Σύστημα Παρακολούθησης και Ελέγχου Συμπεριφοράς για Δίκτυα Ομοτίμων

Νικόλαος Ντάρμος

Α.Μ.: 314

Μεταπτυχιακή Διατριβή

στα πλαίσια του Μεταπτυχιακού Διπλώματος Ειδίκευσης

«Επιστήμη και Τεχνολογία των Υπολογιστών»

του τμήματος Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής

του Πανεπιστημίου Πατρών

Επιβλέπων Καθηγητής

Παναγιώτης Τριανταφύλλου, Καθηγητής

Τριμελής επιτροπή

Ιωάννης Γαροφαλάκης, Επίκουρος Καθηγητής

Παύλος Σπυράκης, Καθηγητής

Παναγιώτης Τριανταφύλλου, Καθηγητής

Πανεπιστήμιο Πατρών

Απρίλιος 2004

Σύστημα Παρακολούθησης και Ελέγχου Συμπεριφοράς για Δίκτυα Ομοτίμων

Εξεταστική Επιτροπή:

Παναγιώτης Τριανταφύλλου, Επιβλέπων

Παύλος Σπυράκης

Ιωάννης Γαροφαλάκης

Σύστημα Παρακολούθησης και Ελέγχου Συμπεριφοράς για Δίκτυα Ομοτίμων

Νικόλαος Ντάρμος

Α.Μ.: 314

Πανεπιστήμιο Πατρών, 2004

Περίληψη

Η αρχιτεκτονική δόμησης ενός κατανεμημένου δικτύου υπολογιστών ως δίκτυο ομοτίμων (peer-to-peer – P2P) έχει πλέον γίνει ο προτεινόμενος τρόπος σχεδίασης κατανεμημένων εφαρμογών. Πρώτα στην προτίμηση των χρηστών του διαδικτύου βρίσκονται τα δίκτυα ομοτίμων που χρησιμοποιούνται για τον διαμοιρασμό αρχείων, καθώς και τα δίκτυα κατανεμημένης επίλυσης προβλημάτων.

Ωστόσο, η ευρεία αποδοχή των συστημάτων αυτών από τους τελικούς χρήστες, δημιουργεί νέα προβλήματα όσον αφορά την διαχείριση των διαμοιραζόμενων αρχείων/πόρων. Στα δίκτυα ομοτίμων, οι χρήστες διαχειρίζονται μόνο τους πόρους του κόμβου τους, οι οποίοι αποτελούν πολύ μικρό μέρος του συνόλου των πόρων του δικτύου. Έτσι, για την αποδοτική λειτουργία του συστήματος, απαιτείται οι κόμβοι να είναι συνδεδεμένοι και να συνεργάζονται, συμβάλλοντας στην επίτευξη της υψηλής διαθεσιμότητας και της συνολικής αποδοτικότητας του δικτύου. Ο μεγάλος πιθανός αριθμός συμμετεχόντων, η μεγάλης κλίμακας αποκεντρικοποιημένη λειτουργία και διανομή δεδομένων και πόρων, καθώς και η αυτονομία των κόμβων ενός τέτοιου δικτύου, καθιστούν το έργο της διαχείρισης των δεδομένων, των πόρων και των προσβάσεων σε αυτά,

ιδιαίτερα επίπονη αλλά και ερευνητικά ενδιαφέρουσα διαδικασία.

Οι πρώτες P2P αρχιτεκτονικές (αλλά και πολλές σύγχρονες), βασίζονται σε μία «αλτρουιστική» μοντελοποίηση της λειτουργίας του δικτύου, σύμφωνα με την οποία υποτίθεται ότι οι κόμβοι του δικτύου είναι πρόθυμοι να μοιραστούν δεδομένα και πόρους με την υπόλοιπη κοινότητα. Σε ένα τέτοιο συνεργατικό περιβάλλον τα προαναφερθέντα προβλήματα απλοποιούνται υπερβολικά. Ωστόσο, αναλύσεις των προτύπων συμπεριφοράς των χρηστών τέτοιου είδους δικτύων, έχουν οδηγήσει στο συμπέρασμα ότι, στον κόσμο των δικτύων ομοτίμων, η «ιδιοτελής» συμπεριφορά είναι ο κανόνας και όχι η εξαίρεση. Όπως περιγράφεται και στη σχετική βιβλιογραφία, η μεγάλης κλίμακας ιδιοτελής συμπεριφορά μπορεί να οδηγήσει ένα δικτυοκεντρικό σύστημα, όπως τα δίκτυα ομοτίμων διαμοιρασμού αρχείων, σε σημείο εκ των έσω κατάρρευσης: όσο ισχυροί και αν είναι οι αλτρουιστές κόμβοι, έχουν να αντιμετωπίσουν και να εξυπηρετήσουν έναν ολοένα αυξανόμενο αριθμό ιδιοτελών κόμβων. Είναι προφανές ότι ένα τέτοιο δίκτυο καταλύει την βασικότερη σχεδιαστική αρχή των δικτύων ομοτίμων – την ομοτιμία των συμμετεχόντων – και μετατρέπει το δίκτυο σε κλασσικό σύστημα Πελάτη-Εξυπηρετητή, με λίγους κόμβους στο «κέντρο» να εξυπηρετούν υπερβολικά υπεράριθμους πελάτες στα «άκρα» του δικτύου.

Στην εργασία αυτή παρουσιάζουμε το σύστημα SeAl – ένα σύστημα παρακολούθησης και ελέγχου συμπεριφοράς για δίκτυα ομοτίμων (P2P) – με δυνατότητα διαφανούς ενσωμάτωσης τόσο σε δομημένα όσο και σε αδόμητα δίκτυα ομοτίμων. Σκοπός μας είναι η χρησιμοποίηση όσο το δυνατόν περισσοτέρων συνολικά πόρων (υπολογιστικής ισχύος, αποθηκευτικής δυνατότητας, εύρους ζώνης δικτύου) των συμετεχόντων του δικτύου, με τρόπο όσο το δυνατόν πιο δίκαιο για τους κόμβους.

Το σύστημα SeAl ελέγχει και διαχειρίζεται τα δεδομένα και τους πόρους που διατίθενται σε ένα δίκτυο ομοτίμων και τις προσβάσεις σε αυτά, (i) με τη συλλογή, αποθήκευση και διαχείριση πληροφορίας σχετικά με την συμμετοχή και την προσφορά στην κοινότητα των κόμβων του δικτύου ομοτίμων, μέσω ενός αποκεντρικοποιημένου,

ψευδωνυμικού μηχανισμού επίβλεψης και καταγραφής, και (ii) χρησιμοποιώντας κρυπτογραφικές τεχνικές για να εγγυηθεί την ορθότητα της λειτουργίας του συστήματος σε εχθρικά περιβάλλοντα, και την ταυτοποίηση και τον εντοπισμό «ιδιοτελών» κόμβων με τρόπο ανθεκτικό σε ψευδόμενους ή συνεργαζόμενους κόμβους.

Στα πλαίσια της εργασίας αυτής δίνονται ορισμοί και μετρικές για τον διαχωρισμό και την κατηγοριοποίηση των κόμβων ενός δικτύου ομοτίμων, ανάλογα με τα πρότυπα συμπεριφοράς που αυτοί επιδεικνύουν, και υλοποιούνται νέα πρωτόκολλα και μηχανισμοί για να ταυτοποιούνται οι «ιδιοτελείς» κόμβοι (free-loaders/free-riders), και να παρέχονται κίνητρα στους χρήστες ώστε να συμπεριφέρονται αλτρουιστικά όσον αφορά το περιεχόμενο και τους υπολογιστικούς/δικτυακούς πόρους που διαμοιράζονται με την υπόλοιπη κοινότητα του P2P δικτύου.

Επίσης, παρουσιάζουμε αποτελέσματα εκτενών προσομοιώσεων του συστήματος SeAl για δίκτυα ομοτίμων τυπικού μεγέθους και δυναμικών προτύπων συμπεριφοράς, σύμφωνα με τα οποία το SeAl επιτυγχάνει τους στόχους του γρήγορα, ενώ οι επιπλέον απαιτήσεις σε αποθηκευτικό χώρο, υπολογιστική δύναμη και διασύνδεση με το δίκτυο, καθώς και η όποια αύξηση στον απαιτούμενο χρόνο απόκρισης, που εισαγάγει η χρησιμοποίηση των μηχανισμών επίβλεψης και καταγραφής του SeAl, είναι πολύ μικρές.

Περιεχόμενα

Περίληψη **iii**

Κατάλογος Πινάκων **ix**

Κατάλογος Σχημάτων **x**

Κατάλογος Αλγορίθμων **xi**

Κεφάλαιο 1 Εισαγωγή **1**

1.1 Δίκτυα ομοτίμων 1

1.1.1 Δομημένα δίκτυα ομοτίμων 3

1.1.2 Αδόμητα δίκτυα ομοτίμων 5

1.2 Μοντελοποίηση των κόμβων ενός δικτύου ομοτίμων 6

1.2.1 Ιδιοτελείς κομβοι 7

1.2.2 Κακόβουλοι κόμβοι 10

1.2.3 Μεταβλητότητα των προτύπων συμπεριφοράς 11

1.3 Ορισμός του προβλήματος 11

1.4 Το σύστημα SeAl 14

1.5 Σημειολογία 15

1.6 Δομή της διατριβής 16

Κεφάλαιο 2 Το σύστημα SeAl **17**

2.1 «Χάρες» 18

2.1.1 Ορισμός ιδιοτέλειας/αλτρουισμού 18

2.1.2 VAT – το Διάνυσμα Αλτρουισμού 19

2.2 Βασική υποδομή 20

2.2.1 Δίαυλος επικοινωνίας 20

2.2.2 Διαχείριση δεδομένων καταγραφής και προσβάσεων 21

2.2.3 Ζεύγη κλειδιών 22

2.2.4 Ταυτοποίηση πόρων και συναλλαγών 23

2.2.5 Ταυτοποίηση κόμβων 23

2.2.6 Αντίγραφα 25

Κεφάλαιο 3 SAL: Ο μηχανισμός επίβλεψης **26**

3.1 Αποδείξεις συναλλαγών και «χάρες» 26

3.2 Αποπληρωμή «χαρών» (και επιβολή της) 28

3.3 Κακή φήμη – «Μαρτυρίες Κατηγορίας» 31

3.4 Οι «χάρες» στην πράξη – «Μαρτυρίες Προσφοράς» 32

3.5 Αποπληρωμή χρεών 35

Κεφάλαιο 4 SVL: Ο μηχανισμός επαλήθευσης **36**

4.1 SVL και Αποδείξεις Συναλλαγών 36

4.2 SVL και Μαρτυρίες Κατηγορίας 37

4.3 SVL και Μαρτυρίες Προσφοράς 38

4.4 Πρωτόκολλο μεταφοράς αρχείων 40

4.5 Το σύστημα SeAl στην πράξη 42

Κεφάλαιο 5 Προσομοιώσεις και αποτελέσματα μετρήσεων **46**

5.1 Περιβάλλον προσομοίωσης 46

5.2 Αποτελέσματα 50

5.2.1 Ευστάθεια 50
5.2.2 Επιπλέον κόστος 54

Κεφάλαιο 6 Σχετικές εργασίες **60**

Κεφάλαιο 7 Συμπεράσματα και μελλοντικές επεκτάσεις **65**

Βιβλιογραφία **68**

Κατάλογος Πινάκων

1.1 Βασική σημειολογία 15

2.1 Βασικές παράμετροι συμμετεχόντων κόμβων 25

5.1 Βασικές παράμετροι προσομοίωσης 51

Κατάλογος Σχημάτων

5.1 Μεταβολή μέση τιμής και διασποράς του επιπέδου αλτρουισμού των κόμβων 52
5.2 Τετραγωνικοί συντελεστές απόκλισης του μέσου επιπέδου αλτρουισμού των κόμβων 53
5.3 $|Fd|$ προς $|Fo|$ (90% ιδιοτελείς κόμβοι) 54
5.4 Επιπλέον κόστη σε εύρος ζώνης δικτύου και αποθηκευτικό χώρο από τη λειτουργία του SeAl 55
5.5 Αποδοτικότητα του μηχανισμού επίβλεψης/καταγραφής 56
5.6 Επιπλέον κόστος χρόνου 57
5.7 Χρόνοι ανακατεύθυνσης, αναμονής, και απόκρισης 58

Κατάλογος Αλγορίθμων

1 Αλγόριθμος ανακατεύθυνσης εισερχόμενης αίτησης 30
2 Αλγόριθμος δημοσίευσης Μαρτυρίας Κατηγορίας 32
3 Αλγόριθμος δημιουργίας μηνύματος αίτησης προσπέλασης 33
4 Αλγόριθμος προεπεξεργασίας εισερχόμενης αίτησης 34
5 Αλγόριθμος μεταφοράς αρχείων 41

Κεφάλαιο 1

Εισαγωγή

Η αρχιτεκτονική δόμησης ενός κατανεμημένου δικτύου υπολογιστών ως δίκτυο ομοτίμων (peer-to-peer – P2P) έχει πλέον γίνει ο προτιμώμενος/προτεινόμενος τρόπος σχεδίασης κατανεμημένων εφαρμογών. Πρόκειται για μία τάξη συστημάτων που χρησιμοποιούν και διαχειρίζονται πόρους (αποθηκευτικό χώρο, επεξεργαστικούς κύκλους, περιεχόμενο, κτλ.) τα οποία βρίσκονται στα «άκρα» του Διαδικτύου (στους προσωπικούς υπολογιστές των τελικών χρηστών), σε αντίθεση με τις κλασσικές αρχιτεκτονικές Πελάτη-Εξυπηρετητή (client-server) οι οποίες χρησιμοποιούν πόρους τοποθετημένους στο «κέντρο» του Διαδικτύου (πόρους των εξυπηρετητών).

1.1 Δίκτυα ομοτίμων

Καθώς η πρόσβαση στους κατανεμημένους αυτούς πόρους συνεπάγεται τη λειτουργία σε ένα περιβάλλον ασταθούς συνδεσιμότητας και υψηλής δυναμικότητας της παρουσίας των κόμβων και των πόρων, οι κόμβοι του P2P δικτύου πρέπει να λειτουργούν έξω και πέρα από τις κλασσικές μεθόδους εντοπισμού και διασύνδεσης (π.χ. ονοματολογία μέσω DNS εξυπηρετητών, προώθηση μηνυμάτων βάσει αποκλειστικά διευθύνσεων IP, κτλ.) και να χαρακτηρίζονται από σημαντική ή ολική αυτονομία από

τους κεντρικούς εξυπηρετητές του Διαδικτύου. Οι βασικές σχεδιαστικές αρχές των δικτύων αυτών συνοψίζονται στα ακόλουθα:

1. Οι κόμβοι του δικτύου ομοτίμων πρέπει να είναι αυτόνομοι και ανεξάρτητοι. Οι διαχειριστές των κόμβων πρέπει να μπορούν να ορίσουν αυτόνομα της προτιμήσεις τους όσον αφορά τις παραμέτρους λειτουργίας του κόμβου τους (π.χ. διαμοιραζόμενοι πόροι) και το δίκτυο ομοτίμων να προσαρμόζεται αυτόματα στης ρυθμίσεις αυτές.

2. Το δίκτυο πρέπει να λειτουργεί αποδοτικά χωρίς την ύπαρξη κάποιας κεντρικοποιημένης οντότητας (π.χ. κεντρικού εξυπηρετητή), καθώς αυτή μπορεί να αποτελέσει «μεμονωμένο σημείο αποτυχίας» (single point of failure).

3. Οι αλγόριθμοι που εκτελούνται στο δίκτυο αυτό πρέπει να βασίζονται σε τοπικές αποφάσεις ή σε στοιχεία/δεδομένα προσβάσιμα από όλους τους κόμβους.

4. Το δίκτυο πρέπει να μπορεί να κλιμακώνεται σε πολύ μεγάλους αριθμούς κόμβων (της τάξης των δεκάδων εκατομμυρίων) και διαμοιραζόμενων αρχείων/πόρων (της τάξης των εκατοντάδων εκατομμυρίων).

5. Το δίκτυο πρέπει να είναι ανεκτικό σε λάθη και να επιτρέπει μέγιστη δυναμικότητα (είσοδο, έξοδο, μεταφορά, μετατροπή, κτλ.), τόσο στο σύνολο των κόμβων, όσο και στο σύνολο των διαμοιραζόμενων πόρων.

Στα δίκτυα ομοτίμων κάθε οντότητα (κόμβος, αντικείμενο ή συναλλαγή) ταυτοποιείται από έναν αριθμό (το *ID* του). Τα δίκτυα ομοτίμων χωρίζονται σε «δομημένα» και «αδόμητα», ανάλογα με τον τρόπο που διασυνδέονται οι κόμβοι τους και την μέθοδο που ακολουθείται στην επιλογή του ή των κόμβων που χρησιμοποιούνται κατά την δρομολόγηση πακέτων σε αυτά.

1.1.1 Δομημένα δίκτυα ομοτίμων

Στα δομημένα δίκτυα, οι κόμβοι του δικτύου οργανώνονται αυτόματα σε συγκεκριμένες δομές με χαρακτηριστικά και ιδιότητες που βοηθούν στην αναζήτηση πληροφορίας και στην δρομολόγηση πακέτων με εγγυήσεις στον αριθμό των μεταβάσεων από κόμβο σε κόμβο, στον συνολικό χρόνο απόκρισης, ή στον φόρτο που ανατίθεται σε κάθε κόμβο. Τα δίκτυα αυτά έχουν δύο κύρια χαρακτηριστικά, τα οποία αποτελούν και τις ειδοποιούς διαφορές ανάμεσά τους[35]: (i) τον τύπο της δομής ή αλλιώς την τοπολογία την οποία χτίζουν και διατηρούν, και (ii) την συνάρτηση με την οποία καθορίζουν την απόσταση δύο οποιονδήποτε αντικειμένων στον χώρο διευθυνσιοδότησής τους (και κατ΄επέκταση και τον επόμενο κόμβο κατά την δρομολόγηση μηνυμάτων). Έτσι έχουμε:

- Δίκτυα με τους κόμβους κατανεμημένους σε δακτύλιο. Στα δίκτυα αυτά οι συμμετέχοντες κόμβοι διατάσσονται σε έναν δακτύλιο κατά αύξουσα σειρά βάσει του ID τους και κάθε κόμβος διατηρεί συνδέσμους σε έναν προκαθορισμένο αριθμό επόμενων (και προαιρετικά προηγούμενων) κόμβων στον δακτύλιο. Η απόσταση δύο κλειδιών καθορίζεται από την αριθμητική διαφορά των κλειδιών αυτών, με αποτέλεσμα κάθε μήνυμα να μεταδίδεται προς μία φορά του δακτυλίου (π.χ. αντίθετα από τη φορά κίνησης των δεικτών του ρολογιού), από κάθε κόμβο προς τον πλέον απομακρυσμένο «γείτονά» του, του οποίου το ID είναι μικρότερο αριθμητικά από το ID του τελικού παραλήπτη. Τέτοιο σύστημα είναι το Chord[39], όταν χρησιμοποιούνται μόνο οι σύνδεσμοι προς γειτονικούς κόμβους (successor/predecessor pointers).

- Δίκτυα βασισμένα σε δέντρα ή πλέγματα. Τα πλέγματα αυτά συνήθως είναι κατανεμημένες δεντρικές δομές, στις οποίες κάθε κόμβος του δικτύου είναι η ρίζα και ενός δέντρου. Εν ολίγοις, κάθε κόμβος ενός τέτοιου πλέγματος διατηρεί συνδέσμους προς έναν αριθμό κόμβων λογαριθμικό ως προς το συνολικό πλή-

θος κόμβων στο σύστημα, και δρομολογεί μηνύματα προς τον κόμβο εκείνο που βρίσκεται πιο κοντά (με βάση την χρησιμοποιούμενη μετρική απόστασης) στον τελικό αποδέκτη. Κλασσικά παραδείγματα τέτοιων δικτύων είναι τα συστήματα Tapestry[81] και Pastry[22], τα οποία βασίζονται σε πλέγματα Plaxton[60] – κάθε κόμβος διατηρεί συνδέσμους προς κόμβους των οποίων το ID διαφέρει από το δικό του κατά ένα ψηφίο (bit) και η δρομολόγηση γίνεται διορθώνοντας ένα ψηφίο κάθε φορά – και το σύστημα Kademlia[50], το οποίο βασίζεται σε μία παρόμοια πλεγματική δομή αλλά στο οποίο η απόσταση δύο κλειδιών καθορίζεται από το Αποκλειστικό-Η (XOR) των ID τους.

- Δίκτυα βασισμένα σε υπερκυβικές ή γενικά πολυδιάστατες δομές. Τα δίκτυα αυτά μοντελοποιούν τον χώρο των διευθύνσεων των κόμβων τους ως έναν πολυδιάστατο καρτεσιανό χώρο, με τμήματα των ID των κόμβων να αποτελούν τις συντεταγμένες του κόμβου σε κάθε διάσταση του χώρου. Οι κόμβοι διατηρούν συνδέσμους προς τους «γειτονικούς» τους κόμβους σε κάθε διάσταση και η δρομολόγηση γίνεται με βάση την Ευκλείδια απόσταση ανάμεσα σε έναν κόμβο και στον τελικό προορισμό του δρομολογούμενου μηνύματος. Στην κατηγορία αυτή εμπίπτουν (υπό όρους) τα συστήματα CAN[61] και Chord[39] (όταν χρησιμοποιούνται οι σύνδεσμοι προς κόμβους σε εκθετικά αυξανόμενες αποστάσεις).

- Δίκτυα βασισμένα σε δίκτυα πεταλούδων, όπως το σύστημα Viceroy[49]. Τα δίκτυα αυτά έχουν καλύτερες θεωρητικές ιδιότητες και παρέχουν καλύτερες εγγυήσεις λειτουργίας από τα παραπάνω αλλά δεν έχουν τύχει ευρείας αποδοχής, κυρίως λόγω της πολυπλοκότητας στην μοντελοποίηση και υλοποίησή τους.

Τα δομημένα δίκτυα προσφέρουν ισχυρές (στατιστικές) εγγυήσεις όσον αφορά (i) τον αριθμό των κόμβων που θα εμπλακούν κατά την δρομολόγηση ενός μηνύματος – συνήθως λογαριθμικός ως προς τον συνολικό αριθμό των κόμβων στο σύστημα – και (ii) τον βαθμό (πλήθος συνδέσμων) των κόμβων – συνήθως λογαριθμικός ή

ανεξάρτητος (στην περίπτωση του Viceroy) ως προς τον συνολικό αριθμό των κόμβων στο σύστημα. Καθώς οι τελεστές που παρέχουν τα δίκτυα αυτά μοιάζουν αρκετά με τους τελεστές των πινάκων κατακερματισμού (hash tables), συνήθως αναφέρονται στη σχετική βιβλιογραφία ως «υποστρώματα κατανεμημένων πινάκων κατακερματισμού» (Distributed Hash Table – DHT – overlays).

1.1.2 Αδόμητα δίκτυα ομοτίμων

Στα αδόμητα δίκτυα ομοτίμων κάθε κόμβος διατηρεί συνδέσμους προς έναν αριθμό τυχαίων άλλων κόμβων του δικτύου. Το σύνολο των συνδέσμων εμπλουτίζεται με την πάροδο του χρόνου, μέσω των συναλλαγών που κάθε κόμβος έχει με το υπόλοιπο δίκτυο, και διαχειρίζεται είτε χρησιμοποιώντας κλασσικές μεθόδους αντικατάστασης (π.χ. Least Recently Used (LRU) ή Least Frequently Used (LFU)), είτε μετρικές βασισμένες σε χαρακτηριστικά του δικτύου (π.χ. απόσταση δρομολόγησης από κόμβο σε κόμβο, χρονική καθυστέρηση μετάδοσης, κτλ.), είτε κάποια ιβρύδιο των παραπάνω.

Ως αποτέλεσμα αυτής ακριβώς της μεθόδου απόκτησης και διαχείρισης των συνδέσμων κάθε κόμβου, τα δίκτυα αυτά παρουσιάζουν φαινόμενα «Μικρού Κόσμου» (small-world effects)[54]· η πιθανότητα να έχει ένας κόμβος βαθμό λ είναι μία φθίνουσα εκθετική συνάρτηση στο λ (γνωστή και ως Power-Law), της μορφής λ^c (με c μια αρνητική σταθερά η τιμή της οποίας εξαρτάται από τα χαρακτηριστικά του δικτύου). Σε ένα τέτοιο δίκτυο υπάρχουν πολλοί κόμβοι με μικρούς βαθμούς (λίγους συνδέσμους) και λίγοι κόμβοι με μεγάλους βαθμούς (πολλούς συνδέσμους). Το φαινόμενο «μικρού κόσμου» και η στατιστική συμπεριφορά τύπου Zipf, Pareto, ή Power-Law θεωρούνται χαρακτηριστικά της ανθρώπινης φύσης· «μικροί κόσμοι» παρατηρήθηκαν και μοντελοποιήθηκαν για πρώτη φορά στον πραγματικό κόσμο.

