbie, 4238, e(Dog4238)
Tony, 2918, e(6%%TaeFF2918)
Laura, 6902, e(Shakespeare6902)
Mark, 1694, e(XaB#Bwcz1694)
Deborah, 1902, e(LordByron, 1092)

- Ασφάλεια κωδικών πρόσβασης στο UNIX
  - Η τεχνική του αλατιού (salt)
    - Σε κάθε κωδικό αντιστοιχεί ένας τυχαίος αριθμός
    - Ο τυχαίος αριθμός αποθηκεύεται στο αρχείο κωδικών
    - Ο κωδικός κρυπτογραφείται μαζί με τον τυχαίο αριθμό
    - Η σύγκριση με έτοιμο αρχείο κωδικών δεν δουλεύει
      - Για τυχαίο αριθμό n bit έχουμε 2^n παραλλαγές του κωδικού
  - Στο UNIX η ανάγνωση γίνεται μέσω προγράμματος
    - Το οποίο καθυστερεί επίτηδες τις απαντήσεις του

Κωδικοί πρόσβασης (10 από 11)

- Κωδικοί πρόσβασης μίας χρήσης (one time passwords)
  - Μονόδρομη αλυσίδα κατακερματισμού (hash chain)
    - Χρησιμοποιεί μία μονόδρομη συνάρτηση y=f(x)
    - Ο χρήστης επιλέγει μυστικό κωδικό (s) και πλήθος κωδικών (n)
    - Για n=4 ο πρώτος κωδικός θα είναι P_1=f(f(f(f(s))))
    - Ο διακομιστής ξεκινάει με P_0=f(P_1) και αριθμό 1
    - Πρώτη σύνδεση: ο διακομιστής στέλνει 1 και ο χρήστης P'_1
    - Ο διακομιστής ελέγχει αν f(P'_1)=P_0
    - Αν ναι, αποθηκεύει το P_1 =P'_1 και αριθμό 2
    - Ο χρήστης υπολογίζει εύκολα όλα τα P_i γιατί ξέρει το s
    - Ο διακομιστής δεν μπορεί να υπολογίσει το επόμενο P_i

Κωδικοί πρόσβασης (11 από 11)

- Πιστοποίηση ταυτότητας με ερωταποκρίσεις
  - Ο χρήστης δημιουργεί μία λίστα ερωτήσεων/απαντήσεων
    - Οι απαντήσεις αποθηκεύονται σε κρυπτογραφημένη μορφή
  - Σε κάθε σύνδεση ο διακομιστής επιλέγει μία ερώτηση
  - Ο χρήστης πρέπει να δώσει τη σωστή απάντηση
- Παραλλαγή: ερωταπόκριση (challenge-response)
  - Ο χρήστης επιλέγει έναν αλγόριθμο f και ένα κλειδί k
  - Σε κάθε σύνδεση ο διακομιστής του στέλνει μία τιμή x
  - Ο χρήστης απαντά με την y=f(x,k)
  - Ο εισβολέας πρέπει να ξέρει f (εύκολο) και k (δύσκολο)

Φυσικά αντικείμενα (1 από 2)

- Πιστοποίηση με χρήση φυσικού αντικειμένου
  - Στα ATM έχουμε αντικείμενο (κάρτα) και κωδικό (PIN)
  - Οι απλούστερες κάρτες έχουν μία μαγνητική λωρίδα
    - Οικονομικές, αλλά διαβάζονται και αντιγράφονται εύκολα
  - Οι πιο σύνθετες κάρτες περιέχουν τσιπ μνήμης
    - Η αποθηκευμένη τιμή αλλάζει από τη συσκευή ανάγνωσης
    - Παράδειγμα: τηλεκάρτες με αποθηκευμένο χρόνο ομιλίας
  - Ακόμη πιο σύνθετες είναι οι έξυπνες κάρτες (smart cards)
    - Το τσιπ περιέχει επεξεργαστή (μπορεί να εκτελεί αλγόριθμους)
    - Το τσιπ περιέχει μνήμη (μπορεί να αποθηκεύει και χρήματα)

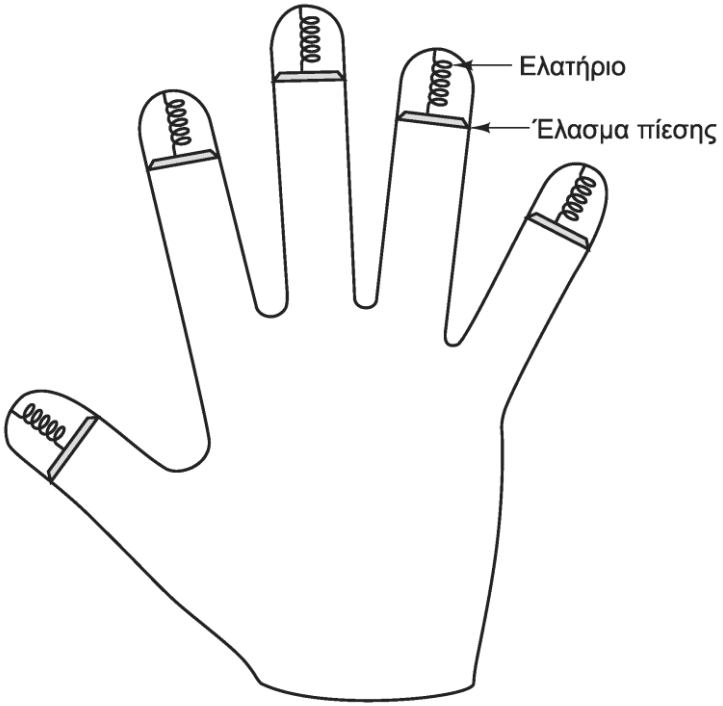
Φυσικά αντικείμενα (2 από 2)

- Πιστοποίηση ταυτότητας με έξυπνες κάρτες
  - Ο διακομιστής στέλνει τυχαίο αριθμό r στον χρήστη
  - Η κάρτα υπολογίζει f(r,k) με βάση το μυστικό κλειδί k
  - Η κάρτα περιέχει και το κλειδί και τον αλγόριθμο
- Ο αλγόριθμος μπορεί να μην είναι σταθερός
  - Αν βρεθεί ότι είναι ευάλωτος, αντικαθίσταται

Βιομετρία (1 από 3)

- Πιστοποίηση ταυτότητας με χρήση βιομετρίας
  - Αξιοποιεί μοναδικά χαρακτηριστικά του χρήστη
    - Πρέπει να υπάρχει μεγάλη ποικιλία στους ανθρώπους
    - Πρέπει να μην αλλοιώνονται εύκολα με τον χρόνο
  - Υποσυστήματα εγγραφής και αναγνώρισης
    - Στην εγγραφή αναλύονται τα χαρακτηριστικά
    - Στην αναγνώριση συγκρίνονται με αυτά της βάσης
- Τυπικά χαρακτηριστικά για βιομετρικό έλεγχο
  - Ίριδα: διαφέρει ακόμη και στους δίδυμους
  - Μήκη δακτύλων χεριού

Βιομετρία (2 από 3)



- Πολλές τεχνικές βιομετρίας σπάνε εύκολα
  - Εκμαγείο του χεριού για μήκη δακτύλων
  - Φωτογραφία της ίριδας για αναγνώριση

Βιομετρία (3 από 3)

- Ανάλυση υπογραφής
  - Ο χρήστης γράφει το όνομά του στο τερματικό
    - Αντί για έλεγχο του αποτελέσματος, ελέγχονται οι κινήσεις
    - Η υπογραφή πλαστογραφείται, αλλά οι κινήσεις όχι
- Βιομετρία φωνής
  - Ο χρήστης επαναλαμβάνει μια φράση που του δίνεται
    - Το σύστημα αναλύει τα χαρακτηριστικά της φωνής
    - Δεν μπορεί να χρησιμοποιηθεί μια έτοιμη ηχογράφηση
- Γιατί να μην κάνουμε ανάλυση αίματος;
  - Οι τεχνικές πρέπει να είναι αποδεκτές από τους χρήστες

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Εσωτερικές επιθέσεις

Μάθημα: Λειτουργικά Συστήματα, **Ενότητα # 7:** Ασφάλεια

Διδάσκων: Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής

Λογικές βόμβες

- Επιθέσεις από χρήστες του συστήματος
  - Η πιστοποίηση ταυτότητας κρατάει τους τρίτους εκτός
  - Πολλές επιθέσεις όμως ξεκινάνε από μέσα
- Λογικές βόμβες (logic bombs)
  - Κώδικας που κρύβεται από προγραμματιστή
  - Όσο δουλεύει στην εταιρεία δεν γίνεται τίποτα
  - Όταν φύγει η βόμβα «εκρήγνυται»
    - Μπορεί να αλλοιώνει αρχεία ή προγράμματα
  - Έτσι ο προγραμματιστής εκβιάζει την εταιρεία

Καταπακτές

```
while (TRUE) {
    printf("login: ");
    get_string(name);
    disable_echoing();
    printf("password: ");
    get_string(password);
    enable_echoing();
    v = check_validity(name, password);
    if (v) break;
}
execute_shell(name);
```