Οι δομές αυτές παρουσιάζουν πολλά ενδιαφέροντα χαρακτηριστικά, όπως το γνωστό αποτέλεσμα των «7 συνδέσμων»[54], χαρακτηριστικά ανεκτικότητας σε λάθη

και κλιμάκωσης ως κάποιου σημείου[47], κτλ. Ωστόσο, το κυριότερο πρόβλημα των συστημάτων αυτού του τύπου είναι το μεγάλο κόστος λειτουργίας τους όσον αφορά τους απαιτούμενους δικτυακούς πόρους· η αναζήτηση και δρομολόγηση μηνυμάτων σε ένα τέτοιο δίκτυο γίνεται χρησιμοποιώντας τεχνικές «πλημμυρισμού» (flooding) ή μετάδοσης-προς-όλους (broadcast), με αποτέλεσμα το πλήθος των κόμβων που εμπλέκονται σε κάθε βήμα να αυξάνει εκθετικά.

Κατά καιρούς έχουν προταθεί διάφορες λύσεις για το πρόβλημα αυτό, οι κυριώτερες των οποίων είναι τα υβριδικά δίκτυα ([78, 80]) και η υλοποίηση αδόμητων δικτύων πάνω από δομημένα ([12]). Κλασσικοί αντιπρόσωποι αδόμητων δικτύων ομοτίμων είναι τα συστήματα Gnutella[30] και Mojonation[55], καθώς και τα συστήματα που βασίζονται στο πρωτόκολλο FastTrack[25] (π.χ. Kazaa[43], Grokster[34] και Morpheus[56]).

1.2 Μοντελοποίηση των κόμβων ενός δικτύου ομοτίμων

Στην ενότητα αυτή θα επιχειρήσουμε μία περισσότερο ή λιγότερο πρακτική μοντελοποίηση και κατηγοριοποίηση των προτύπων συμπεριφοράς των χρηστών ενός δίκτυου ομοτίμων, διαχωρίζοντας τους «ιδιοτελείς» χρήστες από τους «κακόβουλους». Αυτή η ενότητα θα αναδείξει περαιτέρω την δυσκολία του έργου της αντιμετώπισης ιδιοτελών συμπεριφορών σε ένα ευρέως κατανεμημένο σύστημα, όπως τα σύγχρονα δίκτυα ομοτίμων.

Στα πλαίσια της μοντελοποίησης αυτής, ορίζουμε ως *ιδιοτελή* την συμπεριφορά του κόμβου εκείνου ο οποίος αρνείται, μερικά ή ολικά, να μοιραστεί τους πόρους του (αρχεία, επεξεργαστική ισχύς, εύρος ζώνης δικτύου, κτλ.) με τους υπόλοιπους κόμβους του δικτύου ή προσπαθεί συστηματικά να αποκτήσει πρόσβαση σε περισσότερους πόρους από αυτούς που του αναλογούν βάσει της συνεισφοράς του στο δίκτυο ομοτίμων. Αναλυτικά μοντέλα τέτοιων συμπεριφορών έχουν εξετασθεί σε βάθος στο πεδίο των κοινωνικών επιστημών[29, 36, 38]. Η περιγραφή τους ξεφεύγει από τα όρια

της εργασίας αυτής· περισσότερες πληροφορίες υπάρχουν στην σχετική βιβλιογραφία.

Από την άλλη, ορίζουμε ως *κακόβουλη* την συμπεριφορά του κόμβου εκείνου ο οποίος προσπαθεί συγκεκριμένα να προκαλέσει βλάβη στην ορθή λειτουργία του συστήματος. Συμπεριφορές αυτού του τύπου συνήθως εκφράζονται είτε μέσω επιθέσεων στα διάφορα τμήματα του συστήματος, είτε μέσω σκόπιμης παρεκτροπής από τα πρωτόκολλα και τις πολιτικές του συστήματος.

1.2.1 Ιδιοτελείς κομβοι

Ακολουθώντας το πνεύμα του παραπάνω ορισμού της ιδιοτελούς συμπεριφοράς, διακρίνουμε δύο κύριες κατηγορίες ιδιοτελών χρηστών: (i) τους «τσιγγούνηδες» (no-givers), και (ii) τους «πλεονέκτες» (all-takers).

Οι «τσιγγούνηδες»

Οι «τσιγγούνηδες» είναι χρήστες οι οποίοι αρνούνται να μοιραστούν τους πόρους τους με την υπόλοιπη κοινότητα ομοτίμων. Η άρνηση αυτή μπορεί να είναι είτε μερική («μερικώς τσιγγούνηδες») είτε ολική («πλήρως τσιγγούνηδες»). Ως μερική άρνηση ορίζεται η κατάσταση στην οποία ένας κόμβος/χρήστης είτε συνεισφέρει μόνο ένα τμήμα των πόρων ή του περιεχομένου του, είτε δηλώνει ότι διαθέτει λιγότερους πόρους/περιεχόμενο απ΄ότι κατέχει πραγματικά, πιθανότατα σε μία προσπάθεια να αποθαρρύνει άλλους κόμβους από το να τον επιλέξουν ως εξυπηρετητή/προωθητή μηνυμάτων, ανάλογα και με την εσωτερική λειτουργία του υπό εξέταση δικτύου ομοτίμων. Ως ολική άρνηση ορίζεται η κατάσταση στην οποία ένας κόμβος/χρήστης δε συνεισφέρει καθόλου πόρους/περιεχόμενο στην κοινότητα ομοτίμων στην οποία ανήκει.

Οι free-riders είναι κλασσικό παράδειγμα («πλήρως τσιγγούνηδων»). Από την άλλη, οι («μερικώς τσιγγούνηδες») απαντούνται συχνά σε δίκτυα ομοτίμων διαμοιρασμού αρχείων τα οποία βασίζονται στο πρωτόκολλο FastTrack[25], καθώς και σε πολλά σύγχρονα δίκτυα ομοτίμων διαμοιρασμού αρχείων στα οποία υπάρχει πρόβλεψη για

αποθήκευση και κατανομή μεταπληροφορίας σχετικά με τις δυνατότητες των συμμετεχόντων κόμβων.

Όπως αναφέρεται και στο [6], ο αυξανόμενος αριθμός των free-riders μπορεί να οδηγήσει ένα δίκτυο ομοτίμων μεγάλων διαστάσεων στο σημείο της κατάρρευσης· οι αλτρουιστικοί κόμβοι καταλήγουν τελικά να λαμβάνουν ένα ολοένα αυξανόμενο πλήθος ερωτημάτων και αιτήσεων πρόσβασης, καταναλώνοντας έτσι τους υπολογιστικούς πόρους τους και το διαθέσιμο εύρος ζώνης δικτύου τους. Οι διαχειριστές αυτών των κόμβων έχουν τότε τρεις επιλογές: (i) να αποσυνδέσουν τους κόμβους τους από το δίκτυο ομοτίμων, αφαιρώντας μαζί τους κι ένα σημαντικό ποσοστό των διαθέσιμων πόρων/περιεχομένου, (ii) να περιορίσουν κάπως τους πόρους/περιεχόμενο που καθιστούν προσβάσιμους από την υπόλοιπη κοινότητα, ή (iii) να ελπίζουν ότι τα συστήματά τους είναι αρκετά ισχυρά ώστε να ανταπεξέλθουν στον φόρτο του δικτύου ομοτίμων. Σε όλες τις περιπτώσεις, η ύπαρξη και η δράση των free-riders έχει ως αποτέλεσμα την υποβίβαση της απόδοσης του συστήματος ή/και της ποιότητας και ποσότητας των διαθέσιμων πόρων/περιεχομένου.

Οι «πλεονέκτες»

Από την άλλη, οι «πλεονέκτες» είναι κόμβοι οι οποίοι, υπό την προϋπόθεση της ύπαρξης ενός (έστω και στοιχειώδους) μηχανισμού επίβλεψης, προσπαθούν να αποκτήσουν πρόσβαση σε περισσότερους πόρους απ΄όσους τους αναλογούν βάσει της προσφοράς τους στην κοινότητα ομοτίμων.

Ένα από τα πρώτα παραδείγματα τέτοιας συμπεριφοράς στον χώρο τον υπολογιστών παρατηρήθηκε στις μέρες των «υπηρεσιών πίνακα ανακοινώσεων» (Bulletin-Board Services – BBS)· οι χρήστες συνδέονταν σε έναν εξυπηρετητή BBS για να αποκτήσουν πρόσβαση σε αρχεία κειμένου, αρχεία «τέχνης ascii», κτλ. Οι εξυπηρετητές ήταν εξ ορισμού ρυθμισμένοι έτσι ώστε να αναγκάζουν τους χρήστες να στείλουν πρώτα κάποιο δικό τους αρχείο πρωτού τους δώσουν πρόσβαση στα αποθηκευμένα αρχεία,

ενώ συνήθως υπήρχε και κάποιος περιορισμός στην αναλογία του όγκου δεδομένων που οι χρήστες έστελναν στον εξυπηρετητή προς τον όγκο δεδομένων που έπαιρναν από αυτόν. Παρομοίως, πριν την έλευση και την τεράστια επιτυχία των συστημάτων Napster[57] και Audiogalaxy[9], ο πλέον συνηθισμένος τρόπος απόκτησης πρόσβασης σε αρχεία μουσικής MP3 ήταν μέσω εξυπηρετητών FTP, στους οποίους έπρεπε πρώτα να στείλει κανείς έναν όγκο δεδομένων από δικά του αρχεία, ώστε να αποκτήσει αρκετά «τεκμήρια» (tokens), πρωτού του δοθεί το δικαίωμα να «κατεβάσει» από τα ήδη αποθηκευμένα αρχεία (και πάλι υπό τον περιορισμό κάποιας αναλογίας στους όγκους δεδομένων που ο χρήστης έστελνε και λάμβανε).

Ένα κοινό πρόβλημα των τεχνικών αυτών ήταν ότι οι χρήστες μπορούσαν να στείλουν χαμηλής ποιότητας ή τελείως άχρηστα αρχεία, ως αντάλλαγμα σε υψηλής ποιότητας αρχεία MP3 ή τέχνης ascii. Ο διαχειριστής του εξυπηρετητή θα καταλάβαινε εν τέλει ότι το υλικό που έλαβε δεν είχε καμία αξία. Ωστόσο, όταν αυτό θα γινόταν τελικά, ο αρχικός χρήστης θα είχε πιθανότατα ολοκληρώσει τις μεταφορές αρχείων και θα είχε αποσυνδεθεί, έχοντας αποκτήσει πρόσβαση σε πόρους/περιεχόμενο που δεν του αναλογούσε βάσει της προσφοράς του.

Σ΄αυτό το γεγονός βρίσκεται ο βασικός παράγοντας δυσκολίας στην αντιμετώπιση τέτοιου είδους χρηστών· προφανώς το σύστημα δε μπορεί να διακρίνει αυτόματα αν ένα συγκεκριμένο αρχείο είναι κάποιας αξίας ή τελείως άχρηστο, ενώ η ανθρώπινη παρέμβαση συνήθως γίνεται σε υπερβολικά προχωρημένο στάδιο της όλης διαδικασίας ώστε να αντιμετωπισθεί τέτοιου είδους συμπεριφορά.

Τα συστήματα που βασίζονται στο πρωτόκολλο FastTrack[25] έκαναν ένα βήμα παραπέρα, ενσωματώνοντας στο P2P λογισμικό τους υποστήριξη για «βαθμολόγηση» (scoring) της συνεισφοράς των χρηστών τους. Ωστόσο η σχεδίασή τους υπέφερε από ένα βασικό πρόβλημα: η βαθμολογία κάθε κόμβου αποθηκευόταν τοπικά στον κόμβο, με μοναδικό μέσο προστασίας την μη αποκάλυψη του πηγαίου κώδικα του λογισμικού. Έτσι, εμφανίστηκε το λογισμικό K-Lite, με την παράμετρο «*K-Lite Master*», η

οποία έθετε εξ ορισμού την βαθμολογία του κόμβου στην μέγιστη δυνατή τιμή της (γνωστή με την κωδική ονομασία «Υπέρτατο Ον» – «Supreme Being»), με αποτέλεσμα την κατάργηση των όποιων εγγυήσεων του μηχανισμού βαθμολόγησης.

1.2.2 Κακόβουλοι κόμβοι

Με τον όρο «κακόβουλος» κόμβος ή χρήστης ορίζουμε τον κόμβο ή χρήστη εκείνον ο οποίος προσπαθεί ηθελημένα και συγκεκριμένα να βλάψει τη λειτουργία του συστήματος. Η συμπεριφορά αυτή συνήθως εκφράζεται μέσω επιθέσεων στα δομικά συστατικά του συστήματος ή μέσω παράκαμψης/μη συμμόρφωσης με τα πρωτόκολλα και τις πολιτικές του συστήματος. Οι επιθέσεις μπορεί να γίνονται προς:

1. συγκεκριμένους κόμβους του δικτύου (π.χ. κόμβους μεγάλου βαθμού σε αδόμητα δίκτυα ομοτίμων),
2. το υπόστρωμα δρομολόγησης/αναζήτησης (routing/searching layer),
3. το υπόστρωμα αποθήκευσης/ανάκτησης πληροφορίας (information storage/retrieval layer),
4. το υπόστρωμα ταυτοποίησης (authentication layer), ή
5. το υπόστρωμα επίβλεψης/καταγραφής (auditing/accounting layer), όταν υπάρχει κάτι τέτοιο.

Επιπλέον, διακρίνουμε τις επιθέσεις σε δύο κύριες κατηγορίες:

1. παθητικές επιθέσεις, στις οποίες ο επιτιθέμενος απλά παρακολουθεί το σύστημα και άρα απειλεί μόνο την ανωνυμία των χρηστών και την μυστικότητα των δεδομένων, και
2. ενεργητικές επιθέσεις, στις οποίες ο επιτιθέμενος προσπαθεί να διαγράψει, να προσθέσει, ή εν γένει να μεταβάλει τη μεταδιδόμενη πληροφορία και το διαθέσι-

μο περιεχόμενο του συστήματος, και άρα απειλεί επιπλέον τόσο την ακεραιότητα των δεδομένων όσο και την ορθή ταυτοποίηση των συμμετεχόντων.

Περισσότερα για τις επιθέσεις αυτές θα αναφερθούν και σε επόμενο κεφάλαιο, στο οποίο θα αναλυθεί και ο τρόπος με τον οποίο το σύστημα SeAl αντιδρά σε αυτές.

1.2.3 Μεταβλητότητα των προτύπων συμπεριφοράς

Δυστυχώς ή ευτυχώς, ένας ιδιοτελής ή κακόβουλος χρήστης/κόμβος δεν συμπεριφέρεται διαρκώς με τον ίδιο τρόπο. Διακρίνουμε δύο είδη μεταβλητότητας στα πρότυπα συμπεριφοράς των χρηστών:

1. Χρονική μεταβλητότητα – ο χρήστης/κόμβος παρουσιάζει διαφορετικά πρότυπα συμπεριφοράς σε διαφορετικές χρονικές στιγμές, και

2. Χωρική μεταβλητότητα – ο χρήστης/κόμβος παρουσιάζει διαφορετικά πρότυπα συμπεριφοράς σε διαφορετικούς χρήστες/κόμβους ή για διαφορετικούς πόρους/αιτήσεις.

Για να αντιμετωπισθούν αστάθειες αυτού του είδους στην συμπεριφορά των κόμβων, απαιτείται συνεργασία ανάμεσα στους κόμβους του δικτύου ώστε να εντοπίζονται και να τιμωρούνται οι ιδιοτελείς/κακόβουλοι κόμβοι (για περιπτώσεις χωρικής μεταβλητότητας), ενώ πρέπει να παρέχονται και εγγυήσεις όσον αφορά την επικαιρότητα και εγκυρότητα των σχετικών πληροφοριών (για περιπτώσεις χρονικής μεταβλητότητας).

1.3 Ορισμός του προβλήματος

Πρώτα στην προτίμηση των χρηστών του διαδικτύου βρίσκονται τα δίκτυα ομοτίμων που χρησιμοποιούνται για τον διαμοιρασμό αρχείων ([25, 30, 57]), καθώς και τα δίκτυα κατανεμημένης επίλυσης προβλημάτων ([20, 67]). Ωστόσο, η ευρεία αποδοχή

των συστημάτων αυτών από τους τελικούς χρήστες, δημιουργεί νέα προβλήματα όσον αφορά την διαχείριση των διαμοιραζόμενων αρχείων/πόρων. Στα δίκτυα ομοτίμων, οι χρήστες διαχειρίζονται μόνο τους πόρους του κόμβου τους, οι οποίοι αποτελούν πολύ μικρό μέρος του συνόλου των πόρων του δικτύου. Έτσι, για την αποδοτική λειτουργία του συστήματος, απαιτείται οι κόμβοι να είναι συνδεδεμένοι και να συνεργάζονται, συμβάλλοντας στην επίτευξη της υψηλής διαθεσιμότητας και της συνολικής αποδοτικότητας του δικτύου. Ο μεγάλος πιθανός αριθμός συμμετεχόντων, η μεγάλης κλίμακας αποκεντρικοποιημένη λειτουργία και διανομή δεδομένων και πόρων, καθώς και η αυτονομία των κόμβων ενός τέτοιου δικτύου, καθιστούν το έργο της διαχείρισης των δεδομένων, των πόρων και των προσβάσεων σε αυτά, ιδιαίτερα επίπονη αλλά και ερευνητικά ενδιαφέρουσα διαδικασία.

Οι πρώτες P2P αρχιτεκτονικές (αλλά και πολλές σύγχρονες), βασίζονται σε μία «αλτρουιστική» μοντελοποίηση της λειτουργίας του δικτύου, σύμφωνα με την οποία υποτίθεται ότι οι κόμβοι του δικτύου είναι πρόθυμοι να μοιραστούν δεδομένα και πόρους με την υπόλοιπη κοινότητα. Σε ένα τέτοιο συνεργατικό περιβάλλον τα προαναφερθέντα προβλήματα απλοποιούνται υπερβολικά. Ωστόσο, αναλύσεις των προτύπων συμπεριφοράς των χρηστών τέτοιου είδους δικτύων, έχουν οδηγήσει στο συμπέρασμα ότι, στον κόσμο των δικτύων ομοτίμων, η «ιδιοτελής» συμπεριφορά είναι ο κανόνας και όχι η εξαίρεση ([6, 66]).

Το 1968 ο Garrett Hardin, καθηγητής Ανθρώπινης Οικολογίας στο πανεπιστήμιο της Καλιφόρνια (Σάντα Μπάρμπαρα), περιέγραψε την έννοια της «Τραγωδίας των Κοινών»[36] ως την κατάσταση στην οποία κάθε άτομο σε μία κοινωνία αυτόνομων ατόμων/οντοτήτων προσπαθεί να αυξήσει απεριόριστα το ποσοστό προσωπικής του εκμετάλλευσης των πόρων με κοινή πρόσβαση από όλα τα άτομα της κοινότητας, με αποτέλεσμα την αναπόφευκτη εξάντλησή τους. Στον κόσμο των P2P δικτύων το φαινόμενο αυτό απαντάται ολοένα και περισσότερο, με την μορφή «ιδιοτελών» κόμβων (free-riders ή free-loaders), με τους κοινούς πόρους να αποτελούνται από επεξεργαστι-

κούς κύκλους, αποθηκευτικό χώρο, διαμοιραζόμενα αρχεία και εύρος ζώνης δικτύου.

Θεωρούμε ότι το πρόβλημα της αντιμετώπισης «ιδιοτελών» συμπεριφορών σε δίκτυα ομοτίμων διαμοιρασμού αρχείων, ιδιαίτερα στην κλίμακα μεγέθους των σημερινών δικτύων, είναι (i) υψίστης σημασίας για την απόδοση, τη σταθερότητα, και την κλιμάκωση των συστημάτων αυτών, και (ii) ιδιαίτερα ενδιαφέρουσα ερευνητικά, λόγω των περιορισμών που θέτει η ευρέως κατανεμημένη λειτουργία των συμμετεχόντων κόμβων. Όπως περιγράφεται και στη σχετική βιβλιογραφία ([6, 66]), η μεγάλης κλίμακας ιδιοτελής συμπεριφορά μπορεί να οδηγήσει ένα δικτυοκεντρικό σύστημα, όπως τα δίκτυα ομοτίμων διαμοιρασμού αρχείων, σε σημείο εκ των έσω κατάρρευσης: όσο ισχυροί και αν είναι οι αλτρουιστές κόμβοι, έχουν να αντιμετωπίσουν και να εξυπηρετήσουν έναν ολοένα αυξανόμενο αριθμό ιδιοτελών κόμβων. Είναι προφανές ότι ένα τέτοιο δίκτυο καταλύει την βασικότερη σχεδιαστική αρχή των δικτύων ομοτίμων – την ομοτιμία των συμμετεχόντων – και μετατρέπει το δίκτυο σε κλασσικό σύστημα Πελάτη-Εξυπηρετητή, με λίγους κόμβους στο «κέντρο» να εξυπηρετούν υπερβολικά υπεράριθμους πελάτες στα «άκρα» του δικτύου.

Όπως προαναφέρθηκε, η αντιμετώπιση του προβλήματος αυτού σε ένα δίκτυο ομοτίμων αποτελεί πρόκληση. Πρώτον επειδή, λόγω της αυτονομίας των κόμβων και την φυσική υψηλή δυναμικότητα τέτοιων συστημάτων, όλοι οι αλγόριθμοι και οι τεχνικές που θα εφαρμοστούν πρέπει να βασίζονται σε καθαρά τοπικές αποφάσεις σε κάθε κόμβο του δικτύου. Επιπλέον, δεν υπάρχει κάποια κεντρικοποιημένη «αποθήκη» πληροφοριών· όλη η διαθέσιμη πληροφορία βρίσκεται κατακερματισμένη και αποθηκευμένη στους διάφορους κόμβους που αποτελούν το P2P δίκτυο. Τέλος, η κατανάλωση δικτυακών πόρων (εύρους ζώνης του δικτύου) είναι συνήθως ο καθοριστικότερος παράγοντας της απόδοσης αλλά και της αποδοχής από τους χρήστες των συστημάτων P2P· παρά τα τεράστια ποσά πληροφορίας που είναι διαθέσιμα σε ένα δίκτυο ομοτίμων, συνήθως αναγκαζόμαστε είτε να περιορίσουμε το ποσοστό της πληροφορίας αυτής που είναι προσβάσιμη από κάποιον κόμβο (θέτοντας π.χ. ένα άνω όριο στο σύνολο των

κόμβων που είναι προσβάσιμοι από τον κόμβο αυτόν), είτε προσπαθώντας να εντοπίσουμε και να μεταφέρουμε τμήματα της πληροφορίας αυτής χρησιμοποιώντας όσο το δυνατόν λιγότερες μεταβάσεις μέσω συνδέσμων του δικτύου (network hops).

1.4 Το σύστημα SeAl

Παρουσιάζουμε το σύστημα SeAl: ένα υπόστρωμα λογισμικού με δυνατότητα διαφανούς ενσωμάτωσης τόσο σε δομημένα όσο και σε αδόμητα δίκτυα ομοτίμων διαμοιρασμού αρχείων.

Σε σχεδιαστικό επίπεδο, το σύστημα SeAl αποτελείται από δύο ανεξάρτητους και συνεργαζόμενους μηχανισμούς:

1. Το μηχανισμό επίβλεψης/καταγραφής (SeAl Auditing/Accounting Layer – SAL), ο οποίος είναι υπεύθυνος για την συλλογή, αποθήκευση και διαχείριση της πληροφορίας σχετικά με την συμμετοχή και την προσφορά στην κοινότητα των κόμβων του δικτύου ομοτίμων. Πρόκειται για έναν εντελώς αποκεντρικοποιημένο, ψευδωνυμικό μηχανισμό επίβλεψης και καταγραφής, ο οποίος διατηρεί και διαχειρίζεται την «φήμη» (reputation) των κόμβων του δικτύου, βασιζώμενος στην συμπεριφορά τους και στην ανάδραση των συναλλαγών μεταξύ κόμβων του δικτύου.

2. Το μηχανισμό επαλήθευσης (SeAl Verification Layer – SVL), ο οποίος χρησιμοποιεί κρυπτογραφικές τεχνικές για να εγγυηθεί την ορθότητα της λειτουργίας του συστήματος σε εχθρικά περιβάλλοντα, και την ταυτοποίηση και τον εντοπισμό «ιδιοτελών» κόμβων με τρόπο ανθεκτικό σε ψευδόμενους ή συνεργαζόμενους κόμβους.

Τα τμήματα αυτά λειτουργούν σε δύο επίπεδα: (i) δίνουν κίνητρα στους κόμβους να δρουν αλτρουιστικά, έτσι ώστε να συγκεντρώσουν θετικά σχόλια από τους

υπόλοιπους κόμβους και να αποκτήσουν καλή «φήμη» στο σύστημα, και (ii) παρέχουν στο P2P σύστημα τα κατάλληλα εργαλεία και την απαιτούμενη υποδομή ώστε να μπορεί να κατηγοριοποιήσει τους κόμβους του και να τους επιτρέπει ελεγχόμενη πρόσβαση στους πόρους του συστήματος, με βάση την συνεισφορά τους στην υπόλοιπη κοινότητα.

1.5 Σημειολογία

Σύμβολο	**Περιγραφή**
n_i	Κόμβος του δικτύου
r	Διαμοιραζόμενος πόρος του δικτύου
id	Αναγνωριστικό ταυτοποίησης
TR	Απόδειξη συναλλαγής
$r.id$	Αναγνωριστικό ταυτοποίησης του πόρου r
$r.size$	Μέγεθος του πόρου r
$f(n_1, n_2, r)$	Χάρη που χρωστά ο κόμβος n_1 στον κόμβο n_2 για προσπέλαση στον πόρο r
$\|\alpha\|$	Πλήθος στοιχείων στο σύνολο α
$\alpha\|\|\beta$	Συνένωση (concatenation) των α και β
$\mathcal{H}(\cdot)$	Συνάρτηση κρυπτογραφικού κατακερματισμού (cryptographic hashing) – π.χ. SHA-1
$\mathcal{S}_\kappa(\cdot)$	Ψηφιακή υπογραφή (digital signature) με χρήση του κλειδιού κ
$\mathcal{E}_\kappa(\cdot)$	Κρυπτογράφηση (encryption) με χρήση του κλειδιού κ

Πίνακας 1.1: Βασική σημειολογία

Συμβολίζουμε με $\mathcal{H}(\alpha)$ τον κρυπτογραφικό κατακερματισμό (crytpographic hashing) ενός αντικειμένου α (κόμβος, πόρος, αρχείο, κτλ.), χρησιμοποιώντας κάποια μονόδρομη, ανθεκτική σε συγκρούσεις συνάρτηση κατακερματισμού (one-way collision-resistant hash function – CRHF). Χρησιμοποιούμε τις συναρτήσεις αυτές για σκοπούς επαλήθευσης ακεραιότητας (integrity verification) και ως απόδειξη γνώσης μίας τιμής α χωρίς αποκάλυψη της τιμής αυτής ([53]).

Θα χρησιμοποιούμε τους συμβολισμούς $n.k_p, n.k_s$ και $n.k$ για να δηλώσουμε αντίστοιχα ένα κοινό (public), ένα προσωπικό (private/secret) και ένα συμμετρικό (symmetric) κλειδί τα οποία ανήκουν σε μία οντότητα n. Επιπλέον, θα χρησιμοποιούμε τον συμβολισμό $\mathcal{S}_\kappa(\alpha)$ για να υποδηλώσουμε μία υπογραφή επί του αντικειμένου α και τον συμβολισμό $\mathcal{E}_\kappa(\alpha)$ για να υποδηλώσουμε την κρυπτογράφηση του αντικειμένου α, χρησιμοποιώντας το κλειδί κ.

Επιπλέον, θα συμβολίζουμε με $\alpha.id$ το χαρακτηριστικό ταυτοποίησης του αντικειμένου α, με $|\mathcal{X}|$ το πλήθος των οντοτήτων που υπάρχουν σε ένα σύνολο $\mathcal{X}$, και με $\alpha||\beta$ την αλυσιδωτή ένωση (concatenation) των α και β. Επομένως, το σύμβολο $\mathcal{S}_{n.k_s}(\mathcal{H}(a||b))$ είναι η υπογραφή χρησιμοποιώντας το προσωπικό κλειδί της οντότητας n επί της εξόδου της κρυπτογραφικής συνάρτησης κατακερματισμού με είσοδο την ένωση των a και b, ενώ το σύμβολο $\mathcal{E}_{n.k_p}(a)$ υποδηλώνει την κρυπτογράφηση του αντικειμένου a χρησιμοποιώντας το κοινό κλειδί της οντότητας n.

1.6 Δομή της διατριβής

Στη συνέχεια της διατριβής αυτής (κεφ. 2) θα περιγράψουμε τις κύριες έννοιες καθώς και την βασική υποδομή που απαιτείται για την λειτουργία του SeAl. Κατόπιν θα επικεντρώσουμε στους δύο μηχανισμούς που απαρτίζουν το σύστημα SeAl (κεφ. 3 και κεφ. 4), θα περιγράψουμε τον τρόπο λειτουργίας τους και θα δούμε περιπτώσεις εφαρμογής τους στην πράξη. Στη συνέχεια θα παρουσιάσουμε τα αποτελέσματα των προσομοιώσεων και των πειραματικών μετρήσεών μας (κεφ. 5), θα συζητήσουμε σχετικές εργασίες (κεφ. 6), και θα κλείσουμε (κεφ. 7) με τα συμπεράσματα της εργασίας αυτής και τις πιθανές μελλοντικές της επεκτάσεις.

Κεφάλαιο 2

Το σύστημα SeAl

Το σύστημα SeAl έχει ως σκοπό του την δημιουργία ενός περιβάλλοντος δικτύου ομοτίμων όπου οι ιδιοτελείς και οι κακόβουλοι κόμβοι θα ανιχνεύονται και θα απομονώνονται από το υπόλοιπο δίκτυο, ενώ οι αλτρουιστικοί κόμβοι θα επιβραβεύονται. Για να το κατορθώσει αυτό, υλοποιεί έναν μηχανισμό παρακολούθησης και καταγραφής των προτύπων συμπεριφοράς των κόμβων του δικτύου, χρησιμοποιώντας καινοτόμους μηχανισμούς, δομές δεδομένων, και πρωτόκολλα.

Η βασική ιδέα πίσω από το σχεδιασμό του SeAl έγκειται στο ότι οι (αλτρουιστικοί ή μη) κόμβοι του δικτύου ομοτίμων επιβάλλουν στους ιδιοτελείς ή κακόβουλους κόμβους να προσφέρουν περιεχόμενο και να συμορφώνονται με τους κανόνες συμμετοχής του δικτύου, ούτως ώστε να αποφύγουν την απομόνωσή τους από την υπόλοιπη κοινότητα. Ο μηχανισμός επίβλεψης/καταγραφής του συστήματος SeAl βασίζεται στην καινοτόμα έννοια των «χαρών».

Για λόγους απλότητας θα υποθέσουμε ότι το σύστημά μας χρησιμοποιείται σε ένα περιβάλλον διαμοιρασμού αρχείων (π.χ. διαμοιρασμού αρχείων μουσικής MP3). Ωστόσο, οι αλγόριθμοι και οι μηχανισμοί μας είναι εφαρμόσιμοι και σε άλλες κατηγορίες εφαρμογών δικτύων ομοτίμων, ειδικά όταν αυτές διαχειρίζονται μεγάλα αντικείμενα (π.χ. ψηφιακές βιβλιοθήκες).

Στη συνέχεια ορίζουμε τις βασικές έννοιες και τη σημειολογία που χρησιμοποιούμε στη συνέχεια της διατριβής αυτής, καθώς και τη βασική υποδομή που απαιτείται για τη λειτουργία του συστήματος SeAl.

2.1 «Χάρες»

Λέμε ότι ένας κόμβος $n_1 \in \mathcal{N}$, όπου $\mathcal{N}$ το σύνολο όλων των κόμβων του δικτύου ομοτίμων, χρωστά στον κόμβο $n_2 \in \mathcal{N}$ μία χάρη $f(n_1, n_2, r)$, όταν ο n_1 προσπελαύνει τον πόρο r ο οποίος διαμοιράζεται από τον n_2 (π.χ. όταν ο n_1 μεταφορτώνει αρχεία διαμοιραζόμενα από τον n_2). Κάθε κόμβος n_i διατηρεί μία λίστα $n_i.F_o$ χαρών που χρωστά σε άλλους κόμβους (δηλ. $f(n_1, n_2, r) \in n_1.F_o$) και μία λίστα $n_i.F_d$ χαρών που έχει κάνει σε άλλους κόμβους (δηλ. $f(n_1, n_2, r) \in n_2.F_d$).

Οι χάρες αυτές αποτελούν μια πρώτης τάξης ένδειξη του αλτρουισμού ή της ιδιοτέλειας ενός κόμβου του δικτύου. Το γεγονός ότι ένας συγκεκριμένος κόμβος έχει μεγάλη F_d και μικρή F_o, δείχνει ότι ο κόμβος αυτό είναι μάλλον αλτρουιστικός, συνεισφέροντας στο σύστημα περισσότερους πόρους από αυτούς που ο ίδιος καταναλώνει. Αντίστροφα, μία μικρή F_d και μία μεγάλη F_o δείχνουν ότι ο αντίστοιχος κόμβος είναι μάλλον ιδιοτελής.

2.1.1 Ορισμός ιδιοτέλειας/αλτρουισμού

Στην ιδανική περίπτωση, όλοι οι κόμβοι του δικτύου συνεισφέρουν στο σύστημα το ίδιο ποσό περιεχομένου ή τον ίδιο αριθμό πόρων με αυτό που καταναλώνουν και όλο το σύστημα είναι σε απόλυτη ισορροπία. Έτσι, αν υποθέσουμε ότι τα αρχεία που διαμοιράζονται έχουν όλα το ίδιο μέγεθος, οι λίστες F_o και F_d πρέπει να έχουν το ίδιο μέγεθος για κάθε κόμβο του δικτύου ομοτίμων. Άρα θα πρέπει να ισχύει ότι $|n_i.F_o| = |n_i.F_d|, \forall n_i \in \mathcal{N}$, όπου $|\mathcal{X}|$ είναι το πλήθος των στοιχείων του συνόλου $\mathcal{X}$.

Έχοντας αυτό κατά νου, ορίζουμε το επίπεδο αλτρουισμού $n_i.A$ ενός κόμβου n_i ως συνάρτηση του μεγέθους των λιστών F_o και F_d του κόμβου, χρησιμοποιώντας

είτε το λόγο των αντίστοιχων μεγεθών ($n_i.A = \frac{|n_i.F_d|}{|n_i.F_o|}$) είτε τη διαφορά τους ($n_i.A = |n_i.F_d| - |n_i.F_o|$). Όσο μεγαλύτερη (μικρότερη) είναι η τιμή του χαρακτηριστικού αυτού, τόσο πιο αλτρουιστής (ιδιοτελής) είναι ο αντίστοιχος κόμβος.

2.1.2 VAT – το Διάνυσμα Αλτρουισμού

Επιπλέον, προκειμένου να έχουμε μία εποπτική εικόνα της συνολικής κατάστασης του συστήματος όσον αφορά το επίπεδο αλτρουιστικής συμπεριφοράς που έχουμε επιτύχει, χρησιμοποιούμε μία νέα μετρική: το Διάνυσμα Αλτρουισμού (Vector of AlTruism ή VAT).

Το Διάνυσμα Αλτρουισμού κατασκευάζεται ως εξής. Για κάθε κόμβο $n_i \in \mathcal{N}$ στο SeAl, αποθηκεύουμε στο VAT την τιμή του επιπέδου αλτρουισμού του κόμβου $n_i.A$, και στη συνέχεια το ταξινομούμε κατά φθίνουσα σειρά (άρα $VAT[k] \geq VAT[l] \Leftrightarrow k \geq l$). Όπως προαναφέραμε, στην ιδανική κατάσταση ισορροπίας ισχύει ότι $|n_i.F_d| = |n_i.F_o|, \forall n_i \in \mathcal{N}$, και άρα $VAT = \{1, 1, \ldots, 1\}$, αν χρησιμοποιούμε το λόγο, ή $VAT = \{0, 0, \ldots, 0\}$ αν χρησιμοποιούμε τη διαφορά των μεγεθών των λιστών χαρών των κόμβων.

Πρέπει να τονίσουμε ότι το VAT δεν είναι κάποια κατανεμημένη δομή που διατηρείται από τους κόμβους του δικτύου ομοτίμων, παρά μόνο ένας τρόπος απεικόνισης της κατάστασης του συστήματος όσον αφορά το επίπεδο αλτρουισμού των συμμετεχόντων κόμβων. Πιστεύουμε ότι με χρήση των κατάλληλων αναλυτικών ή πιθανοτικών τεχνικών (π.χ. με χρήση τεχνικών από το πεδίο της Θεωρίας Μεγιστοποίησης (majorization theory)) η πληροφορία που αποθηκεύεται στο VAT μπορεί να οδηγήσει σε μια πληρέστερη κατανόηση της εξέλιξης του συστήματος. Περαιτέρω διερεύνηση του θέματος αυτού αφήνεται ως αντικείμενο μελλοντικών επεκτάσεων της παρούσας εργασίας.

2.2 Βασική υποδομή

Θα περιγράψουμε τώρα την βασική υποδομή που απαιτείται για την ορθή και αποτελεσματική λειτουργία του συστήματος SeAl. Πολλά από τα χαρακτηριστικά που χρειαζόμαστε για να εγγυηθούμε την αποδοτικότητα των αλγορίθμων και των μηχανισμών μας, είναι ήδη παρόντα στα περισσότερα συστήματα δικτύων ομοτίμων σε ευρεία χρήση στο διαδίκτυο. Για άλλα είναι υπό συζήτηση η υλοποίηση ή/και η προσθήκη τους σε κάποιο P2P σύστημα, ενώ υπάρχουν και χαρακτηριστικά τα οποία προτείνουμε εμείς ως προσθήκες στα παραπάνω. Σε κάθε περίπτωση, θα αναφέρουμε σε ποιά από τις κατηγορίες αυτές ανήκει το υπό εξέταση χαρακτηριστικό.

2.2.1 Δίαυλος επικοινωνίας

Υποθέτουμε ότι ο υποκείμενος δίαυλος επικοινωνίας προσφέρει υπηρεσίες εξακρίβωσης της αυθεντικότητας (data authentication) και επαλήθευσης της ακεραιότητας (integrity verification) των δεδομένων που μεταδίδονται μέσω αυτού. Αυτό μπορεί να επιτευχθεί με κάποιο σχήμα κρυπτογράφησης και επαλήθευσης από-άκρη-σε-άκρη (end-to-end), από-σημείο-σε-σημείο (point-to-point) ή μεταξύ όλων των σημείων (all-point). Επίσης, η εμπιστευτικότητα (confidentiality) και η ανωνυμία (anonymity)[10] είναι επιθυμητά, αλλά όχι απαιτούμενα, χαρακτηριστικά του διάυλου επικοινωνίας.

Δίαυλοι επικοινωνίας με τα παραπάνω χαρακτηριστικά έχουν ερευνηθεί και αναλυθεί εκτενώς για ανταλλαγή μηνυμάτων ανάμεσα σε ομάδες χρηστών (group communication/application-level multicasting), καθώς και για μετάδοση μηνυμάτων σε εφαρμογές τύπου Usenet. Κλασσικό παράδειγμα τέτοιων μηχανισμών είναι οι λεγόμενοι remailers και τα mixes[2, 13, 14, 17, 41, 42, 44] και οι εφαρμογές τους σε πεδία όπως η πλοήγηση στον Παγκόσμιο Ιστό[11, 31, 62, 63, 71] και τα δίκτυα ομοτίμων[19, 26, 27, 37, 76].

Ανάλογα με το περιβάλλον στο οποίο θα χρησιμοποιηθεί το σύστημα SeAl, εν-

δέχεται να μην απαιτούνται τόσο ισχυρές εγγυήσεις ή να μην επιτρέπεται τόσο μεγάλο λειτουργικό κόστος, όσο δίνουν ή απαιτούν αντίστοιχα οι παραπάνω μηχανισμοί. Η τελική επιλογή εξαρτάται σε μεγάλο βαθμό από την ζητούμενη λειτουργικότητα και σημασιολογία (semantics) της εφαρμογής που υλοποιείται.

2.2.2 Διαχείριση δεδομένων καταγραφής και προσβάσεων

Ανεξάρτητα από τον τύπο και την τοπολογία του υποκείμενου δικτύου ομοτίμων, ο μηχανισμός επίβλεψης/καταγραφής του συστήματος SeAl χρησιμοποιεί ένα δικό του δικτυακό υπόστρωμα (network overlay) βασισμένο σε distributed hash tables (DHTs) για να αποθηκεύει τα δεδομένα που σχετίζονται με τη συμπεριφορά των κόμβων του δικτύου και με τη γενικότερη λειτουργία του SeAl. Οι κύριες σχεδιαστικές παράμετροι πίσω από την επιλογή αυτή είναι:

1. Η απαιτούμενη λειτουργικότητα και σημασιολογία (δηλ. ντετερμινιστική αποθήκευση και ανάκτηση, παροχή εγγυήσεων για ανθεκτικότητα σε σφάλματα, καλή κλιμάκωση, κτλ.)

2. Η δρομολόγηση των αιτήσεων αποθήκευσης και ανάκτησης πληροφορίας με όσο το δυνατόν λιγότερα hops στο δικτυακό υπόστρωμα.

3. Η μικρότερη δυνατή επιβάρυνση του δικτυακού υποστρώματος, δεδομένης της παρεχόμενης λειτουργικότητας.

4. Η δυνατότητα επαναχρησιμοποίησης υλοποιήσεων και ερευνητικών αποτελεσμάτων για την βελτίωση τμημάτων ή ολοκλήρου του υποστρώματος αυτού.

Τα δικτυακά υποστρώματα κατανεμημένων πινάκων κατακερματισμού παρέχουν (στην πλειοψηφία τους) ισχυρές στατιστικές/πιθανοτικές εγγυήσεις και επιτυγχάνουν δρομολόγηση αιτήσεων σε αριθμό hops λογαριθμικό ως προς το πλήθος των συμμετεχόντων κόμβων. Το ακριβές υπόστρωμα που θα χρησιμοποιηθεί μπορεί να είναι οποιοδήποτε από τα ευρέως γνωστά DHT (π.χ. CAN[61], Chord[39], Kademlia[50],

Pastry[22], Tapestry[81], κτλ.) Για το υπόλοιπο της διατριβής αυτής θα υποθέσουμε ότι χρησιμοποιούμε το σύστημα Chord.

Φυσικά, αν το υποκείμενο δίκτυο ομοτίμων χρησιμοποιεί ήδη ένα DHT υπόστρωμα, τότε το σύστημα SeAl μπορεί να εκμεταλλευτεί αυτή την επιπλέον λειτουργικότητα που του παρέχεται, αντί να αναπτύξει ένα δικό του ξεχωριστό υπόστρωμα. Αξίζει να σημειώσουμε εδώ ότι πρόσφατα έχουν υπάρξει αρκετές ερευνητικές εργασίες οι οποίες προσπαθούν να χρησιμοποιήσουν δομημένα δίκτυα ομοτίμων για να βελτιώσουν την απόδοση των υπαρχόντων αδόμητων δικτύων ομοτίμων, με πιο σημαντική αυτή των Castro, Costa και Rowstron[12]. Επίσης, την κατεύθυνση αυτή φαίνεται να ακολουθούν και πολλά από τα αδόμητα δίκτυα ομοτίμων που βρίσκονται σε ευρεία χρήση (π.χ. το σύστημα JXTA[33] της Sun Microsystems χρησιμοποιεί το Chord ως υπόστρωμα ανάλυσης ονομάτων (naming resolution layer), ενώ τα συστήματα MojoNation[55] και LimeWire[45] ήδη χρησιμοποιούν το Chord για την δρομολόγηση αιτήσεων).

2.2.3 Ζεύγη κλειδιών

Κάθε κόμβος n_i στο σύστημα SeAl διατηρεί ένα ζεύγος κλειδιών κρυπτογράφησης κοινού κλειδιού (public-key cryptography[53]) – ένα μυστικό (secret/private) κλειδί $n_i.k_s$ κι ένα κοινό (public) κλειδί $n_i.k_p$. Τα κλειδιά αυτά δημιουργούνται κατά την είσοδο και εγγραφή του κόμβου στο δίκτυο ομοτίμων και διατηρούνται για όσο χρονικό διάστημα ο κόμβος αποτελεί μέλος του συστήματος. Το ζεύγος κλειδιών αυτό χρησιμοποιείται με δύο τρόπους:

1. από το υπόστρωμα επίβλεψης/κατραγραφής (SAL) του συστήματος SeAl, για να ταυτοποιεί τον κόμβο n_i κατά τη διάρκεια των αλληλεπιδράσεών του με τους υπόλοιπους κόμβους του δικτύου, και

2. από το υπόστρωμα επαλήθευσης (SVL) του συστήματος SeAl, για να εγγυάται την ακεραιότητα και την εμπιστευτικότητα των πόρων/αρχείων και των μηνυμάτων

που κυκλοφορούν στο δίκτυο ομοτίμων.

Υποθέτουμε ότι τα κοινά κλειδιά όλων των κόμβων στο σύστημα είναι προσβάσιμα από όλους τους υπόλοιπους κόμβους. Η λειτουργικότητα αυτή μπορεί να παρέχεται είτε από κάποιο σύστημα Υποδομής Κοινού Κλειδιού (Public-Key Infrastructure – PKI), είτε με κάποιο κατανεμημένο τρόπο (π.χ. μέσω ενός συστήματος σαν το SDSI/SPKI[40] ή το PGP Web of Trust). Μία ενδιαφέρουσα εναλλακτική είναι να παρέχει το SeAl από μόνο του την λειτουργικότητα αυτή, με τρόπο παρόμοιο με αυτό του [4]. Σε κάθε περίπτωση, θεωρούμε ότι η μέθοδος που θα ακολουθηθεί για την παροχή της λειτουργικότητας αυτής, αποτελεί ζήτημα ορθογώνιο με την λειτουργία του συστήματος SeAl.

2.2.4 Ταυτοποίηση πόρων και συναλλαγών

Υποθέτουμε ότι οι πόροι καθώς και όλες οι συναλλαγές μεταξύ κόμβων στο σύστημα ταυτοποιούνται με βάση ένα μοναδικό χαρακτηριστικό ταυτοποίησης (π.χ. UUID[52] των 160-bits). Συνήθως (σε πραγματικά συστήματα) το χαρακτηριστικό αυτό παράγεται με χρήση κάποιου κρυπτογραφικού αλγορίθμου κατακερματισμού (secure hash function), όπως για παράδειγμα ο SHA-1[23] ή ο MD5[64].

Αξίζει να αναφέρουμε εδώ ότι έχει αποδειχθεί αναλυτικά[75] ότι ο αλγόριθμος MD5 δεν είναι αρκετά ασφαλής για χρήση σε εφαρμογές που απαιτούν ισχυρή κρυπτογράφηση. Επίσης, πρόσφατα ξεκίνησε και ένα έργο κατανεμημένου υπολογισμού[51] το οποίο ασχολείται με την πρακτική πλευρά του θέματος. Σε κάθε περίπτωση, το σύστημα SeAl χρησιμοποιεί είτε τον αλγόριθμο SHA-1 ή παραλλαγές του (π.χ. SHA-256, SHA-512, κτλ.) περιορίζοντας (με truncation) την έξοδό τους στα 160 bits.

2.2.5 Ταυτοποίηση κόμβων

Τα χαρακτηριστικά ταυτοποίησης των κόμβων του συστήματος αποτελούν ειδική περίπτωση· στο σύστημα SeAl κάθε κόμβος ταυτοποιείται με την κρυπτογραφική

περίληψη (secure hash/digest) του κοινού κλειδιού του. Το γεγονός αυτό επιτρέπει στο SeAl να λειτουργεί με τρόπο πλήρως ψευδωνυμικό:

- Στα υπάρχοντα συστήματα οι κόμβοι ταυτοποιούνται με βάση την κρυπτογραφική περίληψη της IP διεύθυνσής τους. Με τη μέθοδό μας, η ταυτοποίηση των κόμβων ανεξαρτητοποιείται από τέτοιου είδους πληροφορία χαμηλού επιπέδου που μπορεί να αντιστοιχισθεί εύκολα με δεδομένα του πραγματικού κόσμου (παραβιάζοντας έτσι την ανωνυμία των συμμετεχόντων).

- Το χαρακτηριστικό ταυτοποίησης ενός κόμβου παραμένει το ίδιο για όσο χρονικό διάστημα ο κόμβος διατηρεί επισης το ίδιο ζεύγος κλειδιών.

- Ένας κόμβος μπορεί να αρνηθεί την ευθύνη κάποιων πράξεών του απλά με τον δημιουργήσει ένα νέο ζεύγος κλειδιών και να διαγράψει το παλιό του (χάνοντας όμως και οποιαδήποτε απόδειξη για όσα έχει ως τώρα καταφέρει στο δίκτυο).

Επιπλεόν, θέλουμε να απαγορεύουμε στους κόμβους να επιλέγουν κατά βούληση την τιμή του χαρακτηριστικού ταυτοποίησής τους (ακολουθώντας τις διδαχές των συστημάτων Freenet[16] και Achord[37]), ώστε να μη μπορούν να αναλαμβάνουν την ευθύνη για περιοχές του δικτύου που αυτοί επιλέγουν. Συνεπώς, για να επιτραπεί σε έναν κόμβο να συνδεθεί στο δίκτυο χρησιμοποιώντας κάποιο ID (και έτσι να αναλάβει μία περιοχή του δικτύου ομοτίμων), πρέπει ο κόμβος αυτός να αποδείξει ότι γνωρίζει το μυστικό κλειδί που αντιστοιχεί στο κοινό κλειδί του οποίου η κρυπτογραφική περίληψη είναι το υπό εξέταση ID. Αυτό μπορεί να γίνει εύκολα, ανάλογα με το υποκείμενο DHT· για παράδειγμα, στο σύστημα Chord ο «πρόγονος» (predecessor) του κόμβου μπορεί να ζητήσει από τον κόμβο αυτό να υπογράψει κάποιο κομμάτι πληροφορίας με το μυστικό του κλειδί και μετά να επιβεβαιώσει ότι όντως η υπογραφή είναι σωστή[1].

[1] Θεωρούμε ότι η πλαστογράφηση ψηφιακών υπογραφών είναι υπολογιστικά αδύνατη.