(α)

```
while (TRUE) {
    printf("login: ");
    get_string(name);
    disable_echoing();
    printf("password: ");
    get_string(password);
    enable_echoing();
    v = check_validity(name, password);
    if (v || strcmp(name, "zzzzz") == 0) break;
}
execute_shell(name);
```

(β)

- Καταπακτές (trap doors)
  - Κώδικας που παρακάμπτει ελέγχους ασφάλειας
  - Επιτρέπει ειδική σύνδεση στον προγραμματιστή
    - Παράδειγμα: ο zzzzz μπαίνει χωρίς έλεγχο στο σύστημα
  - Αντιμετωπίζεται με επιθεωρήσεις κώδικα (code reviews)

Παραπλανητική σύνδεση

- Παραπλανητική σύνδεση (login spoofing)
  - Πρόγραμμα που παριστάνει τη διαδικασία σύνδεσης
    - Χρησιμοποιείται σε δημόσιους υπολογιστές
  - Ο χρήστης δίνει όνομα και κωδικό ανυποψίαστος
    - Το πρόγραμμα αποθηκεύει τα στοιχεία
    - Εμφανίζει μήνυμα λάθος κωδικού και τερματίζει
  - Στα Windows αποφεύγεται με χρήση CTRL-ALT-DEL
    - Ο κωδικός δεν παγιδεύεται από προγράμματα χρήστη

Αξιοποίηση σφαλμάτων κώδικα

Μάθημα: Λειτουργικά Συστήματα, **Ενότητα # 7:** Ασφάλεια

Διδάσκων: Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

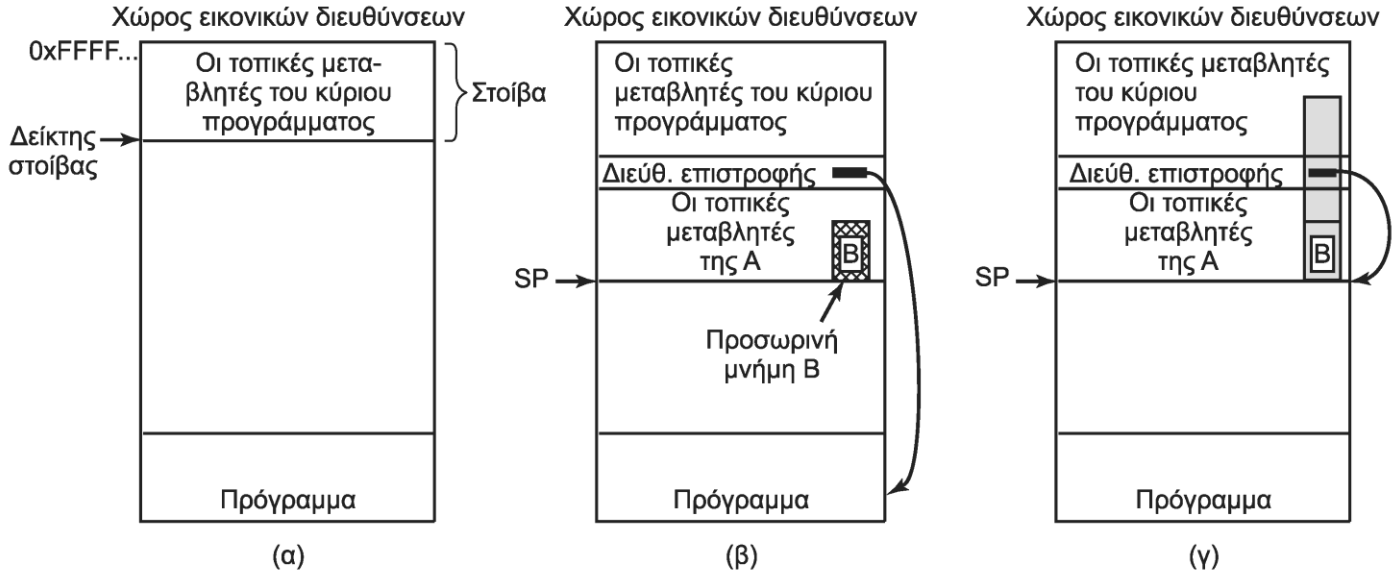
Σφάλματα κώδικα

- Σφάλματα που οδηγούν σε παραβιάσεις ασφάλειας
  - Κάθε σφάλμα είναι διαφορετικό
    - Υπάρχουν όμως ορισμένα πολύ κοινά
  - Αξιοποίηση των σφαλμάτων κώδικα
    - Εκτέλεση σάρωσης θυρών για telnet
    - Δοκιμή σύνδεσης μαντεύοντας όνομα και κωδικό
    - Εκτέλεση προβληματικού προγράμματος με κατάλληλη είσοδο
    - Δημιουργία κελύφους με προνόμια διαχειριστή
    - Προσκόμιση προγράμματος ζόμπι που περιμένει εντολές
    - Αυτόματη ενεργοποίηση του προγράμματος ζόμπι

Υπερχείλιση μνήμης (1 από 3)

- Υπερχείλιση προσωρινής μνήμης
  - Η C δεν ελέγχει τα όρια των πινάκων κατά την πρόσβαση
  - Ένα πρόγραμμα μπορεί να γράψει μνήμη εκτός ορίων
- Λειτουργία επίθεσης μέσω υπερχείλισης;
  - Έστω ότι ζητάμε κάποια παράμετρο από το χρήστη
  - Το πρόγραμμα έχει πίνακα 1024 byte για την είσοδο
  - Ο εισβολέας όμως στέλνει 2000 byte
  - Αν δεν γίνει έλεγχος, αυτά γράφονται εκτός ορίων
    - Αντικαθιστούν τη διεύθυνση επιστροφής της συνάρτησης
    - Περιλαμβάνουν και τον κώδικα στον οποίο επιστρέφουμε

Υπερχείλιση μνήμης (2 από 3)



- Παράδειγμα επίθεσης με υπερχείλιση μνήμης
  - Η παράμετρος γράφεται σε όλη την γκρι περιοχή
    - Αλλάζει η διεύθυνση επιστροφής από τη συνάρτηση
    - Το πρόγραμμα μεταβαίνει σε κώδικα στη γκρι περιοχή
    - Τελικά παίρνει τον έλεγχο ο κώδικας του εισβολέα

Υπερχείλιση μνήμης (3 από 3)

- Λειτουργεί με είσοδο οποιασδήποτε μορφής
  - Αρκεί ο εισβολέας να φτιάξει την σωστή συμβολοσειρά
  - Ο εισβολέας ουσιαστικά εκτελείται με ξένα προνόμια
    - Μπορεί να έχει πάρει προνόμια διαχειριστή
    - Σε αυτή την περίπτωση κάνει ό,τι θέλει
  - Δυστυχώς η gets() δεν ελέγχει το μήκος της εισόδου!
- Εντοπισμός των προβλημάτων υπερχείλισης
  - Δίνουμε μεγάλη είσοδο στο πρόγραμμα
  - Αν καταρρεύσει εξετάζουμε την εικόνα μνήμης του
  - Βρίσκουμε το πρόβλημα στον πηγαίο κώδικα

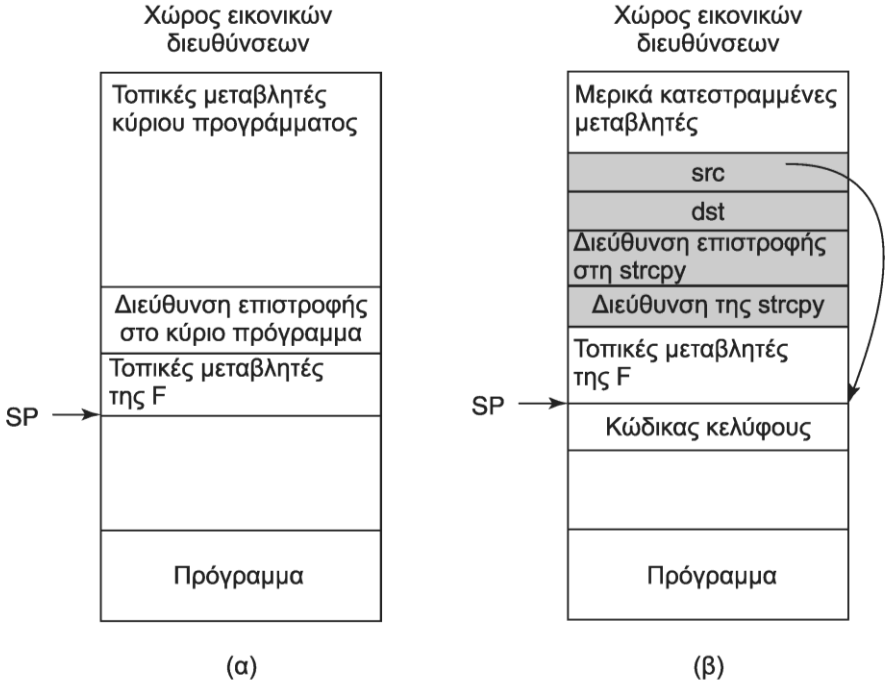
Συμβολοσειρές μορφοποίησης

- Παρόμοια ιδέα με υπερχείλιση μνήμης
  - Η printf() μπορεί να αλλάξει τα ορίσματά της
    - Ο κωδικός %n υπολογίζει πόσοι χαρακτήρες τυπώνονται
    - Ο αριθμός τοποθετείται στο επόμενο όρισμα της printf
    - Παράδειγμα: printf(“Hello %nworld\n”,&i);
  - Έστω το εξής σενάριο
    - Το πρόγραμμα διαβάζει μια συμβολοσειρά s
    - Στη συνέχεια την τυπώνει με printf(s) αντί printf(“%s”,s);
    - Ο εισβολέας περιλαμβάνει κωδικούς μορφοποίησης στην s
    - Έτσι καταφέρνει να τροποποιήσει τη μνήμη