Σύμβολο	Περιγραφή
$n_i.k_s$	Μυστικό κλειδί του κόμβου n_i
$n_i.k_p$	Κοινό κλειδί του κόμβου n_i
$n_i.k$	Κλειδί συμμετρικού αλγορίθμου του κόμβου n_i
$n_i.id$	Αναγνωριστικό ταυτοποίησης του κόμβου n_i (=$\mathcal{H}(n_i.k_p)$)
$n_i.F_d$	Λίστα χαρών που έχει κάνει ο κόμβος n_i
$n_i.F_o$	Λίστα χαρών που χρωστά ο κόμβος n_i
$n_i.A$	Επίπεδο αλτρουισμού του κόμβου n_i (=$\|F_d\| - \|F_o\|$ ή $\frac{\|F_d\|}{\|F_o\|}$)

Πίνακας 2.1: Βασικές παράμετροι συμμετεχόντων κόμβων

2.2.6 Αντίγραφα

Αν και τα δικτυακά υποστρώματα κατανεμημένων πινάκων κατακερματισμού χρησιμοποιούν εσωτερικά δικές τους τεχνικές για να αποφύγουν απώλειες στην διαμοιραζόμενη πληροφορία λόγω κατατμήσεων του δικτύου (network segmentation) ή άλλων δυσλειτουργιών, μπορούμε να διατηρούμε επιλεκτικά αντίγραφα ασφαλείας (replicas) σημαντικών δεδομένων (π.χ. των κοινών κλειδιών των κόμβων του δικτύου).

Η δημιουργία των αντιγράφων αυτών μπορεί να είναι είτε πλήρης (στην οποία περίπτωση διατηρούμε πλήρη αντίγραφα της σχετικής πληροφορίας σε κόμβους του συστήματος), είτε «κρυφή» (π.χ. χρησιμοποιώντας ένα μηχανισμό κρυφής μνήμης (caching)), είτε χρησιμοποιώντας κάποια τεχνική «διαμοιρασμού μυστικών» (secret-sharing)[68] ή «κωδικοποίησης απαλοιφής» (erasure coding)[46, 48, 77].

Για το υπόλοιπο της διατριβής αυτής θα υποθέτουμε ότι δε χρησιμοποιούμε κάποιο ξεχωριστό μηχανισμό παραγωγής αντιγράφων ασφαλείας και θα στηριζόμαστε στο υποκείμενο δίκτυο ομοτίμων για την παροχή της λειτουργικότητας αυτής.

Κεφάλαιο 3

SAL: Ο μηχανισμός επίβλεψης

Θα ασχοληθούμε τώρα με την ανάλυση του SAL – του μηχανισμού επίβλεψης/καταγραφής του συστήματος SeAl. Προς το παρόν θα περιοριστούμε στην αντιμετώπιση μόνο περιπτώσεων ιδιοτελών χρηστών/κόμβων, όπου οι κόμβοι μπορούν είτε να προσφέρουν όλους τους πόρους τους και τα δεδομένα τους στην P2P κοινότητα («αλτρουιστές»), είτε να κατακρατούν (μέρος των) πόρων και των δεδομένων τους εκτός δικτύου («ιδιοτελείς»).

Υποθέτουμε ότι οι κόμβοι που συμμετέχουν στο δίκτυο ομοτίμων δεν προσπαθούν να υπονομεύσουν τα πρωτόκολλα και τους μηχανισμούς του συστήματος και δεν παρεμβαίνουν στην σωστή λειτουργία των υποστρωμάτων επικοινωνίας και επίβλεψης/καταγραφής. Τέτοιες «εχθρικές» συμπεριφορές θα αντιμετωπιστούν με το SVL – τον μηχανισμό επαλήθευσης του συστήματος SeAl – το οποίο και θα περιγράψουμε στο επόμενο κεφάλαιο.

3.1 Αποδείξεις συναλλαγών και «χάρες»

Κάθε συναλλαγή (μεταφόρτωση) στο σύστημα SeAl τερματίζει με τους δύο εμπλεκόμενους κόμβους να έχουν στην κατοχή τους μία ψηφιακή «απόδειξη» της συναλ-

λαγής, την οποία ονομάζουμε Απόδειξη Συναλλαγής (Transaction Receipt ή TR).

Έστω ότι ο κόμβος n_1 προσπελαύνει τον πόρο/αρχείο r ο οποίος συνεισφέρεται από τον κόμβο n_2. Χρησιμοποιούμε το συμβολισμό $TR(n_1.id, n_2.id, r.id, t)$ για να συμβολίσουμε την Απόδειξη Συναλλαγής για αυτή την πρόσβαση, όπου:

- $n_1.id$ και $n_2.id$ τα χαρακτηριστικά ταυτοποίησης (ID) των δύο συμμετεχόντων κόμβων (παραλήπτη και αποστολέα αντίστοιχα) του αρχείου/πόρου που ανταλλάχθηκε,
- $r.id$ πληροφορίες για την ταυτοποίηση του πόρου/αρχείου που ανταλλάχθηκε (π.χ. το χαρακτηριστικό ταυτοποίησης του πόρου/αρχείου που ανταλλάχθηκε, το μέγεθός του, ένα checksum του κτλ.), και
- t μία χρονοσφραγίδα (timestamp) η οποία καθορίζει τη χρονική στιγμή στην οποία έγινε η αντίστοιχη συναλλαγή.

Δεδομένου ότι στο πεδίο των δικτύων ομοτίμων διαμοιρασμού αρχείων ο κύριος πόρος και ο βασικός ανασχετικός παράγοντας είναι το εύρος ζώνης δικτύου (network bandwidth) που καταναλώνεται για την εκτέλεση των διαφόρων λειτουργιών του συστήματος, οι Αποδείξεις Συναλλαγών περιέχουν (μεταξύ άλλων) και το μέγεθος του διαμοιραζόμενου πόρου, ώς ένδειξη του μεγέθους του έργου που κατανάλωσε ο κόμβος-αποστολέας για την εξυπηρέτηση της αντίστοιχης αίτησης, καθώς και το μέγεθος της «χάρης» που του χρωστά ο κόμβος-παραλήπτης του πόρου/αρχείου. Συνεπώς, οι Αποδείξεις Συναλλαγών δεν είναι απλά ενδείξεις εκτίμησης, αλλά έχουν και ποσοτικό χαρακτήρα.

Η χρονοσφραγίδα χρησιμοποιείται για την υλοποίηση ενός μηχανισμού γήρανσης (aging) για τις Αποδείξεις Συναλλαγών, για δύο κύριους λόγους:

- για να διατηρούνται οι απαιτήσεις σε μνήμη/αποθηκευτικό χώρο σε προκαθορισμένα πλαίσια, και

- για να επιτρέπεται στο σύστημα να προσαρμόζεται σε χρονικές μεταβολές της συμπεριφοράς των χρηστών του.

Σε κάθε περίπτωση, ο ακριβής αλγόριθμος αντικατάστασης/γήρανσης παραμένει θέμα μελλοντικής έρευνας και πειραματισμού.

Οι «χάρες» υλοποιούνται στο σύστημα SeAl χρησιμοποιώντας Αποδείξεις Συναλλαγών. Έτσι, μία εγγραφή στην F_d λίστα του κόμβου n_2, σχετικά με μία συναλλαγή με τον κόμβο n_1 για έναν πόρο r (όπως αυτά αναφέρθηκαν παραπάνω), είναι της μορφής: $\{n_1.id, r.id, t, TR(\cdots)\}$. Μία τέτοια Απόδειξη Συναλλαγής θεωρούμε ότι έχει αξία ίση με $TR.r.size \times \frac{TR.t}{current\ time}$, όπου ο χρόνος (τόσο στο $TR.t$ όσο και στο $current\ time$) εκφράζεται σε σύγκριση με μία αρχική ημερομηνία (Epoch), όπως καθορίζεται από τα πρότυπα BSD και SVr4.

3.2 Αποπληρωμή «χαρών» (και επιβολή της)

Ας υποθέσουμε ξανά ότι ο κόμβος n_1 προσπελαύνει το αρχείο r το οποίο συνεισφέρεται από τον κόμβο n_2 και επομένως ο n_1 χρωστά στον n_2 μία χάρη (δηλ. $f(n_1, n_2, r) \in \{n_2.F_d, n_1.F_o\}$). Τότε, ο κόμβος n_2 μπορεί να χρησιμοποιήσει αυτή την χάρη ως εξής: κάποια επόμενη φορά που ένας κόμβος n_3 θα ζητήσει πρόσβαση στον πόρο r, ο κόμβος n_2 – ο *κόμβος-αποστολέας* – θα έχει την επιλογή να ανακατευθύνει την αίτηση σε έναν από τους κόμβους που του χρωστούν χάρη για τον συγκεκριμένο πόρο (συμπεριλαμβανομένου και του n_2) – τον *κόμβο-αποδέκτη* – και να περιμένει ανάδραση από τον n_3 για το αν τελικά η αίτησή του εξυπηρετήθηκε.

Στη γενική περίπτωση, οι αποδέκτες των ανακατευθύνσεων αυτών επιλέγονται ανάμεσα στους κόμβους που εμφανίζονται στην F_d λίστα του κόμβου-αποστολέα. Ο αλγόριθμος επιλογής του καταλληλότερου αποδέκτη – όταν υπάρχουν περισσότεροι από ένας – μπορεί να βασίζεται σε διάφορες ευρεστικές μεθόδους, όπως για παράδειγμα:

- το ιστορικό των συναλλαγών ανάμεσα στους κόμβους αποστολέα και αποδέκτη

(π.χ. ποιά αρχεία έχουν μεταφορτωθεί στον αποδέκτη από τον αποστολέα),

- τις υπολογιστικές/αποθηκευτικές/δικτυακές δυνατότητες του αποδέκτη (π.χ. μετρούμενες ή αναφερόμενες μέσω PING μηνυμάτων, όπως στο σύστημα Gnutella),

- τη συνολική εικόνα του κόμβου αποδέκτη (όπως αποτυπώνεται από το επίπεδο αλτρουισμού του κόμβου αυτού), ή

- μία ευρεστική μέθοδος η οποία να συνδυάζει δύο ή περισσότερα από τα παραπάνω.

Σε κάθε περίπτωση, η ακριβής υλοποίηση του αλγορίθμου επιλογής κόμβου-αποδέκτη παραμένει αντικείμενο μελέτης και μελλοντικής έρευνας.

Και οι τρεις εμπλεκόμενοι κόμβοι κρατούν στοιχεία για τη συναλλαγή αυτή, προσθέτοντας τις αντίστοιχες εισαγωγές στις λίστες τους· ο n_3 χρωστά πλέον χάρη στον n_1 (δηλ. $f(n_3, n_1, r) \in \{n_1.F_d, n_3.F_o\}$), ενώ τόσο ο n_1 όσο και ο n_2 σημειώνουν τις αντίστοιχες εισαγωγές στις λίστες τους ως «αποπληρωμένες».

Όπως αναφέραμε σε προηγούμενο κεφάλαιο, κάθε κόμβος διατηρεί ένα χαρακτηριστικό που ονομάζεται «επίπεδο αλτρουισμού» και συμβολίζεται ως $n_i.A$. Η ακριβής μαθηματική έκφραση που χρησιμοποιείται από κάθε κόμβο για τον καθορισμό του επιπέδου αλτρουισμού του εξαρτάται μόνο από τοπικές αποφάσεις του διαχειριστή του κόμβου· ο διαχειριστής μπορεί αυτόνομα να επιλέξει ανάμεσα στις συναρτήσεις $\frac{|F_d|}{|F_o|}$ και $|F_d| - |F_o|$. Επιπλέον, οι διαχειριστές των κόμβων ορίζουν μία άνω και μία κάτω τιμή κατωφλίου για το επίπεδο αλτρουισμού τους: $n_i.A_{max}$ και $n_i.A_{min}$ αντίστοιχα. Με αυτά τα δεδομένα, οι κόμβοι επιλέγουν αυτόματα αν θα ανακατευθύνουν ή όχι μία εισερχόμενη αίτηση πρόσβασης, βασιζόμενοι στο τρέχον επίπεδο αλτρουισμού τους και στις παραπάνω τιμές κατωφλίου· πάντα ανακατευθύνουν όταν $n_i.A > n_i.A_{max}$, ποτέ όταν $n_i.A < n_i.A_{min}$, και ακολουθώντας μία προκαθορισμένη συνάρτηση πιθανότητας $\mathcal{P}(\mathcal{R})$ όταν $n_i.A_{min} \leq n_i.A \leq n_i.A_{max}$.

Μία αίτηση μπορεί να ανακατευθύνεται αναδρομικά από κόμβο σε κόμβο, μέχρι ενός προκαθορισμένου βάθους αναδρομής, δημιουργώντας έτσι μία αλυσίδα από κόμβους (οι κύκλοι αποφεύγονται χρησιμοποιώντας το αναγνωριστικό ταυτοποίησης της αίτησης). Στην περίπτωση αυτή, μόνο οι δύο τελευταίοι κόμβοι της αλυσίδας σημειώνουν ως αποπληρωμένες τις αντίστοιχες εισαγωγές στις λίστες τους. Το σκεπτικό πίσω από αυτό είναι ότι το κόστος ανακατεύθυνσης μίας αίτησης είναι αμελητέο σε σύγκριση με το κόστος εξυπηρέτησής της· έτσι μόνο ο τελευταίος κόμβος στην αλυσίδα έχει πραγματικά αποπληρώσει την χάρη που χρωστούσε στον προτελευταίο κόμβο της αλυσίδας.

Αλγόριθμος 1 Αλγόριθμος ανακατεύθυνσης εισερχόμενης αίτησης

$n.A$: επίπεδο αλτρουισμού τρέχοντα κόμβου.
$n.A_{min,max}$: τιμές κατωφλίου επιπέδου αλτρουισμού τρέχοντα κόμβου.
$\mathcal{P}(\mathcal{R})$: πιθανότητα ανακατεύθυνσης.
$send(Msg, n_{target})$: συνάρτηση αποστολής μηνύματος του υποκείμενου DHT.
$F_{o,d}.find(favor)$: συνάρτηση αναζήτησης σε λίστες χαρών.

status_t redirect(Msg m **)**

1: **if** (($n.A > n.A_{max}$) **or**
   ($n.A > n.A_{min}$ **and** $n.A < n.A_{max}$ **and** $w.p.\ \mathcal{P}(\mathcal{R})$)) **then**
2: **while** ($\exists f \in n.F_d : \{f.r == m.r \ \&\& \ f.n_{client}.id \notin m.visited \ \&\& \ f.paidBack == false$) **do**
3: $m.visited+ = f.n_{client}.id$;
4: **if** ($f.n_{client}$ does not respond) **then**
5: blacklist(f);
6: **else**
7: Send(m, $f.n_{client}.id$);
8: **if** ($m.n_{client}$ does not report success) **then**
9: blacklist(f);
10: **else**
11: $n.F_d.find(f).paidBack = true$;
12: $f.n_{client}.F_d.find(f).paidBack = true$;
13: Return $success\_t$;
14: Return $failure\_t$;

3.3 Κακή φήμη – «Μαρτυρίες Κατηγορίας»

Κάθε παρέκκλιση ενός κόμβου από την αναμενόμενη συμπεριφορά μπορεί να προκαλέσει την καταγραφή του ως μη νόμιμου κόμβου. Αν θεωρήσουμε το παράδειγμα των κόμβων n_1 και n_2 που αναφέραμε προηγούμενα, και υποθέσουμε ότι ο κόμβος n_1 αρνείται να αποπληρώσει μία χάρη, δηλαδή να εξυπηρετήσει μια αίτηση που του ανακατεύθυνε ο n_2, τότε ο τελευταίος μπορεί να χρησιμοποιήσει την αντίστοιχη εγγραφή από την F_d λίστα του για να «κατηγορήσει» (blacklist) τον n_1 (δηλ. να τον σημειώσει ως μη νόμιμο). Μπορούμε επίσης να χρησιμοποιήσουμε έναν αλγόριθμο δεύτερης ευκαιρίας – second chance – αντί για τον παραπάνω αυστηρό αλγόριθμο. Αξίζει να σημειώσουμε ότι το να είναι ένας κόμβος εκτός δικτύου προσμετράται ως ιδιοτελής συμπεριφορά, αφού προς το παρόν ο κόμβος αυτός δεν προσφέρει τίποτα στην P2P κοινότητα.

Εν ολίγοις, όταν ο κόμβος n_2 θέλει να «κατηγορήσει» τον κόμβο n_1 για παράνομη (ιδιοτελή ή κακόβουλη) συμπεριφορά, δημιουργεί μία «Μαρτυρία Κατηγορίας» (Blacklist Record ή BLR) και την δημοσιεύει στο υποκείμενο DHT. Η Μαρτυρία Κατηγορίας δεν είναι τίποτα άλλο από την Απόδειξη Συναλλαγής της χάρης που δεν αποπλήρωσε ο n_1, στην οποία όμως το ID του n_1 αντικαθίσταται από την περίληψη (secure hash) του αρχικού ID του n_1 και η οποία δημοσιεύεται χρησιμοποιώντας ως κλειδί το $\mathcal{H}(\text{«}BLR\text{»}||\mathcal{H}(n_1.id))$. Έτσι, η πιθανότητα του ενδεχομένουν ένας κόμβος να είναι υπεύθυνος του τμήματος του χώρου των ID στο οποίο αντιστοιχίζονται οι Μαρτυρίες Κατηγορίας που τον αφορούν, είναι αμελητέα. Παράλληλα, ο κόμβος που τελικά θα είναι υπεύθυνος για την αποθήκευση μίας Μαρτυρίας Κατηγορίας δεν γνωρίζει για ποιον κόμβο κρατά πληροφορία, ακριβώς λόγω της χρησιμοποίησης περίληψης του ID του κόμβου στο BLR και της υπολογιστικής πολυπλοκότητας της αντιστροφής μίας συνάρτησης ασφαλούς κατακερματισμού.

Αξίζει να σημειώσουμε ότι η χρησιμοποίηση και η δημοσίευση των BLR δεν θέτει κανέναν περιορισμό στην κλιμάκωση του συστήματός μας. Μεταχειριζόμαστε αυτά

τα αντικείμενα όπως και κάθε άλλο αντικείμενο σε ένα DHT (εκτός, προφανώς, από τις επιπλέον λειτουργίες που προσθέτει ο μηχανισμός επαλήθευσης (SVL), τις οποίες θα περιγράψουμε στο επόμενο κεφάλαιο). Άρα, η αποθήκευση και η ανάκτηση ενός (συνόλου από) BLR, απαιτεί τον αριθμό hops και το χρόνο που καθορίζει και παρέχει το υποκείμενο υπόστρωμα κατανεμημένου πίνακα κατακερματισμού (π.χ. $O(log(N))$ hops για δίκτυο N κόμβων, χρησιμοποιώντας το σύστημα Chord). Ζητήματα όπως η ανεκτικότητα σε σφάλματα και η διαθεσιμότητα των δεδομένων, αφήνονται στο DHT, αν και θα μπορούσε να χρησιμοποιηθεί κάποιος επιπλέον μηχανισμός (βλπ. ενότητα 2.2.6).

Αλγόριθμος 2 Αλγόριθμος δημοσίευσης Μαρτυρίας Κατηγορίας

$DHT.put(key, value)$: συνάρτηση δημοσίευσης υποκείμενου DHT.

void blacklist(Favor f **)**

1: **if** ($f.blackListed == true$) **then**
2: Return;
3: $B_f = f : \{f.n_{client}.id \Leftarrow \mathcal{H}(f.n_{client}.id)\}$;
4: $DHT.put(\mathcal{H}(\text{“}BLR''||f.n_{client}.id), B_f)$;

Ο αποδέκτης της παραπάνω (και οποιασδήποτε) Μαρτυρίας Κατηγορίας μπορεί να επιλέξει είτε να δώσει στον n_1 μία δεύτερη ευκαιρία, με πιθανότητα $\mathcal{P}(\mathcal{SC})$, είτε να τον καταγράψει απευθείας ως παράνομο (δηλ. να αποθηκεύσει το BLR) με πιθανότητα $1 - \mathcal{P}(\mathcal{SC})$.

3.4 Οι «χάρες» στην πράξη – «Μαρτυρίες Προσφοράς»

Οι κόμβοι ερευνούν περιοδικά το υπόστρωμα DHT για Μαρτυρίες Κατηγοριών που έχουν δημοσιευτεί εναντίον τους. Έτσι, κάθε κόμβος μαθαίνει αργά ή γρήγορα το επίπεδο αλτρουισμού του, όπως αυτό γίνεται αντιληπτό από τους υπόλοιπους κόμβους στο δίκτυο ομοτίμων.

Έστω ότι ο κόμβος n_1 επικοινωνεί με τον κόμβο n_2 για να αποκτήσει προσβα-

ση σε έναν πόρο/αρχείο r. Ο n_2 ζητά από τον n_1 να επιλέξει ένα τμήμα W_m των εισαγωγών από την F_d λίστα του ($n_1.F_d$), τις οποίες θα αποκαλούμε «Μαρτυρίες Προσφοράς» (white records), έτσι ώστε το άθροισμα των αξιών των επιλεγμένων Μαρτυριών Προσφοράς να είναι μικρότερο ή ισο από μία τοπικά επιλεγμένη τιμή κατωφλίου.

Αλγόριθμος 3 Αλγόριθμος δημιουργίας μηνύματος αίτησης προσπέλασης

n_{client}: τρέχων κόμβος.
k: τιμή κατωφλίου συνολικής αξίας επιλεγόμενων Μαρτυριών Προσφοράς.
$F_d.select(amount)$: συνάρτηση επιλογής Μαρτυριών Προσφοράς συνολικής αξίας ως μία τιμή κατωφλίου.

Msg genRequest(Node n_{server}**, Resource** r **)**
1: $W_m = F_d.select(k)$;
2: $m_{r_u} = \{n_{server}.id, n_{client}.id, r.id, |W_m|, t\}$;
3: $m_{r_s} = \mathcal{S}_{n_{client}.k_s}(m_{r_u})$;
4: Return m_{r_s};

Ο n_1 παρουσιάζει το άθροισμα των αξιών των Μαρτυριών Προφοράς αυτών στον n_2, ο οποίος αναθέτει στην αίτηση του πρώτου μία αρχική θετική βαθμολογία s_w ίση με το άθροισμα αυτό. Στη συνέχεια, ο n_2 ερευνά το DHT υπόστρωμα για Μαρτυρίες Κατηγορίας εναντίον του n_1 με άθροισμα αξιών s_b μικρότερο ή ίσο με το s_w. Τελικά, η αίτηση βαθμολογείται με $s_w - s_b - r.size$ (με ελάχιστη τιμή την $-r.size$), όπου $r.size$ το μέγεθος του πόρου/αρχείου τον οποίον θέλει να προσπελάσει ο n_1.

Οι εισερχόμενες αιτήσεις στη συνέχεια τοποθετούνται σε μία ουρά αναμονής, ταξινομημένη με βάση τη βαθμολογία των αιτήσεων. Συνεπώς, η χρονοδρομολόγηση της εξυπηρέτησης των αιτήσεων βασίζεται στην προσφορά των κόμβων στην υπόλοιπη κοινότητα. Βασιζόμενοι σε τοπικά κριτήρια τιθέμενα από τους διαχειριστές των κόμβων του δικτύου, αιτήσεις από κόμβους με χαμηλή βαθμολογία μπορούν είτε να εξυπηρετούνται μόνο όταν ο κόμβος-εξυπηρετητής είναι αδρανής, είτε να μην εξυπηρετούνται καθόλου, είτε να ανατίθενται περιορισμένοι πόροι για την εξυπηρέτησή τους (π.χ. με περιορισμό στις συνδέσεις ή στο χρησιμοποιούμενο έυρος ζώνης δικτύου κτλ.), εισαγάγοντας έτσι μία ρύθμιση για την αυστηρότητα της χρονοδρομολόγησης

των αιτήσεων.

Επιπλεόν, αφού η θέση μίας αίτησης στην ουρά αναμονής μπορεί να αλλάξει με την προσθήκη αιτήσεων από κόμβους με υψηλή βαθμολογία, οι κόμβοι με αιτήσεις σε αναμονή πρέπει να ελέγχουν περιοδικά την τρέχουσα θέση τους στην ουρά ή το τρέχον ποσοστό των πόρων του κόμβου-εξυπηρετητή που τους έχει ανατεθεί. Αυτό γίνεται ως τμήμα των πακέτων που ανταλλάσσουν μεταξύ τους οι κόμβοι όυτως ή άλλως, για να διατηρούν το δίκτυο συνδεδεμένο (τα λεγόμενα heartbeats). Ονομάζουμε το μηχανισμό αυτό «Μηχανισμό Ανάδρασης».

Αυτή η στρατηγική χρονοδρομολόγησης δίνει στους κόμβους κίνητρα για να λειτουργούν αλτρουιστικά προς την υπόλοιπη κοινότητα, αφού οι κόμβοι με μεγαλύτερες βαθμολογίες εξυπηρετούνται πρώτοι ή καλύτερα από τους υπόλοιπους. Επιπλέον, πρέπει να τονίσουμε ότι, αφού η συνολική εικόνα ενός κόμβου καθορίζει σε τέτοιο βαθμό και το επίπεδο ικανοποίησης των χρηστών των κόμβων, ο μηχανισμός αυτός αποθαρρύνει και τις Sybil[21] επιθέσεις[1].

Αλγόριθμος 4 Αλγόριθμος προεπεξεργασίας εισερχόμενης αίτησης

$\mathcal{P}(\mathcal{V})$: πιθανότητα επαλήθευσης.

void preprocess(Msg m **)**

1: Verify ($w.p.$ $\mathcal{P}(\mathcal{V})$) the signature on m.
2: **if** (redirect(m)$==failure\_t$) **then**
3: Fetch ($w.p.$ $\mathcal{P}(\mathcal{V})$) and verify ($w.p.$ $\mathcal{P}(\mathcal{V})$) W_m.
4: Compute $m.s_w = \sum_{w \in W_m}(w_i.value)$.
5: Fetch ($w.p.$ $\mathcal{P}(\mathcal{V})$) a subset B_m : $\{m.s'_b = \sum_{b \in B_m}(b_i.value) \leq m.s_w\}$ of the black records filed against n_1.
6: Verify ($w.p.$ $\mathcal{P}(\mathcal{V})$) and compute $m.s_b = \sum_{b \in B_m}(b_i.value)$.
7: Assign $m.score = (m.s_w - m.s_b - r.size)$ and enqueue m.