Επιστροφή στη libc (1 από 2)

- Νέα συστήματα δεν επιτρέπουν εκτέλεση δεδομένων
  - Η στοίβα σημειώνεται ως read/write αλλά όχι execute
  - Το ΛΣ μπορεί να αποφύγει τις προηγούμενες επιθέσεις
- Η διεύθυνση επιστροφής μπορεί να αντικατασταθεί
  - Δεν μπορεί να εκτελεστεί κώδικας από τη στοίβα
  - Μπορεί να εκτελεστεί όμως ο κώδικας της libc
    - Συγκεκριμένα, μπορεί να εκτελεστεί η strcpy()
  - Το κέλυφος αντιγράφεται στη στοίβα με υπερχείλιση
  - Η strcpy() το αντιγράφει στο τμήμα δεδομένων

Επιστροφή στη libc (2 από 2)



- Επίθεση με επιστροφή στη libc
  - Αρχικά «επιστρέφουμε» στην strcpy()
  - Αυτή κάνει την αντιγραφή από το src στο dst
  - Στο τέλος επιστρέφει στον αντιγραμμένο κώδικα

Υπερχείλιση ακεραίων

- Όλοι οι τύποι αριθμών έχουν σταθερά μεγέθη
- Η C όμως δεν ελέγχει για υπερχειλίσεις
  - Οι απρόσημοι αναδιπλώνονται από το μηδέν
  - Οι προσημασμένοι μπορεί να αλλάξουν πρόσημο
- Ο εισβολέας μπορεί να προκαλέσει υπερχείλιση
  - Παράδειγμα: ζητάμε ύψος και πλάτος εικόνας
  - Ο εισβολέας προκαλεί υπερχείλιση ακεραίου
  - Το πρόγραμμα δεσμεύει πολύ λίγη μνήμη
  - Δυνατότητα επίθεσης υπερχείλισης προσωρινής μνήμης

Παρεμβολή κώδικα

```
int main(int argc, char *argv[])
{
        char src[100], dst[100], cmd[205] = "cp ";      /* δήλωση 3 συμβολοσειρών */
        printf("Όνομα αρχείου προέλευσης: ");           /* αίτηση για αρχείο προέλευσης */
        gets(src);                                      /* λήψη εισόδου από πληκτρολόγιο */
        strcat(cmd, src);                               /* συνένωση src μετά από cp */
        strcat(cmd," ");                                /* κενό διάστημα στο τέλος του cmd */
        printf("Όνομα αρχείου προορισμού: ");           /* αίτηση για όνομα αρχείου εξόδου */
        gets(dst);                                      /* λήψη εισόδου από πληκτρολόγιο */
        strcat(cmd, dst);                               /* συμπλήρωση συμβολοσειράς διαταγών */
        system(cmd);                                    /* εκτέλεση της διαταγής cp */
}
```

- Επιθέσεις με παρεμβολή κώδικα
  - Έστω ένα πρόγραμμα που χρησιμοποιεί τη system()
  - Θέλουμε είναι ο χρήστης να δώσει δύο ονόματα αρχείων
  - Ο εισβολέας όμως δίνει “abc” και “xyz; rm –rf /”
  - Ουσιαστικά εκτελείται μια δεύτερη εντολή!

Κλιμάκωση προνομίων

- Ο εισβολέας ξεγελάει το ΛΣ για να πάρει προνόμια
- Ο cron εκτελεί περιοδικές εργασίες για τους χρήστες
  - Εκτελείται με δικαιώματα διαχειριστή
  - Διαβάζει προγραμματισμένες εργασίες από κατάλογο
    - Οι χρήστες δεν μπορούν να γράψουν στον κατάλογο
- Ο εκτελεί το πρόγραμμά του στον κατάλογο του cron
  - Καταρρέει αφήνοντας την εικόνα της μνήμης του
  - Η εικόνα γράφεται στον τρέχοντα κατάλογο
    - Τελικά καταλήγει στον κατάλογο του cron
  - Η εικόνα έχει τη μορφή εντολών cron

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Κακόβουλο λογισμικό

Μάθημα: Λειτουργικά Συστήματα, **Ενότητα # 7:** Ασφάλεια

Διδάσκων: Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Είδη κακόβουλου λογισμικού (1 από 2)

- Κακόβουλο λογισμικό (malware)
  - Κάποτε φτιαχνόταν από κακόβουλους ερασιτέχνες
  - Τώρα φτιάχνεται από κακόβουλους επαγγελματίες
- Μπορεί να εγκαθιστά κερκόπορτα στο σύστημα
  - Επιτρέπει τον έλεγχο της μηχανής από τρίτους
    - Μια τέτοια μηχανή λέγεται ζόμπι ή bot
  - Τα δίκτυα bot ενοικιάζονται για μαζικές εργασίες
    - Παράδειγμα: μαζική αποστολή spam
- Μπορεί να εγκαθιστά καταγραφέα πληκτρολογήσεων
  - Κωδικοί / αριθμοί πιστωτικών καρτών στέλνονται αλλού

Είδη κακόβουλου λογισμικού (2 από 2)

- Μπορεί να κάνει υποκλοπή ταυτότητας (identity theft)
  - Συλλογή δεδομένων χρήστη με keylogger και αναζήτηση
  - Ο εισβολέας μπορεί μετά να παραστήσει τον χρήστη
- Και πολλές άλλες παραλλαγές…
  - Μεταφορά χρημάτων από το λογαριασμό του χρήστη
  - Βιομηχανική κατασκοπεία ή σαμποτάζ
- Γιατί υπάρχει τόσο πολύ κακόβουλο λογισμικό;
  - Υπάρχουν λίγα λειτουργικά συστήματα – στόχοι
  - Πολλά θυσιάζουν την ασφάλεια χάρη της ευχρηστίας
    - Επιτρέπουν σύνδεση χωρίς ή με προφανείς κωδικούς

Δούρειοι ίπποι (1 από 2)

- Δούρειοι ίπποι (trojan horses)
  - Κακόβουλο λογισμικό κρυμμένο σε χρήσιμο πρόγραμμα
- Οι χρήστες εγκαθιστούν οι ίδιοι το πρόγραμμα
- Εναλλακτικά, το εκτελούν κατά λάθος
  - Στα περισσότερα ΛΣ υπάρχει μία μεταβλητή PATH
    - Ο εισβολέας βάζει το πρόγραμμα σε εκτελέσιμο κατάλογο
  - Πληκτρολόγηση λάθος ονόματος εντολής
    - Παράδειγμα: ο χρήστης γράφει “la” αντί για “ls”
  - Χρήση τοπικού προγράμματος σε ξένο κατάλογο
    - Σε πολλά συστήματα πρώτα ψάχνεται ο τρέχων κατάλογος

Δούρειοι ίπποι (2 από 2)

- Κοινές απάτες με δούρειους ίππους
  - Μεταφορά χρημάτων από λογαριασμό χρήστη
    - Σε δημοφιλή προγράμματα διαχείρισης χρημάτων
  - Κλήση αριθμών πρόσθετης χρέωσης
    - Κλείνει τον ήχο του μόντεμ για να μην γίνει αντιληπτό
    - Απαιτεί συμφωνία με πάροχο (σε άλλη χώρα)
    - Ο εισβολέας μοιράζεται τα έσοδα με τον πάροχο

Ιοί (1 από 9)

- Ιοί (viruses)
  - Προγράμματα που αναπαράγονται μόνα τους
  - Προσκολλώνται σε εκτελέσιμα προγράμματα
  - Το μολυσμένο πρόγραμμα μολύνει τα άλλα
- Λειτουργία των ιών
  - Αρχικά ο δημιουργός μολύνει ένα πρόγραμμα
  - Μετά το πρόγραμμα διαδίδεται κάπως
  - Όποτε εκτελείται πρώτα μολύνει και άλλα
  - Μετά εκτελεί τον κύριο κώδικά του

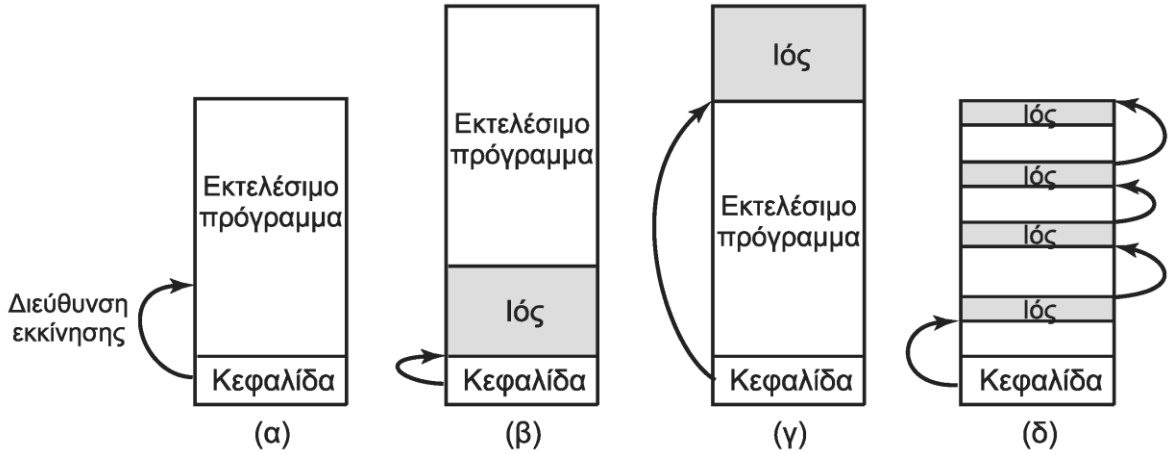
Ιοί (2 από 9)

- Συνοδευτικοί ιοί (companion viruses)
  - Δεν προστίθενται στο πρόγραμμα
  - Εκτελούνται αντί αυτού και μετά το καλούν
  - Παράδειγμα: το αρχείο prog.exe στο MS-DOS
    - Ο ιός είναι στο αρχείο prog.com που καλεί το prog.exe
  - Παράδειγμα: αλλαγή συντόμευσης στα Windows
    - Η συντόμευση δείχνει στον ιό, που εκτελεί το πρόγραμμα
- Ιοί εκτελέσιμου προγράμματος
  - Μολύνουν εκτελέσιμα προγράμματα
  - Πιο συνηθισμένη κατηγορία ιών

Ιοί (3 από 9)

- Ιοί αντικατάστασης (overwriting viruses)
  - Αντικαθιστούν πλήρως το μολυσμένο πρόγραμμα
  - Στη συνέχεια βρίσκουν εκτελέσιμα προγράμματα
  - Αντικαθιστούν τον κώδικά τους με τον δικό τους
  - Συνήθως δεν μολύνουν αμέσως τα πάντα
    - Μολύνουν ορισμένα μόνο εκτελέσιμα (πιθανοτικά)
- Ο ιός αντικατάστασης γίνεται γρήγορα αντιληπτός
  - Ο χρήστης παρατηρεί ότι το πρόγραμμα δεν λειτουργεί
- Ο παρασιτικός ιός προσαρτάται στο πρόγραμμα
  - Το πρόγραμμα λειτουργεί κανονικά παρά τον ιό

Ιοί (4 από 9)

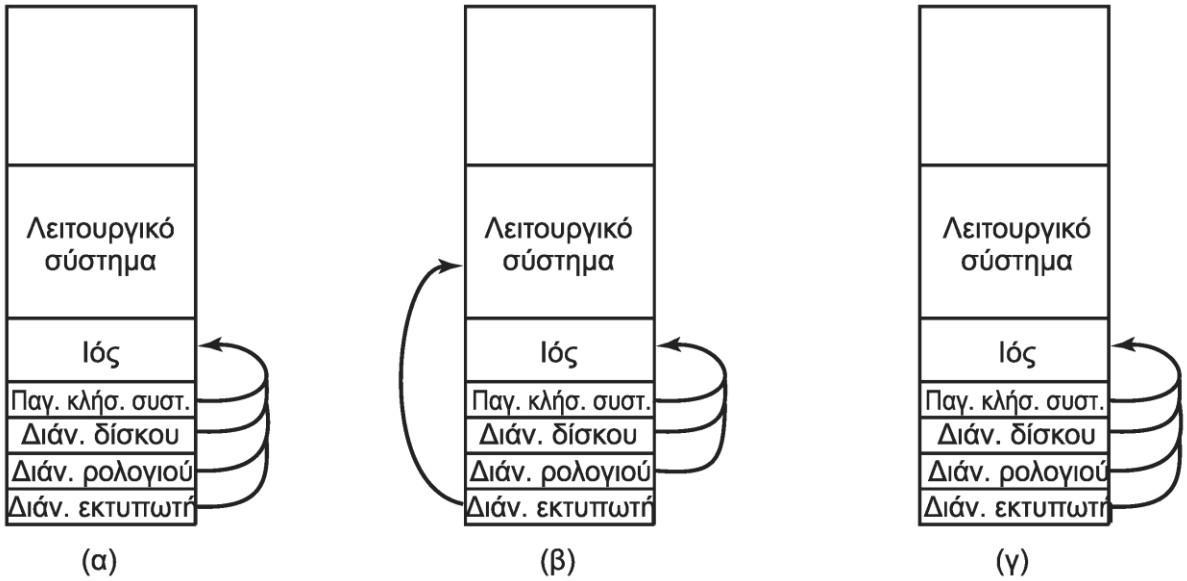


- Παρασιτικοί ιοί (parasitic viruses)
  - Μπαίνει σε αρχή, μέση ή τέλος του προγράμματος
    - Στην αρχή: ίσως να χρειαζόμαστε μετάθεση κώδικα
    - Στο τέλος: αρκεί να είναι μόνο ο ιός μεταθέσιμος
    - Σε αρχή ή τέλος, ο ιός αλλάζει το μέγεθος του αρχείου
    - Μπορεί να κρύβεται στα κενά στο τέλος των τμημάτων

Ιοί (5 από 9)

- Ιοί τομέα εκκίνησης (boot sector viruses)
  - Γράφεται στον τομέα εκκίνησης του λειτουργικού
  - Εκτελείται κατά την εκκίνηση του συστήματος
    - Τη στιγμή αυτή έχει τα προνόμια του υπερχρήστη
    - Καταλαμβάνει διάνυσμα διακοπών
    - Καλείται αυτόματα σε κάθε διακοπή
    - Αφού κάνει τη δουλειά του εκκινεί και το λειτουργικό

Ιοί (6 από 9)



- Ιοί τομέα εκκίνησης (boot sector viruses)
  - Πώς αποφεύγει τη διαγραφή από το λειτουργικό;
    - Αρχικά καταλαμβάνει όλα τα διανύσματα
    - Όπως φορτώνονται οι οδηγοί χάνει κάποια διανύσματα
    - Όποτε καλείται καταλαμβάνει ξανά τα διανύσματα που έχασε

Ιοί (7 από 9)

- Ιοί οδηγών συσκευών (device driver viruses)
  - Ο μολυσμένος οδηγός συσκευής είναι ο τέλειος ιός
    - Φορτώνεται αυτόματα από το λειτουργικό σύστημα
    - Εκτελείται σε κατάσταση πυρήνα με ενισχυμένα προνόμια
- Μακροϊοί ή ιοί μακροεντολών (macro viruses)
  - Πολλά προγράμματα επιτρέπουν τις μακροεντολές
    - Ο ιός μπορεί να προσαρτηθεί στη μακροεντολή Open File
    - Αρκεί να το στείλουμε με e-mail σε έναν εύπιστο χρήστη
  - Τα περισσότερα προγράμματα δίνουν προειδοποίηση
    - Δεν ξέρουμε όμως αν οι μακροεντολές είναι κακόβουλες

Ιοί (8 από 9)

- Ιοί πηγαίου κώδικα (source code viruses)
  - Μολύνουν πηγαίο κώδικα ενός προγράμματος
    - Αρκεί εντολή #include και μία κλήση συνάρτησης
  - Στη μεταγλώττιση ο ιός είναι ήδη εκεί

Ιοί (9 από 9)

- Πώς εξαπλώνονται οι ιοί;
  - Συχνά από μολυσμένο δωρεάν πρόγραμμα
    - Το πρόγραμμα διανέμεται μέσω ιστοσελίδων
    - Όταν εκτελεστεί μολύνει άλλα προγράμματα
    - Μπορεί έτσι να μολυνθούν τομείς εκκίνησης μηχανών
  - Εναλλακτικά, στέλνεται μέσω e-mail
    - Με θέμα το οποίο να τραβάει την προσοχή

Σκουλήκια

- Το σκουλήκι του Internet (1988)
  - Βασίζεται σε δύο κενά ασφάλειας στο BSD UNIX
    - Αρχικά δοκίμαζε να εκτελεστεί με rsh (remote shell)
    - Μετά δοκίμαζε ένα σφάλμα υπερχείλισης στο finger
    - Τέλος δοκίμαζε ένα σφάλμα του sendmail για εκτέλεση e-mail
  - Το σκουλήκι εκτελούσε τον βασικό του κώδικα
  - Ο κώδικας αυτός κατέβαζε τον κυρίως κώδικα
  - Μετά δοκίμαζε να σπάσει όσους κωδικούς μπορούσε
  - Αν έβρισκε αντίγραφό του σε μια μηχανή δεν έκανε νέο
    - Μία φορά στις εφτά όμως, έκανε και νέο αντίγραφο

Λογισμικό κατασκοπίας (1 από 3)