Τέλος, αξίζει να παρατηρήσουμε ότι:

- ο μηχανισμός αυτός επιτρέπει την εκμετάλλευση των «θετικών συγκυριών» (posi-

[1] Στην επίθεση Sybil ο επιτιθέμενος κόμβος συνδέεται στο σύστημα με διαφορετικό ID κάθε φορά, για να αποφύγει τις συνέπειες μιας πιθανώς αρνητικής φήμης που έχει δημιουργήσει με το προηγούμενο ID του.

tive externalities), όπως είναι οι υποχρησιμοποιούμενοι ή αδρανείς κόμβοι/πόροι.

- οι συμμετέχοντες κόμβοι δεν έχουν να κερδίσουν τίποτα αλλάζοντας ταυτότητα, αφού σε κάθε περίπτωση η απουσία Μαρτυριών Προσφοράς για έναν κόμβο θα έχει ως αποτέλεσμα την βαθμολόγησή του με τον ελάχιστο δυνατό βαθμό.

3.5 Αποπληρωμή χρεών

Όπως προαναφέραμε, οι κόμβοι αργά ή γρήγορα μαθαίνουν την εικόνα που έχει η υπόλοιπη κοινότητα για την προσφορά τους στο κοινό καλό. Αυτό δίνει τη δυνατότητα σε κόμβους με άσχημη συνολική εικόνα να επανορθώσουν για την συμπεριφορά τους στο παρελθόν· οι κόμβοι αυτοί μπορούν να επιλέξουν μία ή περισσότερες από τις Μαρτυρίες Κατηγορίας που έχουν δημοσιευτεί εναντίον τους και να προσφερθούν να αποπληρώσουν τις αντίστοιχες χάρες, ενημερώνοντας τον κόμβο στον οποίον χρωστούν την χάρη για την διαθεσιμότητά τους.

Σε κάθε περίπτωση, αν η αποπληρωμή ολοκληρωθεί επιτυχώς, και οι δύο κόμβοι θα έχουν στην κατοχή τους μία Απόδειξη Συναλλαγής που θα το αποδεικνύει. Το μόνο που απομένει μετά από αυτό είναι να σημειώσουν τοπικά ότι οι αντίστοιχες χάρες έχουν αποπληρωθεί και να ζητήσουν (ο ένας ή και οι δύο) από τον κόμβο που αποθηκεύει τις Μαρτυρίες Κατηγορίας του κόμβου που αποπλήρωσε τις χάρες να διαγράψει τα αντίστοιχα BLR.

Κεφάλαιο 4

SVL: Ο μηχανισμός επαλήθευσης

Ως τώρα θεωρούσαμε ότι οι κόμβοι στο σύστημά μας είτε ήταν ιδιοτελείς, είτε αλτρουιστές, αλλά δεν προσπαθούσαν να παρακάμψουν τον μηχανισμό επίβλεψης/καταγραφής του συστήματος SeAl. Δυστυχώς, σε πραγματικές συνθήκες οι συμμετέχοντες των δικτύων ομοτίμων αποδεδειγμένα[66] προσπαθούν να υπονομεύσουν κάθε μηχανισμό επιβολής συμπεριφοράς. Για να αντιμετωπίσουμε τέτοιες συμπεριφορές, εισαγάγουμε έναν επιπλεόν μηχανισμό στο σύστημά μας: τον μηχανισμό επαλήθευσης – το SeAl Verification Layer ή SVL.

4.1 SVL και Αποδείξεις Συναλλαγών

Προκειμένου να αντιμετωπίσουμε και κακόβουλους κόμβους, πρέπει να προσθέσουμε στις Αποδείξεις Συναλλαγών δυνατότητες για έλεγχο προέλευσης δεδομένων (data origin checking), καθώς και να παρέχουμε εγγυήσεις για μοναδικότητα (uniqueness) και χρονική συνέπεια (timeliness), μετατρέποντάς τις έτσι σε αντικείμενα «πιστοποίησης συναλλαγών» (transaction authentication), όπως αυτά ορίζονται στο [53]. Για να το επιτύχουμε αυτό, κατασκευάζουμε τις Αποδείξεις Συναλλαγών όπως και στην περίπτωση του SAL, μόνο που τώρα το τελικό αντικείμενο υπογράφεται ψη-

φιακά, πρώτα από τον κόμβο-παραλήπτη και κατόπιν από τον κόμβο-αποστολέα.

Κάθε τρίτος κόμβος μπορεί να επαληθεύσει την ορθότητα και την ισχύ μιας Απόδειξης Συναλλαγής, επικυρώνοντας τις ψηφιακές υπογραφές αυτές και ελέγχοντας ότι τα κλειδιά που χρησιμοποιήθηκαν για τις υπογραφές αντιστοιχούν στα χαρακτηριστικά ταυτοποίησης των κόμβων που βρίσκονται στην Απόδειξη[1]. Σύμφωνα πάντα με τοπικές αποφάσεις στον κόμβο που διενεργεί τον έλεγχο, παρέχεται η δυνατότητα ανάκτησης των περιλήψεων συκεκριμένων τμημάτων του πόρου/αρχείου που αναφέρεται στην υπό εξέταση Απόδειξη, ή η ανάκτηση (μέσω κανονικής αίτησης προσπέλασης) ολοκλήρου ή τμημάτων του αρχείου, με σκοπό την επαλήθευση και της αντίστοιχης πληροφορίας της Απόδειξης Συναλλαγής.

4.2 SVL και Μαρτυρίες Κατηγορίας

Με το παραπάνω σχήμα και την χρήση ψηφιακών υπογραφών, κανένας κόμβος δε μπορεί να «πλαστογραφήσει» Μαρτυρίες Κατηγορίας, αφού μία τέτοια ενέργεια ισοδυναμεί με «πλαστογράφηση» ψηφιακής υπογραφής – διαδικασία που θεωρείται υπολογιστικά ανέφικτη. Επιπλεόν, όταν ένα BLR δημοσιεύεται στο DHT υπόστρωμα και ένας κόμβος το προσπελαύνει (π.χ. κατά τη διάρκεια βαθμολόγησης μίας εισερχόμενης αίτησης), τότε ο τελευταίος μπορεί να επιλέξει να ελέγξει την εγκυρότητά του, με βάση μία συνάρτηση πιθανότητας $\mathcal{P}(\mathcal{V})$. Στατιστικά, σε ένα μεγάλης κλίμακας σύστημα, η πιθανότητα να μην ελεγχθεί η εγκυρότητα μίας Μαρτυρία Κατηγορίας τείνει στο μηδέν με την πάροδο του χρόνου.

Ο έλεγχος της εγκυρότητας ενός BLR γίνεται επικοινωνώντας με τους εμπλεκόμενους κόμβους. Οι κόμβοι στο SeAl μπορούν, με τη βοήθεια του SVL, να αποδείξουν την εγκυρότητα ή μη ενός BLR, παρουσιάζοντας τις εισαγωγές από τις λίστες F_d και F_o τους, οι οποίες υποδεικνύουν την κατάσταση αποπληρωμής της αντίστοιχης χά-

[1]Υπενθυμίζουμε ότι οι κόμβοι στο SeAl ταυτοποιούνται από το secure hash του κοινού κλειδιού τους και ότι για τις ψηφιακές υπογραφές χρησιμοποιούνται τα προσωπικά κλειδιά τους.

ρης. Στην περίπτωση που διαπιστωθεί κάποια παρέκκλιση από τα πρωτόκολλα του συστήματος (π.χ. κάποιος κόμβος παρουσιάζει ένα BLR ως έγκυρο ενώ ο «κατηγορούμενος» κόμβος έχει απόδειξη αποπληρωμής του), οι εμπλεκόμενοι κόμβοι μπορούν να προχωρήσουν στην δημιουργία Αποδείξεων Κατηγορίας για τον ψευδόμενο κόμβο. Συνεπώς, σύν τω χρόνω, τόσο οι ιδιοτελείς όσο και οι κακόβουλοι κόμβοι εντοπίζονται και σημειώνονται ώς τέτοιοι.

Προφανώς, Μαρτυρίες Κατηγορίας που κρίνονται άκυρες και από τους δύο εμπλεκόμενους κόμβους, δεν οδηγούν σε δημιουργία περαιτέρω Μαρτυριών Κατηγορίας. Τέτοια άκυρα BLR μπορούν είτε να διαγράφονται επί τόπου, ή να παραμένουν στο σύστημα – για ένα προκαθορισμένο χρονικό διάστημα – ως ένδειξη της πρότερης συμπεριφοράς των αντίστοιχων κόμβων.

Το παραπάνω σχήμα εισαγάγει άλλη μία ρύθμιση – αυτή τη φορά σχετική με την αυστηρότητα της επαλήθευσης των Μαρτυριών Κατηγορίας – η οποία παρουσιάζει μία «διελκυνστίδα» ανάμεσα στην απόδοση των κόμβων (απαιτήσεις σε υπολογιστική ισχύ, αποθηκευτικό χώρο και εύρως ζώνης δικτύου) και την ταχύτητα αντίδρασης του συστήματος σε ιδιοτελείς και κακόβουλες συμπεριφορές.

4.3 SVL και Μαρτυρίες Προσφοράς

Ας θεωρήσουμε και πάλι το κλασσικό σενάριο στο οποίο ο κόμβος n_1 προσπελάυνει τον πόρο r ο οποίος διαμοιράζεται από τον κόμβο n_2. Όταν στο σύστημα χρησιμοποιείται το SVL, ο κόμβος n_1 και πάλι επιλέγει ένα υποσύνολο W_m των Μαρτυριών Προσφοράς του με άθροισμα αξιών ίσο ή μικρότερο με την τιμή κατωφλίου που του ορίζει ο n_2, αλλά αυτή τη φορά προωθεί τις Μαρτυρίες αυτές στον n_2 μαζί με την αίτηση πρόσβασής του. Ο n_2 μπορεί τότε να επιλέξει είτε να υπολογίσει τη βαθμολογία της εισερχώμενης αίτησης όπως και στην περίπτωση του SAL, είτε να ελέγξει αν ο n_1 ψεύδεται ή όχι, ανακτώντας και ελέγχοντας την εγκυρότητα μίας ή περισσοτέρων από τις παρουσιαζόμενες Μαρτυρίες Προσφοράς (ακολουθώντας μία συνάρτηση

πιθανότητας $\mathcal{P}(\mathcal{C})$).

Ο υπολογισμός της θετικής βαθμολόγησης s_w γίνεται ως εξής: ας υποθέσουμε ότι $W'_m \subset W_m$ Μαρτυρίες Προσφοράς ανακτώνται και επαληθεύονται. Τότε η εισερχόμενη αίτηση λαμβάνει μία αρχική θετική βαθμολογία s_w ίση με το άθροισμα των αξιών των Μαρτυριών Προσφοράς στο σύνολο W_m, όπου η αξία των Μαρτυριών στο W'_m υπολογίζεται κανονικά, ενώ για τις υπόλοιπες $W_m - W'_m$ Μαρτυρίες θεωρούμε ότι ο χρονος δημιουργίας τους ισουται με τον τρέχοντα χρόνο (και άρα η αξία τους ισούται με το μέγεθος του πόρου που αφορούν).

Στη συνέχεια, ο κόμβος n_2 ερευνά το DHT υπόστρωμα για Μαρτυρίες Κατηγοριών εναντίον του n_1· ο n_2 ζητά από τον κόμβο που αποθηκεύει τις Μαρτυρίες αυτές να του επιστρέψει τόσες Μαρτυρίες ώστε το άθροισμα των αξιών τους να είναι μικρότερο ή ίσο από την τιμή κατωφλίου που έχει θέσει τοπικά ο διαχειριστής του n_2. Η Μαρτυρίες αυτές επαληθεύονται, ακολουθώντας μία συνάρτηση πιθανότητας $\mathcal{P}(\mathcal{V})$.

Έστω ότι B_m είναι το σύνολο των BLR που επιστρέφονται από την παραπάνω αναζήτηση και ότι $B'_m \subset B_m$ είναι το σύνολο των Μαρτυριών αυτών που επαληθεύτηκαν. Τότε, υπολογίζεται και μία αρνητική βαθμολογία s_b ίση με το άθροισμα των αξιών των Μαρτυριών στο σύνολο B_m, όπου η αξία των Μαρτυριών στο B'_m υπολογίζεται κανονικά, ενώ για τις υπόλοιπες $B_m - B'_m$ Μαρτυρίες θεωρούμε ότι ο χρονος δημιουργίας τους ισουται με τον τρέχοντα χρόνο (και άρα η αξία τους ισούται με το μέγεθος του πόρου που αφορούν). Τελικά η αίτηση βαθμολογείται και πάλι με $s_w - s_b - r.size$ και εισάγεται στην ουρά αναμονής.

Το παραπάνω πιθανοτικό σχήμα εισαγάγει μία ακόμα διελκυνστίδα ανάμεσα στον αριθμό των προσβάσεων στο δίκτυο και των καταναλισκόμενων πόρων, και στην πιθανότητα να ληφθεί υπόψιν μία ψευδής Μαρτυρία. Αυτό δίνει την δυνατότητα στους διαχειριστές των κόμβων του δικτύου να επιλέξουν ουσιαστικά το πόσο εμπιστεύονται τα δεδομένα και τις Μαρτυρίες που τους στέλνουν οι υπόλοιποι κόμβοι.

Το σχήμα αυτό έχει δύο πολύ σημαντικές ιδιότητες:

1. φράσει το κόστος πρόσβασης στο δίκτυο που πληρώνει ένας κόμβος για να ανακτήσει Μαρτυρίες Προσφοράς και Κατηγοριών, και

2. αφαιρεί κάθε κίνητρο που θα είχε ένας κόμβος να κάνει μία Sybil επίθεση ή να συνεργαστεί με άλλους κόμβους του δικτύου για να αυξήση τεχνητά την βαθμολογία του και να βελτιώσει την συνολική του εικόνα, καθώς:

    - όσες (πιθανώς ψευδείς) Μαρτυρίες Προσφοράς κι αν συγκεντρώσει, μπορεί να βαθμολογηθεί το πολύ με την ίδια βαθμολογία που θα είχε αν ήταν νέος κόμβος, αν υπάρχει έστω και μία Μαρτυρία Κατηγορίας εναντίον του στο δίκτυο, ενώ
    - το μόνο που θα μπορούσε να κατορθώσει αλλάζοντας το ID του θα ήταν να χάσει ακόμα και τις όποιες Μαρτυρίες Προσφοράς έχει συσσωρεύσει.

Επιπλέον, αφού επαληθεύουμε (στατιστικά) τις Μαρτυρίες Προσφοράς και Κατηγορίας που ανακτούμε από το δίκτυο, δεν είναι πλέον αναγκαίο να ενημερώνονται οι κόμβοι που αποθηκεύουν τις Μαρτυρίες Κατηγορίας, όταν οι αντίστοιχες χάρες αποπληρωθούν· αν κάποιος κόμβος προσπαθήσει να επαληθεύσει την εγκυρότητα μίας τέτοιας Μαρτυρίας ο έλεγχος θα αποτύχει και η Μαρτυρία θα διαγραφεί από το σύστημα σιωπηρά, όπως καθορίζεται από το σχήμα γήρανσης των Μαρτυριών. Να υπενθυμίσουμε ότι ένας τέτοιος αποτυχημένος έλεγχος δεν οδηγεί στη δημιουργία νέων Μαρτυριών Κατηγορίας, αφού και τα δύο εμπλεκόμενα μέλη συμφωνούν για την μη εγκυρότητα της Μαρτυρίας.

4.4 Πρωτόκολλο μεταφοράς αρχείων

Το πρωτόκολλο μεταφοράς αρχείων/διαμοιρασμού πόρων παίζει βασικό ρόλο στη σωστή και αποδοτική λειτουργία του μηχανισμού SVL. Το πρωτόκολλο παρουσιάζεται υπό μορφή ψευδοκώδικα στο σχήμα 5.

Αλγόριθμος 5 Αλγόριθμος μεταφοράς αρχείων

Ο αλγόριθμος εκτελείται στον κόμβο $m.n_{server}$, εκτός κι αν αναφέρεται διαφορετικά.

Require:

$send(msg, node\ ID)$: Send msg to node with given ID.
$\mathcal{E}_k(\alpha)$: Encrypt α using key k.
$\mathcal{S}_k(\alpha)$: Sign α using key k.

process(Msg m **)**

1: Generate k_1, k_2 = random symmetric-cipher keys;
2: $r_e = \mathcal{E}_{k_1}(m.r)$; $k_1' = \mathcal{E}_{k_2}(k_1)$;
3: $send(\{r_e, k_1'\}, m.n_{client}.id)$;
4: $m.n_{client}$:
  4.1: construct $TR' = \{m.n_{server}.id, m.n_{client}.id, m.r.id, t\}$;
  4.2: $TR'_s = \mathcal{S}_{m.n_{client}.k_s}(TR')$;
  4.3: $send(TR'_s, m.n_{server}.id)$;
5: Verify the signature in TR'_s;
6: $TR_s = \mathcal{S}_{m.n_{server}.k_s}(TR'_s)$;
7: $send(\{TR_s, \mathcal{E}_{m.n_{client}.k_p}(k_2), m.n_{client}.id\})$;
8: $m.n_{client}$: recover k_2 and k_1 and decrypt $m.r$;
9: $F_d.add(TR_s)$; $m.n_{client}$: $F_d.add(TR_s)$;

Εν ολίγοις, ας υποθέσουμε και πάλι ότι ό κόμβος n_1 ($m.n_{client}$) προσπελαύνει το αρχείο r ($m.r$) που διαμοιράζεται από τον κόμβο n_2 ($m.n_{server}$). Θεωρούμε ότι η αίτηση πρόσβασης έχει προστεθεί κανονικά στην ουρά αναμονής του n_2 και έστω ότι φτάνει η χρονική στιμή που αρχίζει η εξυπηρέτησή της.

Τότε, ο κόμβος n_2 παράγει δύο τυχαία κλειδιά – k_1 και k_2 – συμμετρικού κρυπτογραφικού αλγορίθμου, και χρησιμοποιεί το k_1 για να κρυπτογραφήσει το διαμοιραζόμενο αρχείο r. Στη συνέχεια, χρησιμοποιεί το k_2 για να κρυπτογρφήσει το k_1 και στέλνει στον κόμβο n_1 τόσο το κρυπτογραφημένο αρχείο, όσο και το κρυπτογραφημένο k_1.

Βασιζόμενος στα δεδομένα που έχει ως τώρα, ο n_1 κατασκευάζει μία αρχική Απόδειξη Συναλλαγής, την υπογράφει και τη στέλνει στον n_2. Ο n_2 επαληθεύει την εγκυρότητα της υπογραφής στην Απόδειξη Συναλλαγής και την υπογράφει και ο ίδιος. Κατόπιν, κρυπτογραφεί το k_2 χρησιμοποιώντας το κοινό κλειδί του n_1 ($n_1.k_p$), και

στέλνει το κρυπτογραφημένο k_2 και την υπογεγραμμένη Απόδειξη Συναλλαγής στον n_1.

Ο n_1 επαληθεύει την ορθότητα της υπογραφής στην Απόδειξη Συναλλαγής, χρησιμοποιεί το μυστικό του κλειδί για να αποκρυπτογραφήσει το k_2 και στη συνέχεια αποκρυπτογραφεί το k_1 και κατόπιν το αρχείο r. Τέλος, τόσο ο n_1 όσο και ο n_2 προσθέτουν την Απόδειξη Συναλλαγής τους στις αντίστοιχες λίστες χαρών τους. Τα κλειδιά k_1 και k_2 δε χρειάζονται μετά το πέρας της συναλλαγής και άρα μπορούν να διαγραφούν ασφαλώς· το ζεύγος κλειδιών ($n_i.k_{\{s,p\}}$) που έχει κάθε κόμβος αρκεί για να επαληθεύσει ή να διαψεύσει Αποδείξεις Συναλλαγών και Μαρτυρίες.

Στα σύγχρονα δίκτυα ομοτίμων διαμορασμού αρχείων, ο πλέον ανεπαρκής πόρος και ο βασικός ανασχετικός παράγοντας είναι το εύρος ζώνης του δικτύου. Θα μπορούσε κάποιος να θεωρήσει ότι ο n_2 θα μπορούσε να διακόψει την εκτέλεση του πρωτοκόλλου μεταφοράς αρχείων μετά το βήμα 6, καταλήγοντας να έχει στην κατοχή του μία καθ΄όλα έγκυρη Απόδειξη Συναλλαγής, ενώ ο n_1 θα είχε μία άχρηστη ακολουθία από ψευδοτυχαία bits. Το βασικό σκεπτικό πίσω από το παραπάνω πρωτόκολλο είναι ότι, αφού ένας κόμβος (ο n_2 στην περίπτωσή μας) έχει ήδη καταναλώσει εύρος ζώνης δικτύου για να στείλει το κρυπτογραφημένο αρχείο (βήμα 3), έχει ήδη προσφέρει στην υπόλοιπη P2P κοινότητα. Το να ακολουθήσει μέχρι τέλους το πρωτόκολλο και να αποστείλει και την Απόδειξη Συναλλαγής μαζί με το δεύτερο κλειδί στον κόμβο-παραλήπτη (τον n_1 στο παράδειγμά μας), είναι προς δικό του όφελος και απλά βελτιώνει τη θετική εικόνα του κόμβου (αφού μία Απόδειξη Συναλλαγής που δε μπορεί να επαληθευθεί είναι στην ουσία άχρηστη).

4.5 Το σύστημα SeAl στην πράξη

Το σύστημα SeAl ασχολείται προτίστως με τους ιδιοτελείς χρήστες – δηλαδή με όσους εμπίπτουν στις κατηγορίες των «τσιγγούνηδων» (ενότητα 1.2.1) και των «πλεονεκτών» (ενότητα 1.2.1). Οι μηχανισμοί επίβλεψης/καταγραφής και επαλήθευσης έχουν

σχεδιαστεί έτσι ώστε τέτοιοι χρήστες να εντοπίζονται και να απομονώνονται, ενώ οι αλτρουιστικοί χρήστες ανταμοίβονται. Επιπλέον, ορισμένες κατηγορίες κόμβων οι οποίοι δρούν κακόβουλα στα πλαίσια του SeAl, αποκαλύπτονται και τιμωρούνται αυστηρά.

Προφανώς, ένας κόμβος ο οποίος δεν προσφέρει τίποτα στην P2P κοινότητα, δε θα συγκεντρώσει ποτέ χάρες. Ένας τέτοιος κόμβος θα έχει πάντα το μικρότερο δυνατό επίπεδο αλτρουισμού και άρα πάντα θα εξυπηρετείται τελευταίος, υπό περιορισμούς (στο εύρος ζώνης δικτύου ή στον αριθμό των παράλληλων συνδέσεων προς έναν κόμβο), ή δε θα εξυπηρετείται καθόλου, ανάλογα με τις τοπικές ρυθμίσεις των κόμβων με τους οποίους αλληλεπιδρά. Παρομοίως, οι κόμβοι δε μπορούν να αποκτήσουν πρόσβαση σε περισσότερους πόρους απ΄όσους τους αναλογούν, σύμφωνα με την συνεισφορά τους στην κοινότητα. Πιστεύουμε ότι τα παραπάνω δίνουν κίνητρα στους κόμβους ώστε να δρουν αλτρουιστικά.

Επιθέσεις ενάντια στο μηχανισμό ταυτοποίησης των κόμβων θεωρούνται υπολογιστικά αδύνατες, εξαιτίας της χρήσης ισχυρών κρυπτογραφικών διεργασιών (κρυπτογράφηση κοινού κλειδιού, ψηφιακές υπογραφές, κτλ.) Για παράδειγμα, επιθέσεις τύπου «pseudospoofing» – οι οποίες έγκεινται στην προσπάθεια ενός κόμβου να εισέλθει στο σύστημα χρησιμοποιώντας το αναγνωριστικό ταυτοποίησης ενός άλλου κόμβου – θα αποτύχουν σίγουρα αφού ο κόμβος αυτός δε μπορεί να αποδείξει γνώση του μυστικού κλειδιού που αντιστοιχεί στο κοινό κλειδί του οποίου το secure hash ισούται με το ID που προσπαθεί να καταλάβει.