- Φορτώνεται σε υπολογιστή κρυφά από το χρήστη
  - Κρύβεται ώστε να μην μπορεί να εντοπιστεί
  - Συγκεντρώνει στοιχεία για τον χρήστη
  - Μεταδίδει τα στοιχεία αυτά κάπου αλλού
  - Προσπαθεί να αποφύγει την αφαίρεσή του
- Μπορεί να έχει διάφορους στόχους
  - Μάρκετινγκ: παρακολουθεί προτιμήσεις του χρήστη
  - Παρακολούθηση: παρακολουθεί ενέργειες του χρήστη
  - Botnet: κάνει τη μηχανή ζόμπι

Λογισμικό κατασκοπίας (2 από 3)

- Τρόποι εξάπλωσης
  - Συνήθως κρύβεται σε χρήσιμα προγράμματα
  - Μπορεί να κατεβαίνει αυτόματα από ιστοσελίδα
  - Σπάνια οι χρήστες καταλαβαίνουν τι λογισμικό είναι
  - Μερικές φορές βάζει το χρήστη να το αποδεχθεί!
- Τι κάνει το λογισμικό κατασκοπείας;
  - Πειρατεία φυλλομετρητή (browser hijacking)
    - Αλλαγή αρχικής σελίδας φυλλομετρητή
    - Τροποποίηση σελιδοδεικτών φυλλομετρητή
    - Προσθήκη νέων γραμμών εργαλείων στο φυλλομετρητή

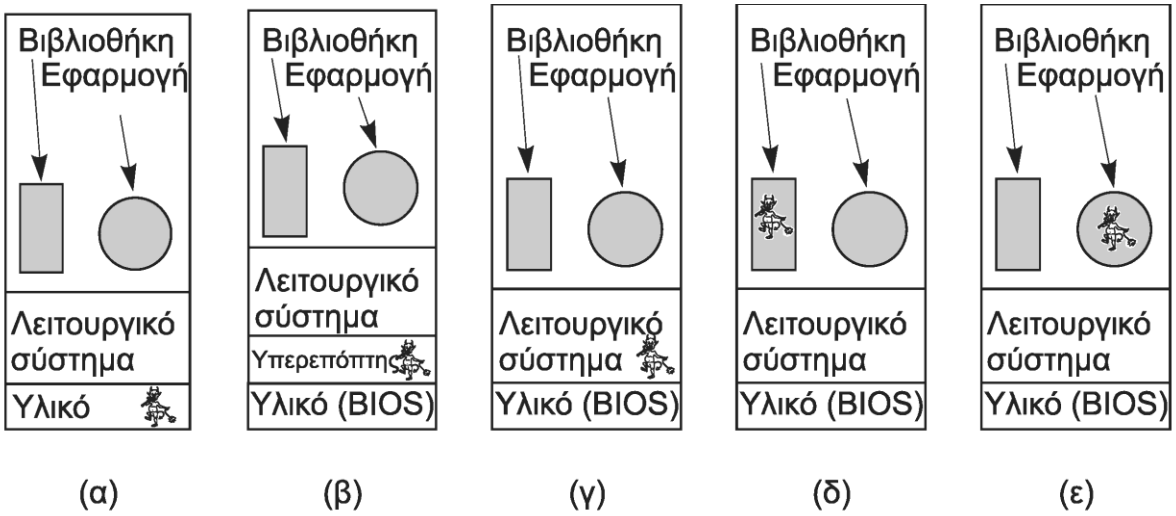
Λογισμικό κατασκοπίας (3 από 3)

- Τι κάνει το λογισμικό κατασκοπείας;
  - Αλλαγή προγράμματος αναπαραγωγής μέσων
  - Αλλαγή μηχανής αναζήτησης χρήστη
  - Προσθήκη νέων εικονιδίων στην επιφάνεια εργασίας
  - Αντικατάσταση διαφημίσεων σε ιστοσελίδες
  - Εισαγωγή διαφημίσεων σε πλαίσια διαλόγου
  - Εμφάνιση αναδυόμενων (pop up) διαφημίσεων
  - Κλείσιμο τείχους προστασίας ή προστασίας από ιούς
  - Δεν είναι το ίδιο με το διαφημιστικό λογισμικό (adware)

Rootkit (1 από 3)

- Προσπαθεί να αποφύγει την αφαίρεσή του
  - Συνήθως, κάποιας μορφής κακόβουλο λογισμικό
- Rootkit υλικολογισμικού (firmware)
  - Κρύβεται στο BIOS σε μνήμη flash
- Rootkit υπερεπόπτη (hypervisor)
  - Εκτελεί το κανονικό ΛΣ ως υπερεπόπτης
- Rootkit πυρήνα (kernel) – πιο συνηθισμένο
  - Κρύβεται σε οδηγό συσκευής ή μέρος του πυρήνα
- Rootkit βιβλιοθήκης (library)
- Rootkit εφαρμογής (application)

Rootkit (2 από 3)



- Τα rootkit υπερεπόπτη είναι δύσκολα στον εντοπισμό
  - Είναι αόρατα στο εκτελούμενο λειτουργικό σύστημα
    - Η συμπεριφορά τους αποκλίνει από αυτή της μηχανής
    - Οι προνομιούχες εντολές είναι αργές λόγω προσομοίωσης
    - Ο υπερεπόπτης χρησιμοποιεί ορισμένους πόρους (π.χ. TLB)

Rootkit (3 από 3)

- Τα rootkit πυρήνα εντοπίζονται με εκκίνηση από USB
  - Σάρωση των αρχείων για περίεργες αλλαγές
  - Μπορεί να συγκρίνονται τα αθροίσματα ελέγχου τους
- Απομάκρυνση του rootkit
  - Είτε προσεκτική αναίρεση των αλλαγών
  - Είτε επανεγκατάσταση του συστήματος
- Το rootkit της Sony BMG
  - Εκτελούνταν αυτόματα από τα CD μουσικής της εταιρείας
  - Μπλόκαρε προγράμματα ανάγνωσης και αντιγραφής
  - Η εταιρεία αναγκάστηκε να πληρώσει αποζημιώσεις

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Τρόποι άμυνας

Μάθημα: Λειτουργικά Συστήματα, **Ενότητα # 7:** Ασφάλεια

Διδάσκων: Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

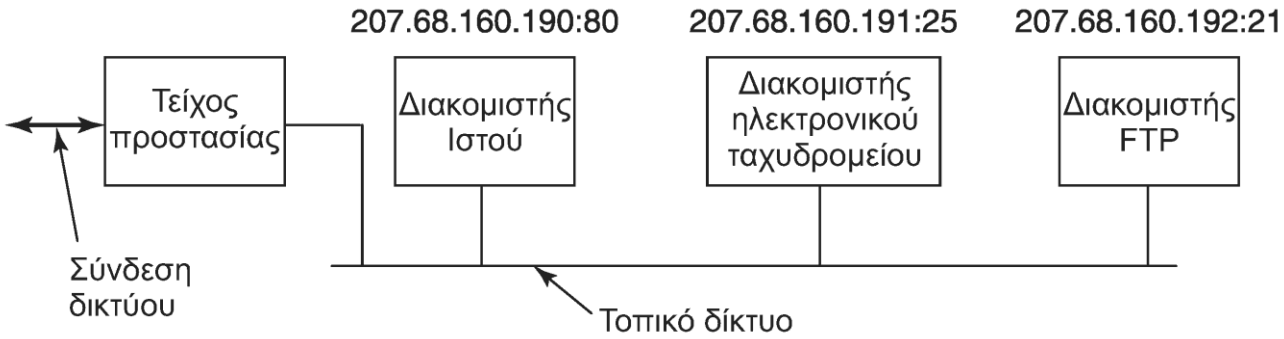
ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Τείχη προστασίας (1 από 4)

- Ασφάλεια σε βάθος: πολλά επίπεδα ασφάλειας
  - Για να εισβάλλει κάποιος θα πρέπει να τα περάσει όλα
- Το Διαδίκτυο εκθέτει το σύστημα σε κινδύνους
  - Εισερχόμενους: κράκερ, κακόβουλο λογισμικό
  - Εξερχόμενους: διαρροή πληροφοριών και κωδικών
- Το τείχος προστασίας προσπαθεί να τους μειώσει
  - Εφαρμογή πολιτικής ασφάλειας στην κίνηση
    - Όλη η κίνηση από/προς το σύστημα ελέγχεται από το τείχος
    - Μόνο η εγκεκριμένη κίνηση επιτρέπεται να περάσει
- Υλοποίηση με υλικό ή λογισμικό

Τείχη προστασίας (2 από 4)



- Τείχη προστασίας υλικού
  - Προστατεύει τη σύνδεση με το Διαδίκτυο
    - Όλη η κίνηση περνάει από το τείχος προστασίας
    - Συνήθως παρέχονται και άλλες λειτουργίες
    - Παράδειγμα: δρομολόγηση
  - Το τι επιτρέπεται να περάσει ορίζεται με κανόνες
    - Μπορεί να ορίζονται μέσω διασύνδεσης Ιστού

Τείχη προστασίας (3 από 4)