Συνεργατικές επιθέσεις

Μία μάλλον δύσκολη κατηγορία «επιθέσεων» κατά του μηχανισμού επίβλεψης/κατραγραφής είναι η συνεργασία περισσοτέρων του ενός κόμβων, ώστε να ανταλλάσσουν ψεύτικες χάρες μεταξύ τους και να έχουν πάντα πολλές Μαρτυρίες Προσφοράς. Αυτό υποθετικά τους δίνει τη δυνατότητα να αποκτήσουν υψηλή βαθμολογία στη μετρική του επιπέδου αλτρουισμού. Επιθέσεις που μπορούν να θεωρηθούν ως

υποπεριπτώσεις της κατηγορίας αυτής είναι και οι επιθέσεις «shilling» και «Sybil»· η βάση των επιθέσεων αυτών είναι η δυνατότητα που έχουν οι κόμβοι σε ένα ευρέως κατανεμημένο σύστημα (όπως τα σύγχρονα δίκτυα ομοτίμων διαμοιρασμού αρχείων) να εγγράφονται με πολλά διαφορετικά ψευδώνυμα – στην πρώτη περίπτωση έχοντας παράλληλα όλα τα ψευδώνυμα ενεργά (διατηρώντας δηλαδή παράλληλα πολλούς ιδεατούς κόμβους ενεργούς στο σύστημα), ενώ στην δεύτερη χρησιμοποιούν μόνο ένα (διαφορετικό) ψευδώνυμο (και ιδεατό κόμβο) σε κάθε εισαγωγή τους στο σύστημα.

Διακρίνουμε δύο περιπτώσεις για την κατηγορία επιθέσεων αυτή:

- οι συνεργαζόμενοι/ιδεατοί κόμβοι συνεισφέρουν λίγους ή καθόλου πόρους στην κοινότητα και παραμένουν συνδεδεμένοι για μικρά χρονικά διαστήματα (κλασσικό πρότυπο συμπεριφοράς για τους λεγόμενους free-riders), και
- οι συνεργαζόμενοι/ιδεατοί κόμβοι συνεισφέρουν λίγους ή καθόλου πόρους στην κοινότητα και παραμένουν συνδεδεμένοι για μεγάλα χρονικά διαστήματα.

Η περίπτωση οι συνεργαζόμενοι/ιδεατοί κόμβοι να συνεισφέρουν πολλούς ή όλους τους πόρους τους στην κοινότητα δεν μας απασχολεί, αφού τότε οι κόμβοι αυτοί δε δρουν ιδιοτελώς.

Στην πρώτη από τις παραπάνω περιπτώσεις, οι κόμβοι δεν κερδίζουν τίποτα συγκεντρώνοντας ψεύτικες Μαρτυρίες Προσφοράς, αφού αυτές δε μπορούν να επαληθευθούν (εξαιτίας του ότι οι κόμβοι δεν μένουν συνδεδεμένοι για μεγάλα χρονικά διαστήματα). Ακόμα, όμως, και στη δεύτερη περίπτωση, όπου οι ψεύτικες Μαρτυρίες Προσφοράς μπορούν να επαληθευθούν, οι συνεργαζόμενοι κόμβοι μπορούν να αποκτήσουν τόσο καλή φήμη όση τους επιτρέπουν (i) οι κόμβοι με τους οποίους αλληλεπιδρούν (μέσω των τιμών κατωφλίου που θέτουν για το άθροισμα των αξιών των Μαρτυριών Προσφοράς που αυτοί μπορούν να παρουσιάσουν), και (ii) το άθροισμα των αξιών των Μαρτυριών Κατηγορίας που έχουν δημοσιευτεί εναντίον τους. Αυτό σημαίνει ότι όσο ένας κόμβος δρα ιδιοτελώς και δεν ανταποδίδει της χάρες (μέσω του μηχανισμού ανακατεύθυνσης αιτήσεων του SAL), θα έχει συσσωρευμένες Μαρτυρίες

Κατηγορίας εναντίον του, οπότε το κέρδος του από τις ψεύτικες Μαρτυρίες Προσφοράς είναι μηδενικό (μπορούμε να πούμε μάλιστα ότι ο κόμβος ζημιώνεται κιόλας, αφού δεσμεύει αποθηκευτικό χώρο στον τοπικό του δίσκο/μνήμη για να διατηρεί τις ψεύτικες αυτές Μαρτυρίες).

Εν κατακλείδι, το πρόβλημα της συνεργασίας ιδιοτελών ή κακόβουλων κόμβων σε ένα περιβάλλον όπως αυτό των δικτύων ομοτίμων παραμένει ανοιχτό ερευνητικό πρόβλημα, ακόμα και για συστήματα τα οποία ασχολούνται αποκλειστικά και μόνο με την διαχείριση σχέσεων εμπιστοσύνης (trust relationships) ανάμεσα στους συμμετέχοντές τους. Μπορούμε να αποδείξουμε (με αναγωγή στο πρόβλημα των επιθέσεων Sybil[21]) ότι συνεργατικές επιθέσεις σε ευρέως κατανεμημένα συστήματα *δεν* μπορούν να αντιμετωπιστούν αποδοτικά ή ρεαλιστικά, εκτός και θεωρήσουμε ότι υπάρχει κάποια κεντρικοποιημένη αρχή επίβλεψης/επιβολής (π.χ. ένα σύστημα όπως το Advogato[7] στο οποίο εφαρμόζονται αλγοριθμικές τεχνικές ανάλυσης ακμών γραφημάτων (graph link analysis) για να δωθούν εγγυήσεις ότι συνεργαζόμενοι κόμβοι δε μπορούν να επιτύχουν υψηλές βαθμολογίες εκτός του συνόλου των κόμβων αυτών) ή κάνουμε μη ρεαλιστικές υποθέσεις για το βαθμό συντονισμού των κόμβων.

Ακόμα και αν εφαρμόσουμε κάποιο πρωτόκολλο «Βυζαντινής συμφωνίας» (Byzantine aggreement) – οπότε μπορούμε να ανεχθούμε $\frac{1}{3}$ του συνολικού πληθυσμού να συνεργάζονται – κανείς δε μπορεί να εμποδίσει έναν κόμβο απ΄το να εγγραφεί με περισσότερα από τόσα ψευδώνυμα! Πιστεύουμε ότι το πρόβλημα αυτό είναι από τα δυσκολότερα που μπορεί να αντιμετωπίσει κανείς στο πεδίο των κατανεμημένων συστημάτων. Εντούτοις, το σύστημα SeAl περιορίζει σε κάποιο βαθμό τις συνέπειες των επιθέσεων αυτών. Περαιτέρω βελτιώσεις στον τομέα αυτό φαίνεται να μπορούν να επιτευχθούν μόνο με την χρησιμοποίηση κάποιου συστήματος χρέωσης για την εγγραφή νέου χρήστη (οπότε κατά κάποιον τρόπο η επίθεση στρέφεται κατά του επιτηθέμενου, αν και σε οικονομικό επίπεδο πλέον). Εν τέλει, η τμηματική δομή του SeAl μας επιτρέπει να χρησιμοποιήσουμε μία πιο αποδοτική και πλήρη λύση, όταν αυτή βρεθεί.

Κεφάλαιο 5

Προσομοιώσεις και αποτελέσματα μετρήσεων

Θα περιγράψουμε τώρα τις προσομοιώσεις και τις μετρήσεις που κάναμε στα πλαίσια της πειραματικής αξιολόγησης της απόδοσης του συστήματος SeAl. Θα αρχίσουμε περιγράφοντας το περιβάλλον στο οποίο έγιναν οι προσομοιώσεις, και θα συνεχίσουμε με την παρουσίαση και την ανάλυση των αποτελεσμάτων.

5.1 Περιβάλλον προσομοίωσης

Οι παράμετροι των προσομοιώσεων συνοψίζονται στον πίνακα 5.1. Υποθέτουμε, όπως έχουμε ήδη αναφέρει, ότι το σύστημα SeAl χρησιμοποιείται σε ένα περιβάλλον δικτύου ομοτίμων διαμοιρασμού (π.χ. μουσικών) αρχείων. Από πραγματικά στοιχεία, γνωρίζουμε ότι τα αρχεία αυτά (ειδικά σε μορφή MP3) έχουν μέγεθος σχεδόν ομοιόμορφα κατανεμημένο στο εύρος 3MBytes-10MBytes, με μέσο μέγεθος 6.5MBytes. Το δίκτυο ομοτίμων της προσομοίωσης αποτελείται από $|\mathcal{N}| = 2.048$ κόμβους, οι οποίοι κατανέμονται ως 90% (ή 70%) ιδιοτελείς κόμβους και 10% (ή 30%) αλτρουιστές κόμβους, με συνδέσεις στο δίκτυο εύρους ζώνης ομοιόμορφα κατανεμη-

μένου από $33,6$kbps (σύνδεση με απλό modem) έως 256kbps (σύνδεση με καλωδιακό (cable) modem ή τύπου DSL) για τους ιδιοτελείς και από 256kbps έως 2Mbps (σύνδεση T1) για τους αλτρουιστές. Προσομοιώσαμε και δίκτυα με περισσότερους κόμβους (με παρόμοια αποτελέσματα), αν και όχι με αντίστοιχο αριθμό αιτήσεων, λόγω περιορισμών στην διαθέσιμη επεξεργαστική ισχύ και κύρια μνήμη.

Η προσομοίωση αποτελείται από $1.000.000$ αιτήσεις προσπέλασης. Οι αιτήσεις αυτές φτάνουν στο σύστημα ακολουθώντας μία κατανομή Poisson, ειδικά κατασκευασμένη ώστε κάθε κόμβος να κάνει περίπου 5 αιτήσεις ανά μέρα χρόνου προσομοίωσης. Οι κόμβοι διαμοιράζονται 50.000 διαφορετικά αρχεία, κατανεμημένα και αντιγραμμένα στο δίκτυο ακολουθώντας μία κατανομή Zipf με παράμετρο $\alpha = 0.7$ και 1.2[70] (με 50.200 και 51.350 συνολικά αρχεία αντίστοιχα). Επιπλέον, τα αρχεία που ζητούνται κατά τις αιτήσεις, επιλέγονται από το σύνολο των διαμοιραζόμενων αρχείων ακολουθώντας μία χωριστή κατανομή Zipf με παράμετρο $\alpha = 0.7$ και 1.2 όπως και παραπάνω. Τα αποτελέσματα είναι παρόμοια και για τις δύο τιμές της παραμέτρου της Zipf, και ως εκ τούτου παραθέτουμε αποτελέσματα μόνο για την τιμή $\alpha = 1.2$ που αποτελεί και δυσκολότερη περίπτωση.

Η αναζήτηση ενός αρχείου, η εύρεση ενός κόμβου που το διαμοιράζεται, καθώς και η μεταφόρτωσή του θεωρούνται λειτουργίες που χρειάζονται μόνο 1 hop· αφ΄ενός το σύστημα SeAl δεν εμπλέκεται στην αναζήτηση αρχείων στο δίκτυο ομοτίμων, αφ΄εταίρου θεωρούμε ότι η μεταφορά των αρχείων γίνεται απευθείας από τον κόμβο-αποστολέα στον κόμβο-παραλήπτη του αρχείου. Σε πραγματικές εφαρμογές, η αναζήτηση μπορεί να γίνει με οποιονδήποτε από τους διαδεδομένους τρόπους (π.χ. μέσω κεντρικοποιημένης βάσης δεδομένων όπως στο σύστημα Napster, με τεχνικές βασισμένες σε «πλημμύρα μηνυμάτων» (message flooding) όπως στο Gnutella, με βελτιωμένες εκδόσεις αλγορίθμων «πλημμυρίσματος» όπως αυτές των [18] και [79], βασισμένες σε ημι-ιεραρχικές δομές (π.χ. Super-peers[78]), ή χρησιμοποιώντας οποιοδήποτε υπόστρωμα DHT, όταν το SeAl λειτουργεί πάνω από δομημένα δίκτυα ομοτίμων,

κτλ.)

Από την άλλη, όλες οι λειτουργίες του SeAl εκτελούνται πάνω από το DHT του μηχανισμού επίβλεψης/καταγραφής, οπότε απαιτούν $O(log(|\mathcal{N}|))$ hops. Για την περίπτωση του Chord αυτό μεταφράζεται σε ≈ 6 hops ($= \frac{1}{2} \times log(|\mathcal{N}|)$), αλλά εμείς θα θεωρήσουμε ότι χρειαζόμαστε 11 ($= log(|\mathcal{N}|)$) hops ανά πρόσβαση, ως σενάριο χειρότερης περίπτωσης. Επιπλέον, προκειμένου να μετρήσουμε το επιπλεόν κόστος σε αποθηκευτικό χώρο που επιφέρει το SeAl, ο μηχανισμός γήρανσης των Μαρτυριών ήταν απενεργοποιημένος κατά τη διάρκεια της προσομοίωσης. Όπως θα δούμε και στα αποτελέσματα, παρά τα αρχικά αυτά εμπόδια, τα συνολικά επιπλέον κόστη λειτουργίας του SeAl, τόσο όσον αφορά τη χρήση του δικτύου όσο και τον απαιτούμενο αποθηκευτικό χώρο, είναι αμελητέα. Αξίζει να τονίσουμε ότι, αν το SeAl λειτουργεί σε δίκτυο τύπου «αποθήκευσε-και-προώθησε» (store-and-forward), όπως τα FreeNet[16] και Achord[37], επιφέρει πολύ μικρότερο (κατά τάξεις μεγέθους) συνολικό επιπλέον κόστος προσβάσεων στο δίκτυο.

Το επίπεδο αλτρουισμού των κόμβων υπολογίζεται με βάση τον τύπο της διαφοράς των μεγεθών των λιστών των κόμβων ($n_i.A = |n_i.F_d| - |n_i.F_o|$). Οι πιθανότητες με τις οποίες ένας κόμβος αποφασίζει αν θα ανακατευθύνει μία αίτηση ισούνται με 0, 1 και $0,5$ όταν το επίπεδο αλτρουισμού του κόμβου είναι μικρότερο της κάτω τιμής κατωφλίου, μεγαλύτερο της άνω τιμής κατωφλίου, ή ανάμεσα στις δύο αυτές τιμές αντίστοιχα. Κάθε κόμβος μπορεί να αντιμετωπίσει σφάλμα στη λειτουργία του (και να βγει εκτός δικτύου) ή να αρνηθεί να εξυπηρετήσει μία αίτηση με πιθανότητα $0,2$ και να διαγράψει ή να πάψει να διαμοιράζεται ένα αρχείο με πιθανότητα $0,1$.

Πρωτού μία αίτηση εισαχθεί στην ουρά αναμονής ενός κόμβου, ο κόμβος-αποστολέας ενημερώνει τον κόμβο-παραλήπτη για τον εκτιμώμενο χρόνο αναμονής της αίτησής του. Επίσης, θυμίζουμε ότι ο κόμβος-παραλήπτης ελέγχει περιοδικά – κάθε 2 λεπτά χρόνου προσομοίωσης κατά τη διάρκεια των πειραμάτων αυτών – την θέση που έχει η αίτησή του (δηλ. τον τρέχοντα εκτιμώμενο χρόνο αναμονής) στην

ουρά αναμονής του κόμβου αποστολέα.

Καθώς ο μηχανισμός παροχής κινήτρων του SeAl βασίζεται στην ικανοποίηση του χρήστη από το δίκτυο (με την τοποθέτηση των αιτήσεων σε ουρές αναμονής και υποχρεώνοντας τους ιδιοτελείς χρήστες να περιμένουν περισσότερο ή να εξυπηρετούνται με τμήμα των διαθέσιμων πόρων), προσπαθήσαμε επίσης να μοντελοποιήσουμε και τη συμπεριφορά των χρηστών. Έτσι, κάθε χρήστης χαρακτηρίζεται και από τα ακόλουθα:

- $\mathcal{P}(\mathcal{C}_a)$: η πιθανότητα με την οποία ένας χρήστης διακόπτει μία συναλλαγή μεταφόρτωσης πρωτού αυτή ολοκληρωθεί επιτυχώς. Χρησιμοποιείται σε κάθε συναλλαγή μεταφορτωσης και η αρχική τιμή της είναι $0,1$. Η παράμετρος αυτή μοντελοποιεί το ενδεχόμενο ένας από τους δύο χρήστες να αντιμετωπίσει σφάλμα ή να βγει εκτός δικτύου κατά τη διάρκεια της μεταφόρτωσης.

- $\mathcal{P}(\mathcal{R}_{a,s})$: η πιθανότητα με την οποία ένας χρήστηες παραμένει ιδιοτελής/αλτρουιστής. Χρησιμοποιείται κάθε φορά που ένας χρήστης κάνει μία αίτηση πρόσβασης, για να μοντελοποιήσει την μεταβλητότητα στη συμπεριφορά των χρηστών. Στην περίπτωση αλτρουιστικών χρηστών, μπορούμε να σκεφτούμε την πιθανότητα $\mathcal{P}(\mathcal{R}_a)$ ώς ισοδύναμη με την πιθανότητα σφάλματος του κόμβου. Οι αρχικές τιμές για τις πιθανότητες αυτές είναι $0,8$ και $1,0$ αντίστοιχα (δηλαδή ένας αλτρουιστής χρήστης έχει 20% πιθανότητα να γίνει ιδιοτελής ή να αντιμετωπίσει πρόβλημα, ενώ ένας ιδιοτελής χρήστης δεν αλλάζει συμπεριφορά με βάση αυτή την πιθανότητα).

- $\mathcal{P}(\mathcal{E}_f)$: η πιθανότητα ένας χρήστης να διαγράψει ή να μή μοιραστεί ένα αρχείο που μόλις μεταφόρτωσε από άλλον χρήστη. Χρησιμοποιείται κάθε φορά που ένας κόμβος-παραλήπτης ολοκληρώνει επιτυχώς μία συναλλαγή μεταφόρτωσης με έναν άλλο κόμβο. Η αρχική τιμή για την παράμετρο αυτή είναι $0,1$.

Επιπλέον, κάθε χρήστης έχει ένα άνω όριο χρόνου αναμονής 20 λεπτών σε

χρόνο προσομοίωσης για να εξυπηρετηθεί μία αίτησή του. Αν σε κάποιον από τους περιοδικούς ελέγχους που κάνει για την θέση μιας αίτησής του ο χρήστης πάρει απάντηση μεγαλύτερη από το όριο αυτό των 20 λεπτών, αποφασίζει να ακυρώσει την αίτηση και να «βελτιώσει» τη συμπεριφορά του με πιθανότητα $\mathcal{P}(\mathcal{S}_d)$. Ο χρήστης «βελτιώνεται» αυξάνοντας το $\mathcal{P}(\mathcal{R}_a)$ του και μειώνοντας το $\mathcal{P}(\mathcal{R}_s)$ του, κατά μία ποσότητα SD. Οι τιμές που χρησιμοποιήθηκαν για τις δύο παραμέτρους αυτές ($\{\mathcal{P}(\mathcal{S}_d), SD\}$) ήταν αντίστοιχα: $\{0,0,\ N/A\}$ (περίπτωση *χωρίς ανάδραση - no feedback*), $\{0,5,\ 0,05\}$ (περίπτωση *μικρής ανάδρασης small feedback*) και $\{0,5,\ 0,2\}$ (περίπτωση *μέσης ανάδρασης medium feedback*).

5.2 Αποτελέσματα

Ακολουθεί η ανάλυση των αποτελεσμάτων των προσομοιώσεων αυτών. Θα αναφερθούμε χωριστά σε πορίσματα σχετικά με την ευστάθεια του συστήματος με και χωρίς το SeAl και σχετικά με το επιπλέον κόστος σε έυρος ζώνης δικτύου, αποθηκευτικό χώρο και χρόνο απόκρισης που επιφέρει η λειτουργία του SeAl.

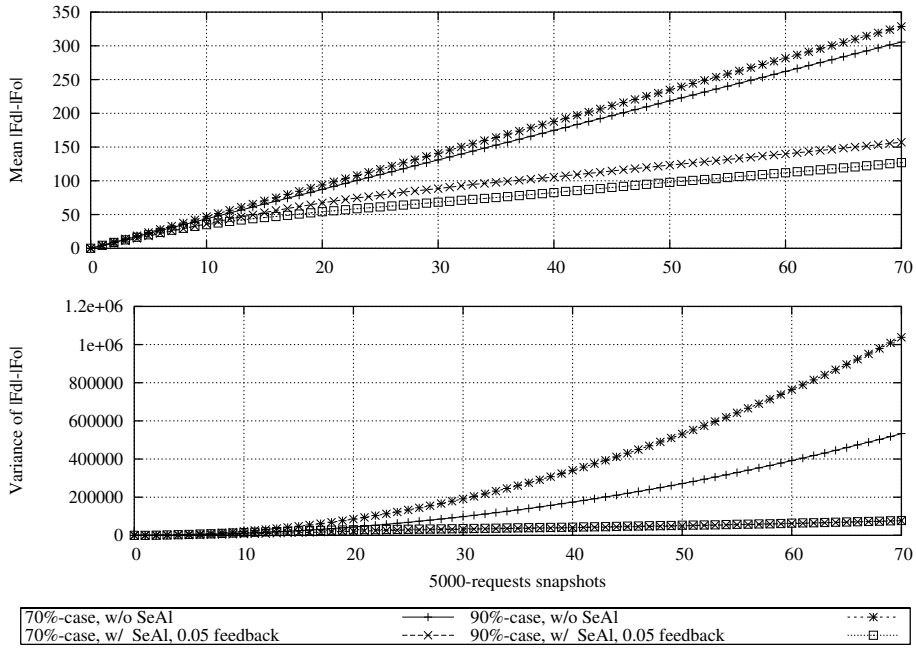
5.2.1 Ευστάθεια

Μετρήσαμε το μέσο επίπεδο αλτρουισμού στο σύστημα, χρησιμοποιώντας και τους δύο τύπους υπολογισμού (διαφορά και λόγος των μεγεθών των λιστών F_d και F_o των κόμβων). Στο σχήμα 5.1 έχουμε σχεδίασει τη μέση τιμή και τη διασπορά για την περίπτωση της διαφοράς των μεγεθών ($n_i.A = |F_d| - |F_o|$). Παρομοίως, στο σχήμα 5.2 έχουμε σχεδιάσει τον τετραγωνικό συντελεστή απόκλισης (square coefficient of variation - SCoV) για το μέσο επίπεδο αλτρουισμού των κόμβων του δικτύου, με τις επιμέρους βαθμολογίες να υπολογίζονται και με τους δύο τύπους. Όπως φαίνεται και από τα γραφήματα αυτά, οι προσομοίωσεις στις οποίες το σύστημα SeAl ήταν ενεργοποιημένο, κλιμακώνονται σαφώς καλύτερα με την πάροδο του χρόνου.