- Μη καταστασιακό (stateless) τείχος προστασίας
  - Ελέγχεται η κεφαλίδα των πακέτων και οι κανόνες
    - Παράδειγμα: αποδοχή θύρας 80 για διεύθυνση 207.68.160.190
    - Επιτρέπει επικοινωνία με τον διακομιστή ιστοσελίδων μόνο
  - Ο τελευταίος κανόνας πάντα απαγορεύει οτιδήποτε άλλο
- Το τείχος προστασίας δεν λύνει όλα τα προβλήματα
  - Δεν επιτρέπει τις μη εξουσιοδοτημένες υπηρεσίες
  - Δεν λύνει προβλήματα με εξουσιοδοτημένες υπηρεσίες
    - Παράδειγμα: σφάλμα υπερχείλισης σε διακομιστή
- Όσο λιγότερες θύρες ανοίγουμε, τόσο το καλύτερο

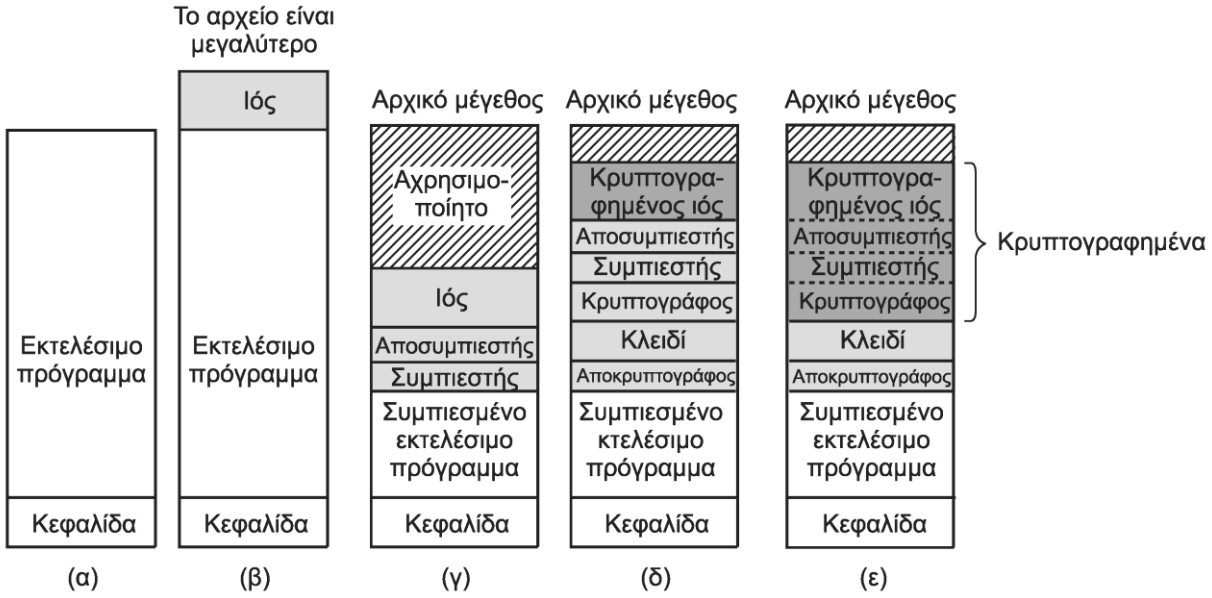
Τείχη προστασίας (4 από 4)

- Καταστασιακό (stateful) τείχος προστασίας
  - Παρακολουθεί και την κατάσταση των συνδέσεων
  - Αντιμετωπίζει επιθέσεις κατά την εγκαθίδρυση
- Σύστημα ανίχνευσης εισβολής (IDS)
  - Ελέγχει το περιεχόμενο των πακέτων
  - Ψάχνει για ύποπτα πακέτα
- Τείχη προστασίας λογισμικού
  - Συνδέονται με κώδικα δικτύωσης του πυρήνα
  - Κατάλληλα για προστασία ενός συστήματος

Προστασία από ιούς (1 από 5)

- Τεχνικές εναντίων των ιών
  - Συνήθως μιλάμε για αντιβιοτικά (antivirus)
    - Για ιούς, σκουλήκια και λογισμικό κατασκοπίας
  - Για κάθε τεχνική αντιβιοτικών, υπάρχει αντίμετρο
    - Συνεχής αγώνας με τα αντιβιοτικά
- Σαρωτές ιών (virus scanners)
  - Οι κατασκευαστές μελετούν συνεχώς νέους ιούς
  - Καταγράφουν σε μία βάση τα χαρακτηριστικά του ιού
  - Αντιπαραβολή προγραμμάτων με τη βάση
    - Πρέπει να ενημερώνονται τακτικά (μέσω δικτύου)

Προστασία από ιούς (2 από 5)



- Ορισμένοι σαρωτές ελέγχουν τα μήκη των αρχείων
- Αντίμετρα από τους ιούς
  - Συμπίεση προγράμματος για να μην αλλάξει μέγεθος
  - Κρυπτογράφηση κώδικα για να μην μοιάζει με τη βάση
  - Ο σαρωτής ψάχνει για τον κώδικα αποκρυπτογράφησης!

Προστασία από ιούς (3 από 5)

```
MOV A,R1
ADD B,R1
ADD C,R1
SUB #4,R1
MOV R1,X
```
(α)

```
MOV A,R1
NOP
ADD B,R1
NOP
ADD C,R1
NOP
SUB #4,R1
NOP
MOV R1,X
```
(β)

```
MOV A,R1
ADD #0,R1
ADD B,R1
OR R1,R1
ADD C,R1
SHL #0,R1
SUB #4,R1
JMP .+1
MOV R1,X
```
(γ)

```
MOV A,R1
OR R1,R1
ADD B,R1
MOV R1,R5
ADD C,R1
SHL R1,0
SUB #4,R1
ADD R5,R5
MOV R1,X
MOV R5,Y
```
(δ)

```
MOV A,R1
TST R1
ADD C,R1
MOV R1,R5
ADD B,R1
CMP R2,R5
SUB #4,R1
JMP .+1
MOV R1,X
MOV R5,Y
```
(ε)

- Πολυμορφικός ιός: αλλάζει λίγο σε κάθε αντιγραφή
- Χρησιμοποιεί μηχανή μετάλλαξης (mutation engine)
  - Προσθήκη κενών εντολών (NOP) (β)
  - Προσθήκη εντολών που δεν έχουν αποτέλεσμα (γ)
  - Προσθήκη περιττών εντολών (δ)
  - Αντιμετάθεση εντολών (ε)

Προστασία από ιούς (4 από 5)

- Ο σαρωτής ελέγχει τομέα εκκίνησης και μνήμη
  - Τι γίνεται όμως αν υπάρχει ένας ισχυρός ιός στη μνήμη;
    - Μπορεί να επιστρέφει τον αρχικό τομέα εκκίνησης
    - Μπορεί να επιστρέφει λάθος στοιχεία για τα αρχεία
  - Η λύση είναι εκκίνηση από άλλο μέσο και έλεγχος
- Ελεγκτές ακεραιότητας (integrity checkers)
  - Υπολογίζουν αθροίσματα ελέγχου των προγραμμάτων
    - Σε κάθε εκτέλεση γίνεται σύγκριση με το αρχείο στο δίσκο
  - Ο ελεγκτής μπορεί να υπογράψει ψηφιακά το αρχείο
    - Ιδανικά χρήση κλειδιού σε εξωτερικό μέσο (έξυπνη κάρτα)

Προστασία από ιούς (5 από 5)

- Ελεγκτές συμπεριφοράς (behavioral checkers)
  - Το αντιβιοτικό είναι συνεχώς στη μνήμη
  - Αναχαιτίζει όλες τις κλήσεις συστήματος
  - Παγιδεύει περίεργη συμπεριφορά
    - Παράδειγμα: εγγραφή τομέα εκκίνησης ή flash
  - Δεν είναι προφανές τι είναι κακό και τι όχι
    - Έγκυρη συμπεριφορά μπορεί να φαίνεται ύποπτη
    - Εγκατάσταση νέας έκδοσης προγράμματος
    - Μακροεντολές χρήστη σε αρχείο