Στο σχήμα 5.3 έχουμε σχεδίασει το μέγεθος της λίστας F_d προς το μέγεθος

Παράμετρος	**Τιμή**
Πλήθος κόμβων ($\|\mathcal{N}\|$)	2.048 κόμβοι
Ποσοστό ιδιοτελών κόμβων	90% ή 70%
Ποσοστό αλτρουιστών κόμβων	10% ή 30%
Σύνδεση δικτύου ιδιοτελών κόμβων	$33,6$kbps-256kbps uniform
Σύνδεση δικτύου αλτρουιστών κόμβων	256kbps-2Mbps uniform
Μέγεθος αρχείων	3-10 MBytes uniform
Αριθμός μοναδικών αρχείων	50.000
Κατανομή δημοτικότητας αρχείων	Zipf, $\alpha = 0,7$ ή $1,2$
Συνολικός αρχικός αριθμός αρχείων	50.200 ή 51.350
Πλήθος αιτήσεων	$1.000.000$
Ακολουθία άφιξης αιτήσεων	Poisson, 5 αιτήσεις ανά κόμβο ανά ημέρα χρόνου προσομοίωσης
Κατανομή δημοτικότητας αιτήσεων	Zipf, $\alpha = 0,7$ ή $1,2$
Πιθανότητα ανακατεύθυνσης	$n_i.A > n_i.A_{max} \Rightarrow \frac{1}{2}$
Πιθ. σφάλματος κόμβου/δικτύου	$0,2$
Περίοδος ελέγχου εξυπηρέτησης	2 λεπτά χρόνου προσομοίωσης
Ανώτατο διάστημα αναμονής	20 λεπτά χρόνου προσομοίωσης
Πιθ. διακοπής μεταφόρτωσης ($\mathcal{P}(\mathcal{C}_a)$)	$0,1$
Πιθ. μη στιγμιαίας μεταβολής συμπεριφοράς ($\mathcal{P}(\mathcal{R}_{a,s})$)	αλτρουιστής κόμβος $\Rightarrow 0,8$ ιδιοτελής κόμβος $\Rightarrow 0,0$
Πιθ. διαγραφής αρχείου ($\mathcal{P}(\mathcal{E}_f)$)	$0,1$
Πιθ. μόνιμης μεταβολής συμπεριφοράς ($\mathcal{P}(\mathcal{S}_d)$) και μέγεθος μεταβολής SD ($\{\mathcal{P}(\mathcal{S}_d), SD\}$)	$\{0,0,\ N/A\}$ (χωρίς ανάδραση) $\{0,5,\ 0,05\}$ (μικρή ανάδραση) $\{0,5,\ 0,2\}$ (μεσαία ανάδραση)

Πίνακας 5.1: Βασικές παράμετροι προσομοίωσης

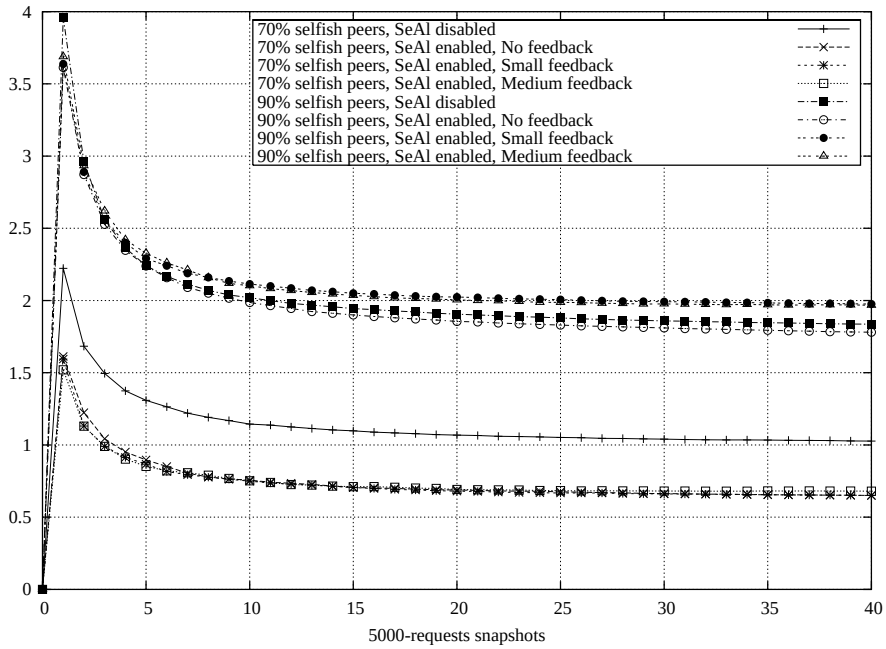


Σχήμα 5.1: Μεταβολή μέση τιμής και διασποράς του επιπέδου αλτρουισμού των κόμβων

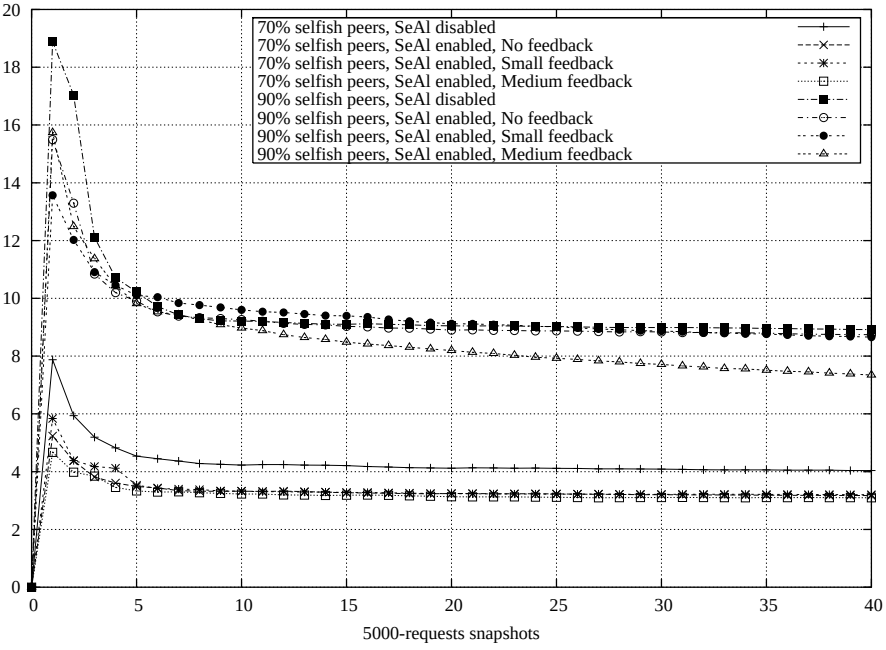
της λίστας F_o και για τους 2048 κόμβους στο δίκτυο, μετά το πέρας και των $1.000.000$ αιτήσεων, για την περίπτωση στην οποία 90% των κόμβων του δικτύου είναι ιδιοτελείς και με το SeAl να χρησιμοποιεί το μοντέλο μικρής ανάδρασης. Το σχήμα αυτό μπορεί να μας βοηθήσει να καταλάβουμε τα πραγματικά πλεονεκτήματα του SeAl. Αν παρατηρήσουμε τα δύο άνω υπογραφήματα, θα δούμε ότι οι κόμβοι χωρίζονται σε δύο κύριες ομάδες (clusters)· τους αλτρουιστές, οι οποίοι έχουν μεγαλύτερο $|F_d|$ από $|F_o|$ και άρα καταλαμβάνουν το άνω τμήμα των γραφημάτων, και τους ιδιοτελείς, οι οποίοι αντίστοιχα καταλαμβάνουν το κάτω τμήμα των γραφημάτων.

Η επίδραση του SeAl εμφανίζεται σε δύο τομείς:

1. η ομάδα σημείων που αντιστοιχεί στους αλτρουιστές είναι πιο πυκνή στο μέσο της – άλλη μία απόδειξη της βελτίωσης στην ευστάθεια τους συστήματος που χρησιμοποιεί το SeAl έναντι του απλού δικτύου ομοτίμων (χωρίς το SeAl), και

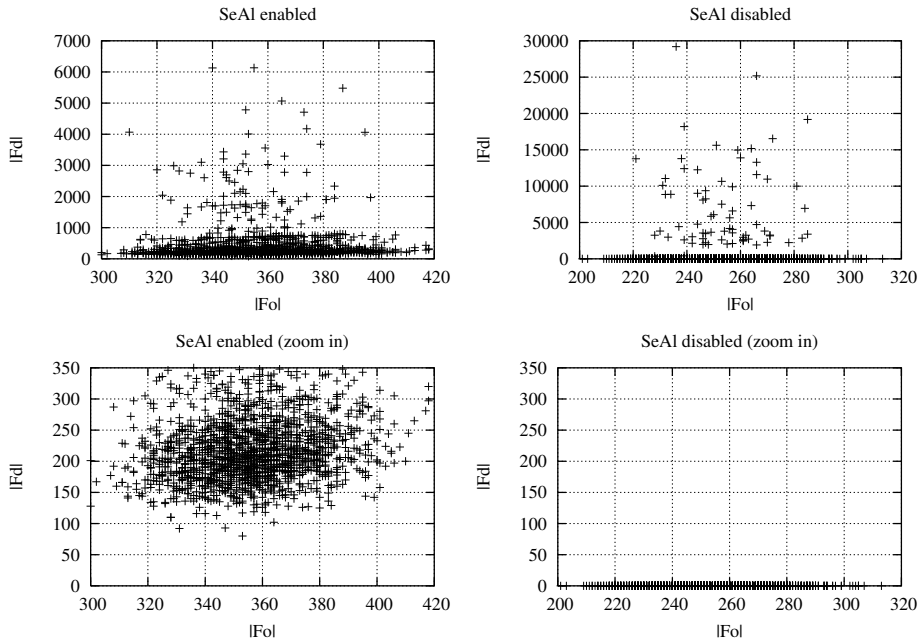


(α') Τετραγωνικός συντελεστής απόκλισης του μέσου $|F_d| - |F_o|$



(β') Τετραγωνικός συντελεστής απόκλισης του μέσου $\frac{|F_d|}{|F_o|}$

Σχήμα 5.2: Τετραγωνικοί συντελεστές απόκλισης του μέσου επιπέδου αλτρουισμού των κόμβων

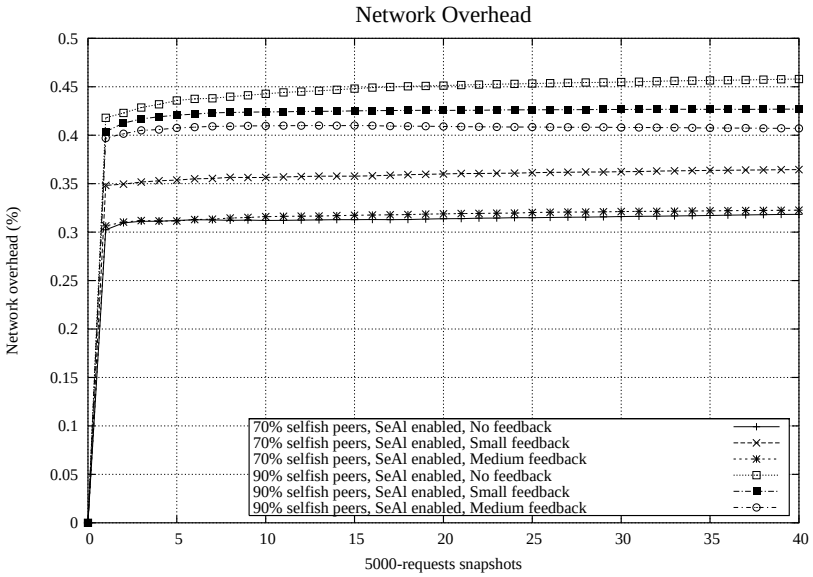


Σχήμα 5.3: $|Fd|$ προς $|Fo|$ (90% ιδιοτελείς κόμβοι)

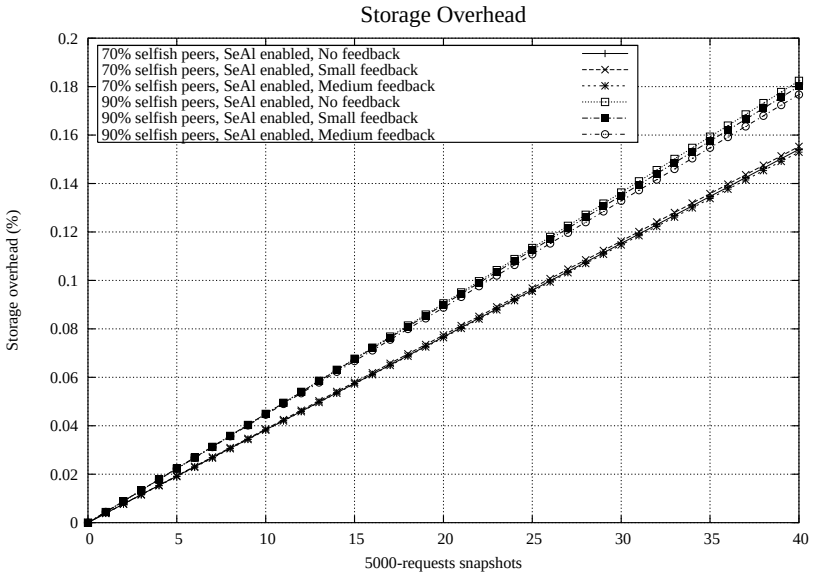
2. αν επικεντρωθούμε στην ομάδα σημείων που αντιστοιχεί στους ιδιοτελείς κόμβους (κάτω υπογραφήματα), θα δούμε ότι, ενώ στο απλό σύστημα όλοι οι free-riders έχουν μηδενικού μεγέθους F_d λίστες, με ενεργοποιημένο το SeAl ένα σημαντικό ποσοστό από τους κόμβους αυτούς έχει μετατοπιστεί πιο κοντά προς το σημείο ισορροπίας του συστήματος ($|n_i.F_d| = |n_i.F_o|$).

5.2.2 Επιπλέον κόστος

Το επιπλέον κόστος σε απαιτήσεις εύρους ζώνης δικτύου για τη λειτουργία του SeAl είναι αμελητέο, με μέση τιμή μόλις στο $0,4\%$ της συνολικής κίνησης (σχήμα 5.4(α΄)). Η κατάσταση είναι ακόμα καλύτερη όσον αφορά τις απαιτήσεις για αποθηκευτικό χώρο· υπενθυμίζουμε ότι στις προσομοιώσεις αυτές δεν είναι σε λειτουργία ο μηχανισμός γήρανσης των Μαρτυριών Κατηγορίας και Προσφοράς. Παρά το εμπόδιο αυτό, το σύστημα SeAl απαίτησε μόλις 0.53% επιπλέον αποθηκευτικό χώρο (περί-



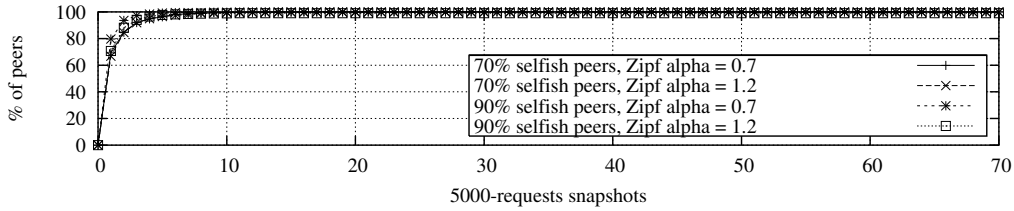
(α') Επιπλέον κόστος σε εύρος ζώνης δικτύου



(β') Επιπλέον κόστος σε αποθηκευτικό χώρο

Σχήμα 5.4: Επιπλέον κόστη σε εύρος ζώνης δικτύου και αποθηκευτικό χώρο από τη λειτουργία του SeAl

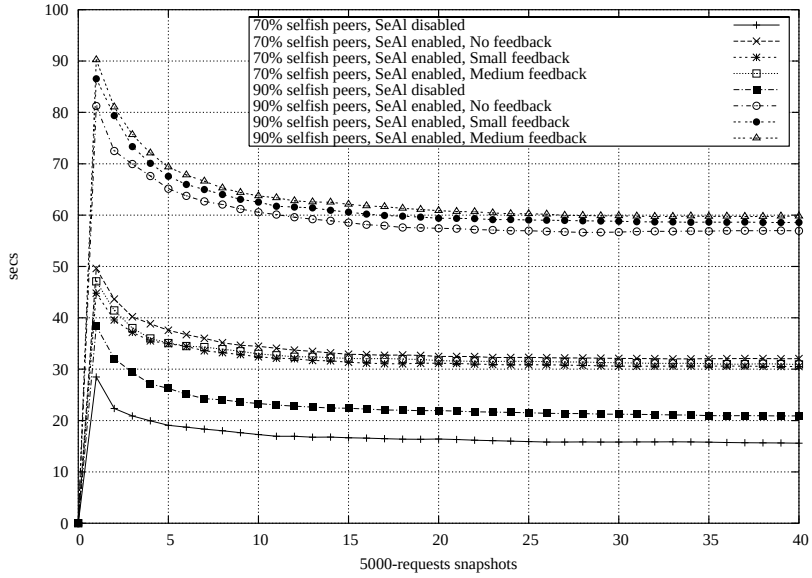
που 780kBytes ανά κόμβο) ύστερα από $1.000.000$ αιτήσεις (σχήμα 5.4(β')), ή περίπου $0,8$Bytes ανά κόμβο ανά αίτηση κατά μέσο όρο! Επίσης, πάνω από το 95% των κόμβων έχουν καταγραφεί τουλάχιστον μία φορά απο τον μηχανισμό επίβλεψης/καταγραφής, ύστερα από τις πρώτες $10.000 - 15.000$ αιτήσεις (σχήμα 5.5) – απόδειξη του ότι το σύστημα SeAl φτάνει γρήγορα στους στόχους του.



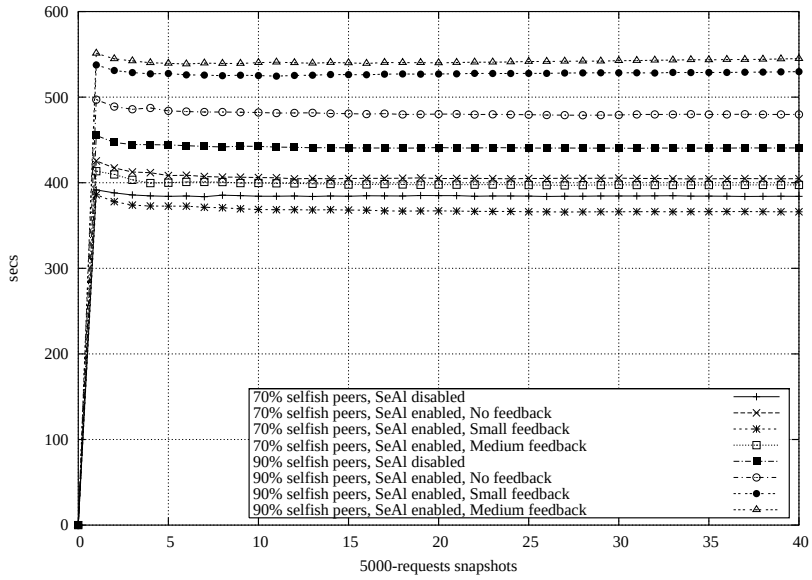
Σχήμα 5.5: Αποδοτικότητα του μηχανισμού επίβλεψης/καταγραφής

Το επιπλέον κόστος όσον αφορά το μέσο χρόνο εξυπηρέτησης των αιτήσεων κυμαίνεται σε ικανοποιητικά επίπεδα (σχήμα 5.6)· σε ένα σύστημα με την πλειοψηφία των κόμβων να δρουν ιδιοτελώς[6], παρουσίαζεται το φαινόμενο λίγοι κόμβοι (αλτρουιστές) να έχουν να εξυπηρετήσουν έναν ολοένα αυξανόμενο αριθμό αιτήσεων από τους υπόλοιπους (ιδιοτελείς και μη) κόμβους. Σε συνδυασμό με το μηχανισμό βαθμολόγησης των αιτήσεων και τις ουρές αναμονής του SeAl, το παραπάνω μπορεί να οδηγήσει σε αύξηση του μέσου χρόνου εξυπηρέτησης των αιτήσεων. Μία άλλη παράμετρος που αυξάνει το μέσο χρόνο απόκρισης, είναι και οι ανακατευθύνσεις που κάνουν οι κόμβοι στο SeAl· λόγω του φαινομένου της ύπαρξης πολλών free-riders στο σύστημα, ένα μεγάλο μέρος των αιτήσεων θα ανακατευθύνονται αρχικά σε ιδιοτελείς κόμβους ή κόμβους εκτός δικτύου πρωτού βρεθεί ένας κόμβος ο οποίος θα εξυπηρετήσει την ανακατευθυνόμενη αίτηση.

Ένα μάλλον ενδιαφέρον φαινόμενο εμφανίζεται στο σχήμα 5.7 για την περίπτωση στην οποία 90% των κόμβων είναι ιδιοτελείς (οι χρόνοι είναι σε δευτερόλεπτα χρόνου προσομοίωσης). Στο σχήμα αυτό, το κάτω υπογράφημα δείχνει το χρόνο που απαιτείται μέχρι να βρεθεί ένας κόμβος που θα εξυπηρετήσει μια νέα αίτηση, το με-



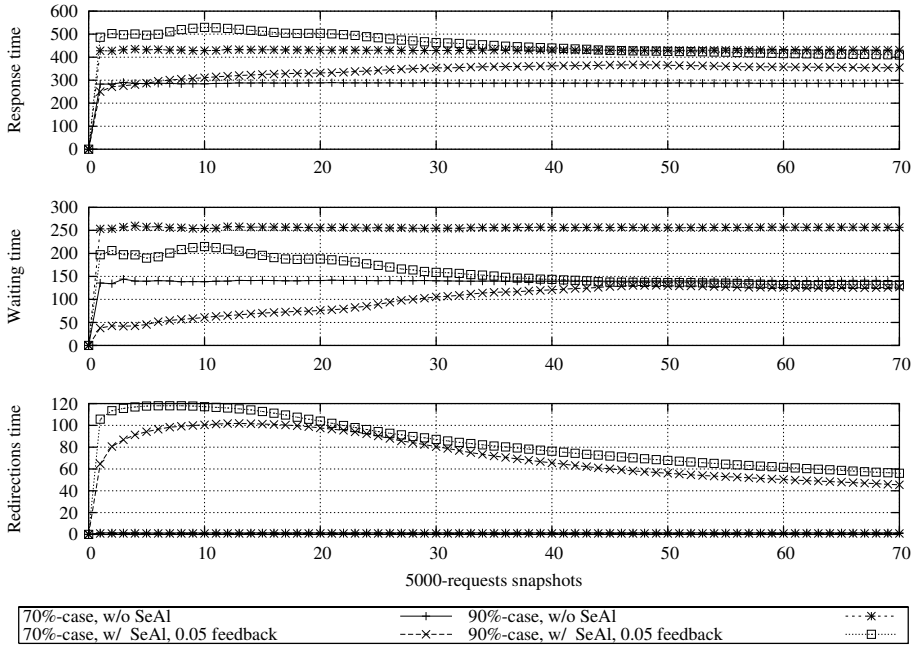
(α') Χρόνος απόκρισης



(β') Συνολικός χρόνος εξυπηρέτησης

Σχήμα 5.6: Επιπλέον κόστος χρόνου

σαίο υπογράφημα δείχνει το χρόνο που καταναλώνεται από τη στιμή που η αίτηση θα εισαχθεί στην ουρά αναμονής ενός κόμβου, και το άνω υπογράφημα δείχνει το συνολικό χρόνο απόκρισης. Υπενθυμίζουμε ότι θεωρούμε ότι βρίσκουμε έναν κόμβο ο οποίος διαμοιράζεται ένα συγκεκριμένο αρχείο σε 1 hop. Πέραν τούτου, το κλασσικό δίκτυο ομοτίμων (χωρίς το SeAl) δεν έχει να πληρώσει κάποιο επιπλέον κόστος (όπως φαίνεται και από την σχεδόν μηδενική καμπύλη του στο κάτω υπογράφημα), σε αντίθεση με το SeAl το οποίο χάνει κατά μέσο όρο $50''$ εξαιτίας αποτυχημένων ανακατευθύνσεων. Παρατηρούμε όμως στο υπογράφημα του συνολικού χρόνου απόκρισης ότι το SeAl καταφέρνει να καλύψει την απόσταση αυτή, με χρόνους απόκρισης $20''$ χαμηλότερους από το κλασσικό σύστημα κατά μέσο όρο.



Σχήμα 5.7: Χρόνοι ανακατεύθυνσης, αναμονής, και απόκρισης

Αυτό μπορεί να εξηγηθεί ως εξής. Υπάρχουν τρεις κύριοι παράγοντες που επιρρεάζουν τον χρόνο απόκρισης: (i) ο αριθμός των (πιθανώς) αποτυχημένων ανακατευθύνσεων, (ii) ο χρονοπρογραμματισμός στης ουρές αναμονής (ο οποίος μοιάζει με

SJF στην περίπτωση του SeAl, αφού οι κόμβοι με καλές βαθμολογίες έχουν συνήθως καλύτερη συνδεσιμότητα και μεγαλύτερο εύρος ζώνης δικτύου από τους κόμβους με χαμηλές βαθμολογίες), και (iii) ο μηχανισμός ανάδρασης, ο οποίος έχει ως αποτέλεσμα (όπως είδαμε και στο σχήμα 5.3) να αυξάνεται ο αριθμός των κόμβων που τελικά εξυπηρετούν αιτήσεις και άρα να μειώνεται το μέσο μήκος των ουρών αναμονής των κόμβων.

Προφανώς, το SeAl εποφελείται από τα (ii) και (iii) αλλά όχι από το (i). Στην περίπτωση που 90% των κόμβων είναι ιδιοτελείς, η επίδραση του πρώτου παράγονται επισκιάζεται από τον δεύτερο και τον τρίτο παράγοντα. Ωστόσο, στην περίπτωση που ιδιοτελείς είναι το 70% των κόμβων, η επίδραση του τρίτου παράγοντα είναι μικρότερη, εξαιτίας του αυξημένου αριθμού (κατά 3 φορές· 30% έναντι 10% των κόμβων) των αλτρουιστών στο σύστημα, εξ΄ ού και η μικρότερη βελτίωση.

Σημείωνουμε ότι το γεγονός της αλλαγής στην συμπεριφορά των ιδιοτελών κόμβων (οπότε και αυτοί αρχίζουν να διαμοιράζονται αρχεία και να εξυπηρετούν αιτήσεις) μπορεί επίσης να αυξήσει το μέσο χρόνο απόκρισης· ενώ στο κλασσικό σύστημα σχεδόν όλες οι αιτήσεις εξυπηρετούνται από αλτρουιστικούς κόμβους με μεγάλο εύρος ζώνης δικτύου, όταν το SeAl και οι μηχανισμοί του μπαίνουν σε λειτουργία, ένα τμήμα των αιτήσεων εξυπηρετείται από κόμβους με μικρό εύρος ζώνης, αυξάνοντας έτσι τον μέσο συνολικό χρόνο απόκρισης.

Κεφάλαιο 6

Σχετικές εργασίες

Τα προβλήματα της «εμπιστοσύνης» (trust), της «φήμης» (reputation) και της αξιοπιστίας (accountability) των συμμετεχόντων σε κατανεμημένα συστήματα, έχουν απασχολήσει την ερευνητική και ακαδημαϊκή κοινότητα για αρκετό καιρό[59]. Ένα αντίστοιχα σημαντικό αλλά κάπως αντικρουόμενο πρόβλημα σε τέτοια συστήματα είναι και αυτό της ανωνυμίας (anonymity) των κόμβων. Συνήθως τα συστήματα που ασχολούνται με δημοσίευση περιεχομένου ανθεκτική σε λογοκρισία (censorship-resistant publishing) και με ζητήματα σχετικά με ανώνυμες λειτουργίες[1, 8, 16, 62, 76], παρέχουν μικρές (η καθόλου) δυνατότητες για επίβλεψη και καταγραφή των διαδικασιών που εκτελούνται στο σύστημα. Από την άλλη, υπάρχοντα συστήματα σε ευρεία χρήση τα οποία κρατούν κάποια μεταδεδομένα για τους συμμετέχοντες κόμβους[25, 30] δεν τα χρησιμοποιούν για να αντιμετωπίσουν τα προβλήματα που προκύπτουν από την ύπαρξη και δράση ιδιοτελών και κακόβουλων χρηστών.