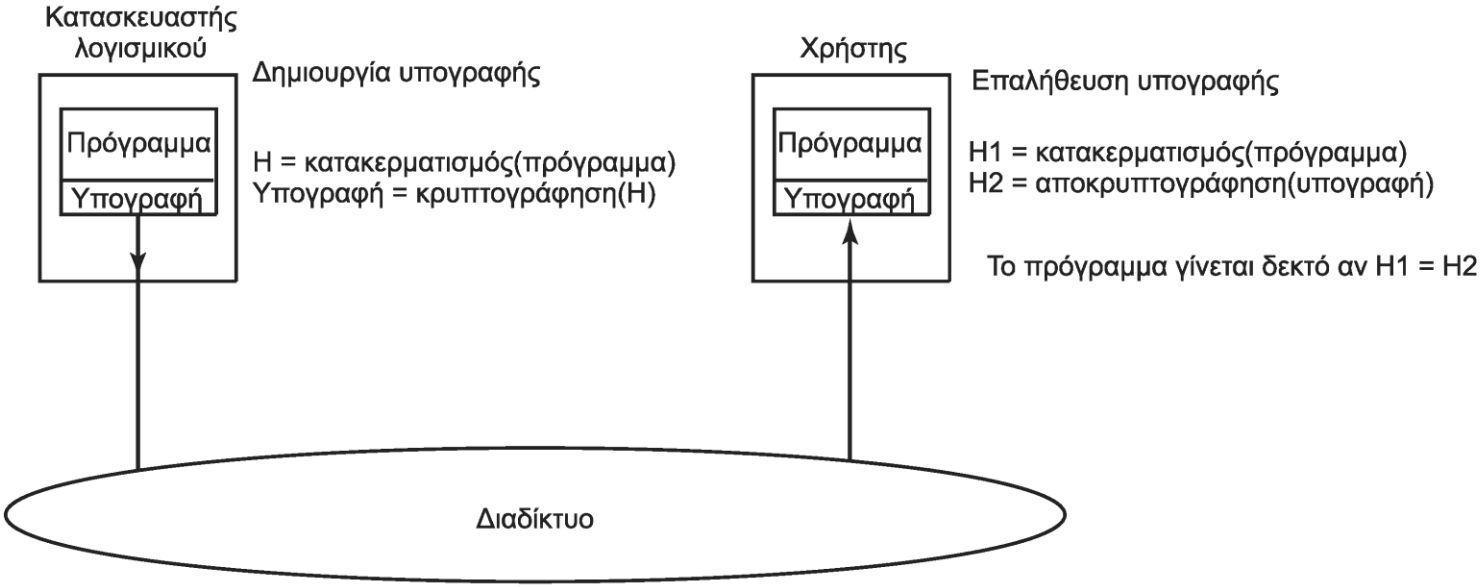
Αποφυγή ιών (1 από 2)

- Οι εταιρείες μπορούν να βοηθήσουν αρκετά
  - Τα απλά ΛΣ είναι πιο ασφαλή
  - Το ενεργό περιεχόμενο είναι γενικά επικίνδυνο
  - Ο βασικός τομέας πρέπει να προστατεύεται
  - Η μνήμη flash πρέπει να προστατεύεται
    - Ιδανικά, με φυσικό διακόπτη
- Οι χρήστες πρέπει να προσέχουν τι κάνουν
- Προτιμάμε τα ασφαλή λειτουργικά συστήματα
  - Πρέπει να διακρίνουν κατάσταση χρήστη και πυρήνα
  - Ο πυρήνας πρέπει να είναι όσο γίνεται μικρός

Αποφυγή ιών (2 από 2)

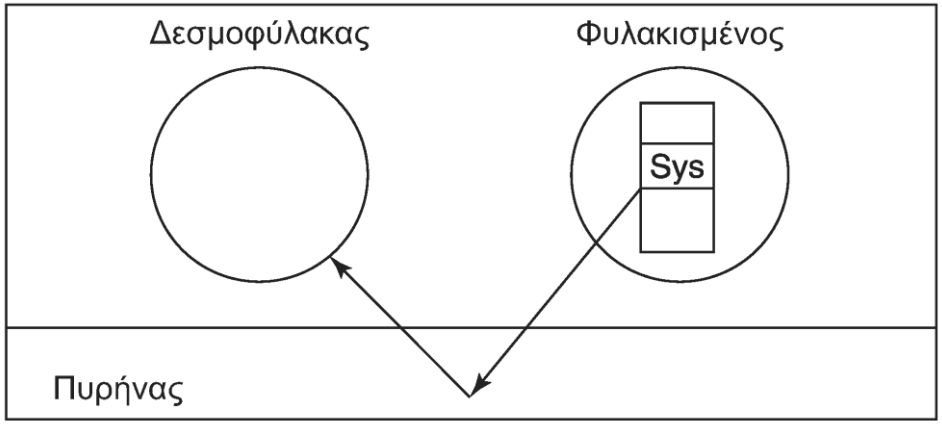
- Εγκαθιστούμε λογισμικό από αξιόπιστες πηγές
- Χρησιμοποιούμε και ενημερώνουμε το αντιβιοτικό
- Δεν ανοίγουμε αρχεία με ενεργό περιεχόμενο
  - Απενεργοποιούμε μακροεντολές
- Παίρνουμε συχνά αντίγραφα ασφαλείας
  - Κρατάμε περισσότερα από ένα
    - Αλλιώς μπορεί να είναι μολυσμένο και το αντίγραφο
- Δεν εκτελούμε λογισμικό από άγνωστες πηγές
  - Κλασική μέθοδος διανομής ιών

Υπογραφή κώδικα



- Πώς ξέρουμε ότι κάποιο λογισμικό είναι σωστό;
  - Ψηφιακή υπογραφή του από τον κατασκευαστή
    - Υπογράφεται η σύνοψη κώδικα με το ιδιωτικό κλειδί
    - Ο χρήστης επιβεβαιώνει την υπογραφή
    - Απαιτεί μία μέθοδο διανομής κλειδιών

Φυλάκιση



- Το πρόγραμμα εκτελείται υπό έλεγχο (φυλακισμένος)
  - Παρακολουθείται από έμπιστη διεργασία (δεσμοφύλακα)
  - Οι κλήσεις συστήματος ελέγχονται από τον δεσμοφύλακα
  - Ο δεσμοφύλακας επιτρέπει μόνο τις ασφαλείς κλήσεις
    - Δίνονται συγκεκριμένα προνόμια στον φυλακισμένο
  - Υλοποιείται σαν πρόγραμμα εκσφαλμάτωσης

Ανίχνευση εισβολής (1 από 3)

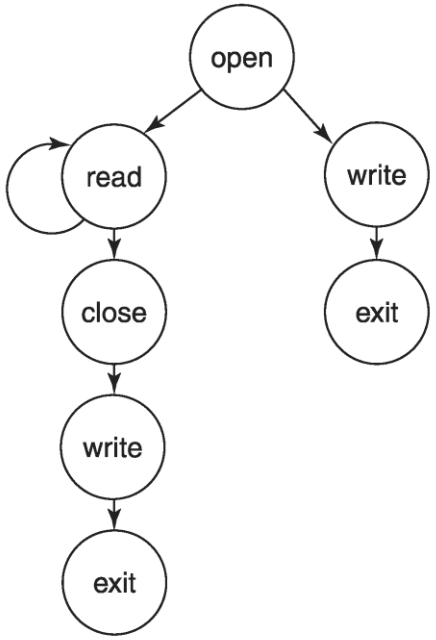
- Υλοποιείται από συστήματα IDS
  - Πιο προχωρημένα από τα IDS δικτύου
- Στατική ανίχνευση εισβολής με βάση μοντέλο
  - Υλοποιείται με την τεχνική φυλάκισης
  - Το πρόγραμμα μοντελοποιείται με γράφο κλήσεων
  - Ο γράφος παράγεται από τον μεταγλωττιστή
  - Ο γράφος καθορίζει ορισμένες ακολουθίες κλήσεων
  - Ο δεσμοφύλακας ελέγχει αν ακολουθείται
  - Αν έχουμε υπερχείλιση ή ιό, η ακολουθία αλλάζει
  - Ο δεσμοφύλακας εντοπίζει το πρόβλημα

Ανίχνευση εισβολής (2 από 3)

```
int main(int argc, *char argv[])
{
  int fd, n = 0;
  char buf[1];

  fd = open("data", 0);
  if (fd < 0) {
    printf("Άκυρο αρχείο δεδομένων\n");
    exit(1);
  } else {
    while (1) {
      read(fd, buf, 1);
      if (buf[0] == 0) {
        close(fd);
        printf("n = %d\n", n);
        exit(0);
      }
      n = n + 1;
    }
  }
}
```

(α)



(β)

Παράδειγμα: πρόγραμμα και γράφος κλήσεων

Ανίχνευση εισβολής (3 από 3)

- Δοχείο μελιού (honeypot)
  - Σύστημα ελάχιστα προστατευμένο
    - Επίτηδες αφήνουμε ανοιχτό το σύστημα
  - Φαίνεται να έχει «ενδιαφέρον» περιεχόμενο
    - Προσωπικά ή οικονομικά δεδομένα
    - Προφανώς, είναι ψεύτικα!
  - Περιμένουμε να δεχτεί επιθέσεις
  - Το IDS μελετάει έτσι τους επιτιθέμενους