Τα σχήματα που βασίζονται σε «ψηφιακές πληρωμές» φαίνονται να αποτελούν την καλύτερη λύση για το πρόβλημα της επιβολής της συνεισφοράς πόρων από τους κόμβους ενός κατανεμημένου συστήματος. Ωστόσο, σχεδόν όλα τα υπάρχοντα συστήματα ψηφιακής πληρωμής[15, 65], ακόμα και δίκτυα ομοτίμων που βασίζονται σε τέτοιες ιδέες[55], χτίζουν πάνω στην ύπαρξη μίας καθολικά έμπιστης οντότητας η οποία

είτε διαχειρίζεται το «ψηφιακό συνάλλαγμα», είτε ταυτοποιεί μοναδικά τους συμμετέχοντες. Οι Golle κ.ά.[32] εξετάζουν το πρόβλημα της δίκαιης συνεισφοράς στα δίκτυα ομοτίμων από την οπτική γωνία της Θεωρίας Παιγνίων. Ωστόσο, και αυτοί περιορίζουν τη μελέτη τους σε συστήματα που χρησιμοποιούν κεντρικοποιημένες υπηρεσίες και εξυπηρετητές.

Οι Cornelli κ.ά.[24], όπως και το SeAl, χρησιμοποιούν ισχυρές κρυπτογραφικές τεχνικές για να εγγυηθούν την ακεραιότητα και την μυστικότητα των δεδομένων. Το σύστημά τους βασίζεται σε ένα σχήμα αίτησης-ψηφοφορίας-πρόκλησης-επιλογής (poll-vote-challenge-select). Ωστόσο, προσπαθούν να λύσουν το πρόβλημα της επιλογής του πιο αξιόπιστου κόμβου από τον οποίον θα ανακτήσουν κάποιο αρχείο από το σύνολο των κόμβων που συνεισφέρουν το αρχείο αυτό, ενώ το σύστημά μας παρέχει σαφώς ευρύτερη λειτουργικότητα.

Στο GNUnet[28] οι κόμβοι αναθέτουν στις αιτήσεις τους *τιμές προτεραιότητας*, οι οποίες συμβολίζουν το ποσό της εμπιστοσύνης/φήμης που διατίθενται να διακινδυνεύσουν για την αντίστοιχη αίτηση, και χτίζουν λίστες εμπιστοσύνσης με βάση της τιμές αυτές και το αποτέλεσμα των αντίστοιχων συναλλαγών. Το GNUnet, όπως και το SeAl, επιτρέπει την εκμετάλλευση των πλεοάζοντων πόρων, τους οποίους επίσης ονομάζουμε «θετικές συγκυρίες» (positive externalities). Ωστόσο, το SeAl χρησιμοποιεί έναν σαφώς ισχυρότερο μηχανισμό επιβολής δίκαιας συνεισφοράς, με τις Αποδείξεις Συναλλαγών και τις Μαρτυρίες, ενώ διατηρεί τον πλήρως κατανεμημένο χαρακτήρα του.

Το σύστημα FreeHaven[19] δομείται γύρω από μια σχετικά στατική ομάδα εξυπηρετητών, με πολλούς πελάτες να χρησιμοποιούν τους πόρους των πρώτων. Ωστόσο, μόνο οι εξυπηρετητές συμμετέχουν στις σχετικές με εμπιστοσύνη διαδικασίες, οπότε η πλειοψηφία των χρηστών του συστήματος δεν λαμβάνεται υπόψιν. Επιπλέον, οι παραβάσεις των σχέσεων εμπιστοσύνης κοινοποιούνται σε όλους τους εξυπηρετητές (broadcast) καθιστώντας το συνολικό κόστος, όσον αφορά το απαιτούμενο εύρος ζώνης

δικτύου, απαγορευτικό για συστήματα μεγάλης κλίμακας όπου όλοι οι κόμβοι είναι τόσο πελάτες όσο και εξυπηρετητές.

Δύο εργασίες πολύ κοντινές με τη δική μας είναι αυτές των Aberer κ.ά.[5] και Ngan κ.ά.[58]. Το [5] χρησιμοποιεί επίσης ένα υπόστρωμα DHT (το P-Grid[3]) για την αποθήκευση πληροφοριών σχετικών με τις συναλλαγές των κόμβων στο σύστημα. Σε σύγκριση με αυτή την εργασία, το SeAl:

- χρησιμοποιεί τόσο θετική (Μαρτυρίες Προσφοράς) όσο και αρνητική (Μαρτυρίες Κατηγορίας) βαθμολόγηση για τους κόμβους του συστήματος, σε αντίθεση με μόνο αρνητική βαθμολόγηση («παράπονα» – «complaints») στην περίπτωση του [5],

- χρησιμοποιεί ένα σχήμα χρονοδρομόγησης των αιτήσεων με βάση την βαθμολογία – επίπεδο αλτρουισμού – των αντίστοιχων κόμβων, το οποίο λειτουργεί σε επίπεδο ικανοποίησης του χρήστη ως κίνητρο για την δίκαιη συνεισφορά αρχείων και πόρων,

- φαίνεται να έχει πιο αποδοτικό μηχανισμό επαλήθευσης, χρησιμοποιώντας τυχαίους ελέγχους και διαφορετικά επίπεδα αυστηρότητας,

- κάνει γενικότερες υποθέσεις, επιτρέποντας μεταβλητότητα στα πρότυπα συμπεριφοράς των χρηστών του, διαφορετικά επίπεδα αυστηρότητας και μεθόδους αντίδρασης, καθώς και την εκμετάλλευση «θετικών συγκυριών», και

- αντιμετωπίζει και χρονικά μεταβαλλόμενες συμπεριφορές, μέσω του μηχανισμού γήρανσης των Μαρτυριών, ζήτημα που δεν θίχθηκε καθόλου από το [5].

Από την άλλη, το [58] ακολουθεί μία διαφορετική προσέγγιση· οι κόμβοι διατηρούν «αρχεία χρήσης» (usage files) – παρόμοια με τις λίστες χαρών του συστήματός μας – και οι συμμετέχοντες κόμβοι εκτελούν ελέγχους στα αρχεία των γειτονικών τους, αλλα και τυχαία επιλεγμένων, κόμβων ανά τυχαία χρονικά διαστήματα. Ωστόσο το

πρόβλημα που προσπαθεί να λύσει η εργασία αυτή είναι περισσότερο σχετικό με «εμπορία» (trading)· οι κόμβοι ενδιαφέρονται να αποθηκεύσουν τα αρχεία τους σε άλλους κόμβους, αντί για να αποκτήσουν πρόσβαση στα αρχεία που οι άλλοι κόμβοι ήδη αποθηκεύουν και διαμοιράζονται. Το SeAl είναι σαφώς καλύτερα ορισμένο όσον αφορά την προσφερόμενη λειτουργικότητα και τον ακριβή τρόπο λειτουργίας του (π.χ. διαχείριση πληροφορίας, υπολογισμός βαθμολογίας κόμβων, κτλ.) Επιπλέον, το [58] χρησιμοποιεί λογική «ποσόστωσης» (quota)· αν ένας κόμβος είναι κάτω από την αντίστοιχη ποσόστωση (under quota) – δηλαδή ιδιοτελής αλλά όχι απαραίτητα free-rider – αποκλείεται από το δίκτυο. Αυτό δεν αφήνει καθόλου περιθώρια για την εκμετάλλευση των πλεοναζόντων υπολογιστικών/δικτυακών/αποθηκευτικών πόρων των κόμβων του δικτύου.

Όσον αφορά τους κακόβουλους κόμβους, πολλές από τις επιθέσεις που αναλύθηκαν στην ενότητα 4.5 είχαν αναφερθεί στο [69]. Οι λύσεις, όμως, που προτείνονται από τους συγγραφείς του [69] βασίζονται στην χρησιμοποίηση μίας αμφίδρομης μοναδικής αντιστοίχισης ανάμεσα στην IP διεύθυνση ενός κόμβου και το αναγνωριστικό ταυτοποίησης του κόμβου στο δίκτυο ομοτίμων. Ωστόσο, η αντιστοίχηση αυτή – εκτός από του ότι δεν είναι εφικτή – δεν είναι καν επιθυμητή αν το σύστημα σκοπεύει να παρέχει και ανωνυμία στους συμμετέχοντές του.

Εκτός από τις παραπάνω επιμέρους διαφορές του SeAl από τις σχετικές εργασίες, το σύστημά μας διακρίνεται για την φιλοσοφία του και την περιεκτικότητά του. Ειδικότερα:

1. συνεισφέρει ένα ολοκληρωμένο σύστημα, το οποίο παρέχει τη δυνατότητα ταυτοποίησης, βαθμολόγησης, και κατηγοριοποίησης των κόμβων σε ιδιοτελείς και αλτρουιστές, αντιμετωπίζει περιπτώσεις ψευδόμενων και συνεργαζόμενων ιδιοτελών/κακόβουλων κόμβων, και αυξάνει τον αριθμό των διαμοιραζόμενων πόρων/αρχείων.

2. στηρίζουμε τα λεγόμενά μας και τις σχεδιαστικές επιλογές μας με μία ολοκλη-

ρωμένη πειραματική ανάλυση τόσο (i) του επιπλέον κόστους (σε αποθηκευτικό χώρο, εύρος ζώνης δικτύου, και χρόνου απόκρισης), όσο και (ii) της επίδρασής του στην σύγκλιση των προτύπων συμπεριφοράς των συμμετεχόντων κόμβων.

3. Το SeAl είναι ένα υπόστρωμα το οποίο μπορεί να ενσωματωθεί τόσο σε αδόμητα όσο και σε δομημένα δίκτυα ομοτίμων και άρα μπορεί να χρησιμοποιηθεί σε οποιοδήποτε υπάρχον ή μελλοντικό P2P σύστημα, αφού δεν εξαρτάται από την υποκείμενη τοπολογία. Το γεγονός αυτό είναι πολύ σημαντικό για την χρησιμότητα και την καταλληλότητα του συστήματός μας.

4. Το SeAl, με τον τμηματοποιημένο σχεδιασμό του, μπορεί να ρυθμιστεί ώστε να λειτουργεί αποδοτικά σε μία πλειάδα περιβαλλόντων και εφαρμογών. Για παράδειγμα, σε ένα συνεργατικό περιβάλλον που απλά μας ενδιαφέρει η δίκαιη κατανομή των πόρων ανάμεσα στα μέλη μιας κοινότητας, μπορούμε να απενεργοποιήσουμε το μηχανισμό επαλήθευσης (SVL) και να χρησιμοποιήσουμε μόνο το μηχανισμό επίβλεψης/καταγραφής.

5. Το SeAl έχει σημαντικές διαφορές στη φιλοσοφία του από τα υπόλοιπα συστήματα: χρησιμοποιεί διαφορετικά επίπεδα αυστηρότητας και αντίδρασης (από απλή δημοσίευση μίας Μαρτυρίας Κατηγορίας), ως την απόρριψη κόμβων από το δίκτυο (π.χ. χρησιμοποιώντας πολιτική μη εξυπηρέτησης των ιδιοτελών κόμβων).

6. Τέλος, το SeAl επιτρέπει την εκμετάλλευση «θετικών συγκυριών» – πλεοναζόντων πόρων – οι οποίες είναι ιδιαίτερα κοινές σε δίκτυα ομοτίμων μεγάλης κλίμακας.

Κεφάλαιο 7

Συμπεράσματα και μελλοντικές επεκτάσεις

Παρουσιάσαμε το SeAl, ένα καινοτόμο υπόστρωμα για δίκτυα ομοτίμων, το οποίο απαντά σε ένα απο τα βασικότερα προβλήματα διαχείρισης δεδομένων και προσβάσεων στα P2P δίκτυα διαμοιρασμού αρχείων: το πρόβλημα της διαφανούς και αποδοτικής ταυτοποίησης των ιδιοτελών, ψευδόμενων, και αλτρουιστικών κόμβων σε ένα P2P δίκτυο και της παροχής κινήτρων ούτως ώστε να αυξηθεί η ποσότητα των υπολογιστικών πόρων και των δεδομένων που είναι διαθέσιμα στην P2P κοινότητα. Το πρόβλημα αυτό είναι θεμελιώδες, καθώς οι ιδιοτελείς συμπεριφορές αποτελούν όχι απλά πραγματικότητα, αλλά τον κανόνα σε πολλά υπάρχοντα συστήματα ομοτίμων, και είναι κοινώς αποδεκτό ότι αν δεν αντιμετωπιστεί, θα ελαττωθούν σημαντικά η κλιμάκωση, η αποδοτικότητα, και η χρησιμότητα των συστημάτων ομοτίμων διαμοιρασμού αρχείων.

Στα πλαίσια της προσπάθειας αυτής το SeAl προσφέρει:

- ορισμούς και μετρικές για την ιδιοτέλεια ή τον αλτρουισμό ενός κόμβου,
- λύσεις που επιτρέπουν την αποδοτική, αξιόπιστη, και επαληθεύσιμη καταγραφή

και ταυτοποίηση των ιδιοτελών κόμβων, και

- μηχανισμούς παροχής κινήτρων δίκαιης συνεισφοράς.

Ταυτόχρονα, σέβεται την αυτονομία του κάθε κόμβου στο να ορίσει την δική του πολιτική και στάση, ενώ επιτρέπει την εκμετάλλευση των πλεοναζόντων πόρων, οι οποίοι αφθονούν στα δίκτυα ομοτίμων.

Το SeAl μπορεί να ενσωματωθεί τόσο σε αδόμητα όσο και σε δομημένα δίκτυα ομοτίμων και άρα μπορεί να χρησιμοποιηθεί σε οποιοδήποτα υπάρχουσα ή μελλοντική εφαρμογή των P2P συστημάτων. Οι προσομοιώσεις και οι εκτενείς πειραματικές μετρήσεις που παρουσιάσαμε, αποδεικνύουν ότι το SeAl επιτυγχάνει τους στόχους του γρήγορα και αποδοτικά. Παράλληλα, τα επιπλέον κόστη σε εύρος ζώνης δικτύου, αποθηκευτικό χώρο, και χρόνο απόκρισης που εισαγάγονται από το SeAl αποδείχθηκαν άκρως ικανοποιητικά.

Το σύστημα SeAl μπορεί να αποτελέσει τη βάση πάνω στην οποία θα χτιστούν μία πλειάδα εφαρμογών και υπηρεσιών για δίκτυα ομοτίμων. Ως παραδείγματα τέτοιων υπηρεσιών μπορούμε να αναφέρουμε:

- τον εντοπισμό δυνατών (αλτρουιστικών) κόμβων σε ένα δίκτυο ομοτίμων και την χρησιμοποίησή τους σε δικτυακές αρχιτεκτονικές[72] με τρόπο τέτοιο ο οποίος επιτυγχάνει σαφώς καλύτερες επιδόσεις στην δρομολόγηση αιτήσεων και στην παρεχόμενη σταθερότητα σε σχέση με τις υπάρχουσες αρχιτεκτονικές,

- την εξισορρόπηση του φόρτου σε ένα P2P δίκτυο[74], μέσω ανακατευθύνσεων των αιτήσεων, όπως κάναμε και κατά την αποπληρωμή των χαρών στο μηχανισμό επίβλεψης/καταγραφής, και

- την ανάπτυξη αποδοτικών αλγορίθμων κατανεμημένης επεξεργασίας και βελτιστοποίησης πολύπλοκων ερωτημάτων[73], όπου το SeAl θα διατηρεί στατιστικά για τα ερωτήματα που εκτελούνται και θα βοηθά στον εντοπισμό υπερφορτωμένων ή υποχρησιμοποιούμενων κόμβων του δικτύου, ή ισχυρών κόμβων που θα

μπορούσαν να χρησιμοποιηθούν για να επισπεύσουν την εκτέλεση των ερωτημάτων.

Βιβλιογραφία

[1] A. Rowstron et al. Scribe: The design of a large-scale event notification infrastructure. In *Proc. COST264 Workshop '01*.

[2] M. Abe. Universally verifiable mix-net with verification work independent of the number of servers. In *Advances in Cryptology - EUROCRYPT '98*.

[3] K. Aberer. P-Grid: A self-organizing access structure for P2P information systems. In *Proc. CoopIS '01*.

[4] K. Aberer, A. Datta, and M. Hauswirth. A decentralized public key infrastructure for customer-to-customer e-commerce. *International Journal of Business Process Integration and Management*, to be published.

[5] K. Aberer and Z. Despotovic. Managing trust in a peer-to-peer information system. In *Proc. CIKM '01*.

[6] E. Adar and B. Huberman. Free riding on Gnutella. Technical report, Xerox PARC, 2000.

[7] Advogato. http://www.advogato.org/.

[8] R. Anderson. The Eternity service. In *Proc. PRAGOCRYPT '96*.

[9] AudioGalaxy. http://www.audiogalaxy.com/.

[10] O. Berthold, H. Federrath, and M. Köhntopp. Anonymity and unobservability on the internet. In *Proc. Workshop on Freedom and Privacy by Design / CFP '00*.

[11] O. Berthold, H. Federrath, and S. Köpsell. Web MIXes: a system for anonymous and unobservable internet access. In *Proc. Workshop on Design Issues in Anonymity and Unobservability '01*.

[12] M. Castro, M. Costa, and A. Rowstron. Should we build gnutella on a structured overlay? In *Proc. HotNets II '03*.

[13] D. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–88, 1981.

[14] D. Chaum. The dining cryptographers problem: Unconditional sender and recipient untraceability. *Journal of Cryptology*, 1:65–75, 1988.

[15] D. Chaum, A. Fiat, and M. Naor. Untraceable electronic cash. In *Proc. CRYPTO '88*.

[16] I. Clarke, O. Sandberg, B. Wiley, and T. Hong. Freenet: A distributed anonymous information storage and retrieval system. In *Proc. ICSI Workshop on Design Issues in Anonymity and Unobservability '00*.

[17] L. Cottrell. Frequently asked questions about Mixmaster remailers. http://www.obscura.com/~loki/remailer/mixmaster-faq.html.

[18] A. Crespo and H. Garcia-Molina. Routing indices for peer-to-peer systems. In *Proc. ICDCS '02*.

[19] R. Dingledine, M. Freedman, and D. Molnar. The Free Haven Project: Distributed anonymous storage service. In *Proc. ICSI Workshop on Design Issues in Anonymity and Unobservability '00*.

[20] Distributed.Net. http://www.distibuted.net/.

[21] J. Douceur. The Sybil attack. In *Proc. IPTPS '02*.

[22] P. Druschel and A. Rowstron. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In *Proc. IFIP/ACM Middleware '01*.

[23] D. Eastlake and P. Jones. *RFC 3174 - US Secure Hash Algorithm 1 (SHA1)*. Network Working Group – IETF, September 2001.

[24] F. Cornelli et al. Implementing a reputation-aware Gnutella servent. In *Proc. Networking Workshops '02*.

[25] FastTrack. http://www.fasttrack.nu/.

[26] M. Freedman and R. Morris. Tarzan: A Peer-to-Peer anonymizing network layer. In *Proc. ACM CCS '02*.

[27] M. Freedman, E. Sit, J. Cates, and R. Morris. Introducing Tarzan, a Peer-to-Peer anonymizing network layer. In *Proc. IPTPS '02*.

[28] C. Ghrothoff. An excess-based economic model for resource allocation in peer-to-peer networks. *Wirtschafts Informatik*, March 2003.

[29] N. Glance and B. Huberman. Dynamics of Social Dilemmas. *Scientific American*, Mar 1994.

[30] Gnutella. http://gnutella.wego.com/.

[31] D. Goldschlag, M. Reed, and P. Syverson. Onion routing for anonymous and private internet connections. *Communications of the ACM*, 42(2):39–41, 1999.

[32] P. Golle, K. Leyton-Brown, and I. Mironov. Incentives for sharing in peer-to-peer networks. In *Proc. ACM EC '01*.

[33] L. Gong. JXTA: A network programming environment. *IEEE Internet Computing*, 5(3):88–95, May/June 2001.

[34] Grokster. http://www.grokster.com/.

[35] R. Gummadi, S. Gribble, S. Ratnasamy, S. Shenker, and I. Stoica. The impact of DHT routing geometry on resilience and proximity. In *Proc. SIGCOMM '03*.

[36] G. Hardin. The Tragedy of the Commons. *Science*, 162(1968):1243–1248.

[37] S. Hazel and B. Wiley. Achord: a variant of the Chord lookup service for use in censorship resistant Peer-to-Peer publishing systems. In *Proc. IPTPS '02*.

[38] B. Huberman and R. Lukose. Social Dilemmas and Internet Congestion. *Science*, 277(1997):535.

[39] I. Stoica et al. Chord: A scalable Peer-To-Peer lookup service for internet applications. In *Proc. ACM SIGCOMM '01*.

[40] IETF. SPKI Working Group. http://www.ietf.org/ html.charters/spki-charter.html.

[41] M. Jakobsson. Flash mixing. In *Proc. PODC '99*.

[42] M. Jakobsson. A practical mix. In *Proc. EUROCRYPT '98*.

[43] Kazaa. http://www.kazaa.com/.

[44] D. Kesdogan, J.Egner, and R. Buschkes. Stop and go mixes: Providing probabilistic anonymity in an open system. In *Proc. Information Hiding Workshop '98*.

[45] LimeWire. http://www.limewire.org/.

[46] M. Luby. LT codes. In *Proc. FOCS '02*.

[47] Q. Lv, S. Ratnasamy, and S. Shenker. Can heterogeneity make Gnutella scalable? In *Proc. IPTPS '02*.

[48] M. Luby et al. Practical loss-resilient codes. In *Proc. STOC '97*.

[49] D. Malkhi, M. Naor, and D. Ratajczak. Viceroy: A scalable and dynamic emulation of the butterfly. In *Proc. ACM PODC '02*.

[50] P. Maymouknov and D. Mazières. Kademlia: A peer-to-peer information system based on the XOR metric. In *Proc. IPTPS '02*.

[51] MD5CRK. http://www.md5crk.com/.

[52] M. Mealling, P. Leach, and R. Salz. *A UUID URN Namespace*. Network Working Group – IETF, October 2002.

[53] A. Menejes, P. van Oorschot, and S. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1996.

[54] S. Milgram. The small world problem. *Psychology Today*, 2:60–67, 1967.

[55] Mojonation. http://www.mojonation.com/.

[56] Morpheus. http://www.morpheus.com/.

[57] Napster. http://www.napster.com/.

[58] T. Ngan, D. Wallach, and P. Druschel. Enforcing fair sharing of peer-to-peer resources. In *Proc. IPTPS '03*.

[59] A. Oram, editor. *Peer-to-Peer: Harnessing the Power of Disruptive Technologies*, chapter Trust. O'Reilly, 2001.

[60] C. Plaxton, R. Rajaraman, and A. Richa. Accessing nearby copies of replicated objects in a distributed environment. In *Proc. ACM SPAA '97*.

[61] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content-addressable network. In *Proc. ACM SIGCOMM '01*.

[62] M. Reiter and A. Rubin. Anonymous web transactions with Crowds. *Communications of the ACM*, 42(2):32–38, 1999.

[63] The Rewebber. http://www.rewebber.de/.

[64] R. Rivest. *RFC 1321 - The MD5 Message-Digest Algorithm*. Network Working Group – IETF, April 1992.

[65] R. Rivest and A. Shamir. PayWord and MicroMint: Two simple micropayment schemes. In *Proc. Security Protocols Workshop '96*.

[66] S. Saroiu, K. Gummadi, and S. Gribble. A measurement study of peer-to-peer file sharing systems. In *Proc. MMCN '02*.

[67] Seti@Home. http://setiathome.ssl.berkeley.edu/.

[68] A. Shamir. How to share a secret. *Communications of the ACM*, 22:612–613, 1979.

[69] E. Sit and R. Morris. Security considerations for peer-to-peer distributed hash tables. In *Proc. IPTPS '02*.

[70] K. Sripanidkulchai. The popularity of gnutella queries and its implications on scalability. White paper, Feb. 2001.

[71] P. Syverson, D. Goldschlag, and M. Reed. Anonymous connections and Onion routing. In *Proc. IEEE Symposium on Security and Privace '97*.

[72] P. Triantafillou. Peer-to-peer network architectures: The next step. In *SIGCOMM FDNA'03 - Invited talk*.

[73] P. Triantafillou and T. Pitoura. Towards a unifying framework for complex query processing over structured peer-to-peer data networks. In *Proc. VLDB DBISP2P '03*.

[74] P. Triantafillou, C. Xiruhaki, M. Koubarakis, and N. Ntarmos. Towards high performance peer-to-peer content and resource sharing systems. In *Proc. ACM SIGMOD/VLDB CIDR '03*. http://www.ceid.upatras.gr/faculty/peter/papers/cidr03.pdf.

[75] P. van Oorschot and M. Wiener. Parallel collision search with application to hash functions and discrete logarithms. In *Proc. ACM CCS '94*.

[76] M. Waldman, A. Rubin, and L. Cranor. Publius: A robust, tamper-evident, censorship-resistant, web publishing system. In *Proc. USENIX Security Symposium '00*.

[77] H. Weatherspoon and J. Kubiatowicz. Erasure coding vs. replication: A quantitative comparison. In *Proc. IPTPS '02*.

[78] B. Yang and H. Garcia-Molina. Designing a super-peer network. In *Proc. ICDE '03*.

[79] B. Yang and H. Garcia-Molina. Efficient search in peer-to-peer networks. In *Proc. ICDCS '02*.

[80] B. Yang and H. Garcia-Molina. Comparing hybrid peer-to-peer systems. In *The VLDB Journal*, Sept. 2001.

[81] B. Zhao, J. Kubiatowicz, and A. Joseph. Tapestry: An infrastructure for fault-tolerant wide-area location and routing. Technical Report UCB/CSD-01-1141, University of California at Berkeley, CSD, 2001.