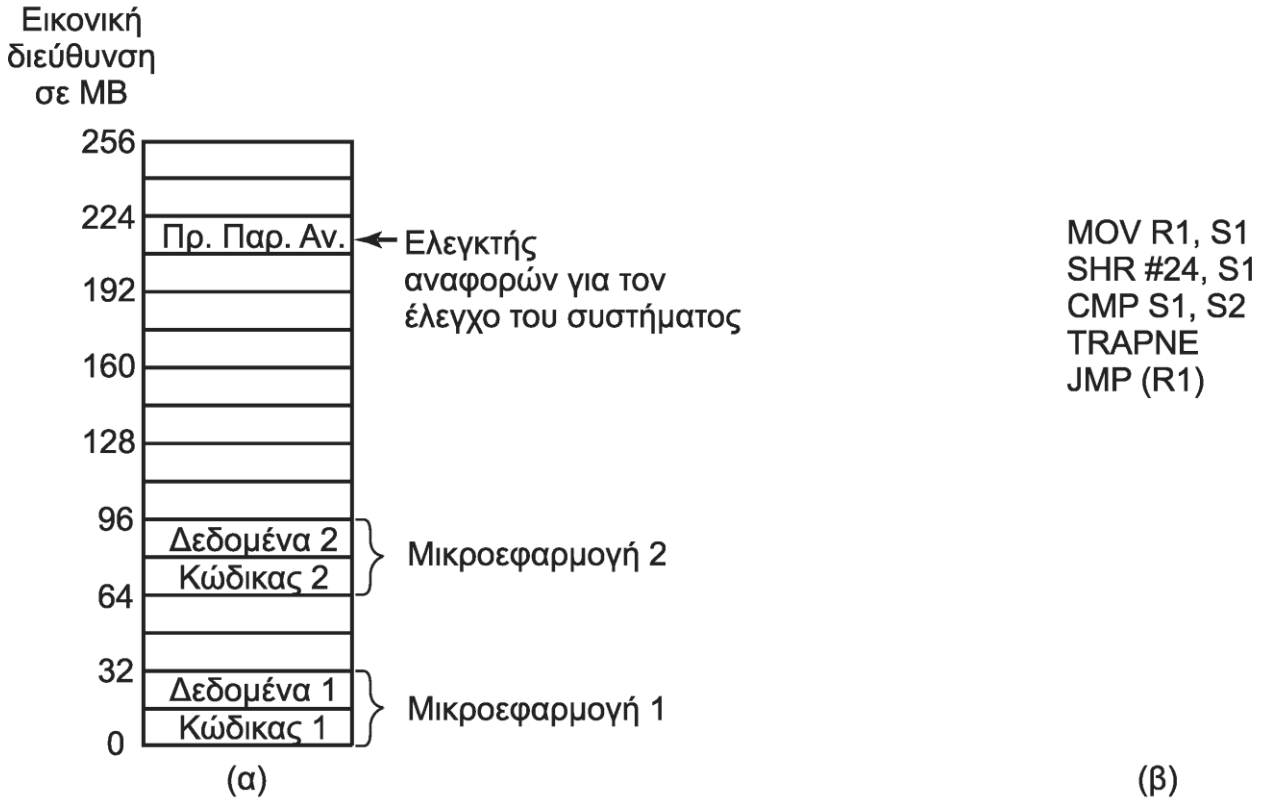
Ενθυλάκωση κινητού κώδικα (1 από 5)

- Διάφορα είδη κινητού κώδικα (mobile code)
  - Μικροεφαρμογές (applets) σε ιστοσελίδες
  - Πράκτορες (agents) που κινούνται στις μηχανές
  - Αρχεία σε γλώσσα PostScript
- Μπορεί να εκτελεστεί ο κώδικας με ασφάλεια;
  - Ναι, αλλά όχι εύκολα
  - Στην τρέχουσα διεργασία θα έχει τα ίδια προνόμια
  - Σε χωριστή διεργασία μπορεί να μην λειτουργεί
- Χρειάζεται γενική μέθοδος ασφαλούς εκτέλεσης

Ενθυλάκωση κινητού κώδικα (2 από 5)

- Αμμοπαγίδα (sandbox)
  - Περιορισμός προγράμματος στη μνήμη
    - Όλες οι διευθύνσεις πρέπει να έχουν το ίδιο πρόθεμα
  - Χρήση συγκεκριμένων περιοχών διευθύνσεων
    - Εκτέλεση κώδικα σε αμμοπαγίδα κώδικα
    - Προσπέλαση δεδομένων σε αμμοπαγίδα δεδομένων
  - Ελεγκτής αναφορών για κλήσεις συστήματος
    - Αλλαγή κλήσεων έτσι ώστε να δείχνουν στον ελεγκτή
    - Αρχείο διάρθρωσης για το τι επιτρέπεται να κληθεί

Ενθυλάκωση κινητού κώδικα (3 από 5)

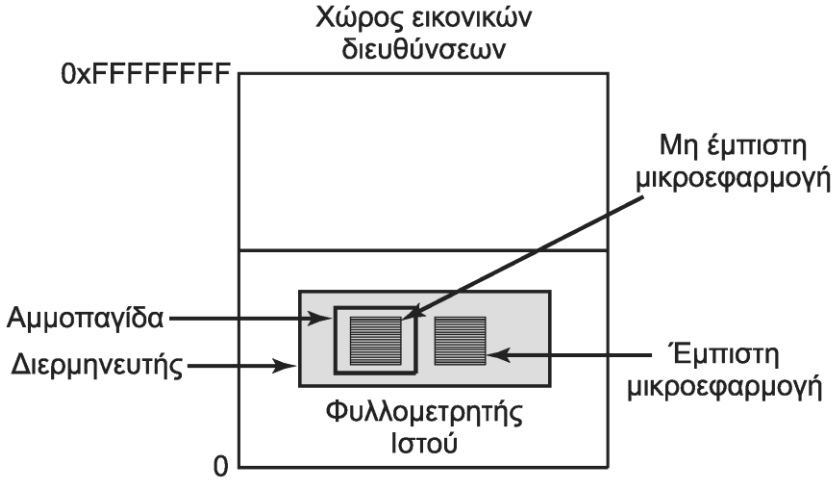


- Κάθε περιοχή έχει διευθύνσεις με κοινό πρόθεμα
  - Παράδειγμα: διαίρεση 256 MB σε 16 περιοχές των 16 MB

Ενθυλάκωση κινητού κώδικα (4 από 5)

- Αμμοπαγίδα (sandbox)
  - Ο κώδικας ελέγχεται εν μέρει στατικά
    - Οι απόλυτες διευθύνσεις πρέπει να είναι έγκυρες
  - Ορισμένα σημεία πρέπει να ελέγχονται δυναμικά
    - Εντολές με σχετικές διευθύνσεις (με βάση τον PC)
  - Προσθήκη κώδικα ελέγχου πριν κάθε αναφορά
    - Αντιγραφή διεύθυνσης σε καταχωρητή ελέγχου 1
    - Απομόνωση του προθέματος με ολίσθηση
    - Σύγκριση με πρόθεμα στον καταχωρητή ελέγχου 2
    - Αν δεν ταιριάζει, παγίδευση της αναφοράς

Ενθυλάκωση κινητού κώδικα (5 από 5)



- Διερμηνεία (interpretation)
  - Οι εντολές ελέγχονται κατά τη διερμηνεία τους
    - Αυτός είναι ο τρόπος που δουλεύει κανονικά η Java
  - Οι έμπιστες εφαρμογές εκτελούνται ελεύθερα
  - Οι μη έμπιστες βρίσκονται σε αμμοπαγίδα

Ασφάλεια στην Java (1 από 3)

- Η Java ελέγχει αυστηρά τους τύπους
  - Δεν επιτρέπει αυτόματες μετατροπές τύπων
  - Δεν επιτρέπει πρόσβαση σε δομές μέσω δεικτών
  - Ελέγχει πλήρως την κατανομή μνήμης
- Εκτέλεση προγραμμάτων σε εικονική μηχανή
  - Μεταγλωττίζονται αρχικά σε ενδιάμεσο κώδικα
  - Ο ενδιάμεσος κώδικας μπορεί
    - Είτε να διερμηνευτεί (δυναμικός έλεγχος)
    - Είτε να μεταγλωττιστεί (στατικός έλεγχος)

Ασφάλεια στην Java (2 από 3)

- Οι εφαρμογές επαληθεύονται πριν εκτελεστούν
  - Έλεγχος του ενδιάμεσου κώδικα (bytecode verifier)
  - Δεν πρέπει να φτιάχνουν δείκτες
  - Δεν πρέπει να βλέπουν ιδιωτικά μέλη κλάσεων
  - Δεν πρέπει να χρησιμοποιούν τύπους λανθασμένα
  - Δεν πρέπει να υπερχειλίζουν/υποχειλίζουν τη στοίβα
  - Δεν πρέπει να κάνουν τυχαίες μετατροπές τύπων
- Οι κλήσεις συστήματος είναι ειδική περίπτωση

Ασφάλεια στην Java (3 από 3)

URL	Υπογράφων	Αντικείμενο	Ενέργεια
www.taxrep.com	TaxRep	/usr/susan/1040.xls	Ανάγνωση
*		/usr/tmp/*	Ανάγνωση, Εγγραφή
www.microsoft.com	Microsoft	/usr/susan/Office/–	Ανάγνωση, Εγγραφή, Διαγραφή

- JDK 1.0: έμπιστες και μη μικροεφαρμογές
  - Οι έμπιστες έκαναν τα πάντα, οι άλλες τίποτα!
- JDK 1.1: υπογραφή κώδικα
  - Αν έχει υπογραφεί από έμπιστή οντότητα, είναι έμπιστη
- JDK 1.2: πιο λεπτομερές μοντέλο ασφάλειας
  - Χρήση πίνακα με προνόμια προσπέλασης
  - Προέλευση εφαρμογής και οντότητα που υπέγραψε
  - Αντικείμενο και τρόπος προσπέλασης

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

Τέλος Ενότητας #7

Μάθημα: Λειτουργικά Συστήματα, **Ενότητα # 7:** Ασφάλεια

Διδάσκων: Γιώργος Ξυλωμένος, **Τμήμα:** Πληροφορικής

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης