



ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΜΕΤΑΠΤΥΧΙΑΚΟ ΔΙΠΛΩΜΑ ΕΙΔΙΚΕΥΣΗΣ (MSc)  
στα ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ

*ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ*

«Ασφάλεια συστημάτων επικοινωνίας ομάδας»

Τσαρούχης Στράτος

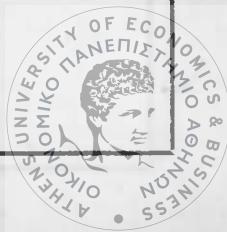
M3040001

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ  
ΚΑΤΑΛΟΓΟΣ



ΑΘΗΝΑ, ΦΕΒΡΟΥΑΡΙΟΣ 2006

0 000000 570435 0



**ΜΕΤΑΠΤΥΧΙΑΚΟ ΔΙΠΛΩΜΑ ΕΙΔΙΚΕΥΣΗΣ (MSc)  
στα ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ**

**ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

ΟΙΚΟΝΟΜΙΚΟ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΑΘΗΝΩΝ  
ΒΙΒΛΙΟΘΗΚΗ  
εισ. 79754  
Αρ.  
παξ.

«Ασφάλεια συστημάτων επικοινωνίας ομάδας»

Τσαρούχης Στράτος

M3040001

Επιβλέπων Καθηγητής: Θεόδωρος Αποστολόπουλος  
Εξωτερικός Κριτής: Δημήτριος Γκρίτζαλης

ΟΙΚΟΝΟΜΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ  
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΑΘΗΝΑ, ΦΕΒΡΟΥΑΡΙΟΣ 2006



# Περιεχόμενα

Εισαγωγή .....	4
Η δομή της εργασίας .....	5
Ενότητα 1 <sup>η</sup> : Συστήματα επικοινωνίας ομάδας .....	5
Ενότητα 2 <sup>η</sup> : Βασικά ζητήματα ασφάλειας των Group Communication συστημάτων ..	7
Αυθεντικοποίηση .....	8
Πρωτόκολλα που στηρίζονται στην χρήση μίας έμπιστης τρίτης οντότητας .....	11
Πρωτόκολλα που απαιτούν την ύπαρξη μίας έμπιστης τρίτης οντότητας μόνο κατά την φάση εγγραφής νέου χρήστη .....	21
Πρωτόκολλα που δεν χρησιμοποιούν έμπιστη τρίτη οντότητα .....	26
Συμπεράσματα .....	29
Έλεγχος πρόσβασης σε μία ομάδα .....	30
Αρχιτεκτονικές ελέγχου πρόσβασης .....	30
ID-based αρχιτεκτονική .....	31
Αρχιτεκτονική που κάνει χρήση πιστοποιητικών δημοσίου κλειδιού .....	38
“Bouncer” – Ένα εργαλείο για τον έλεγχο πρόσβασης σε ένα group communication σύστημα .....	49
Συμπεράσματα .....	62
Διαχείριση μυστικού κλειδιού συνόδου ομάδας .....	63
Ιδιότητες πρωτοκόλλων .....	65
Κεντρικοποιημένα Πρωτόκολλα .....	66
Μη κεντρικοποιημένα πρωτόκολλα .....	75
Κατανεμημένα πρωτόκολλα .....	79
Cliques toolkit – Μία σουίτα πρωτοκόλλων διαχείρισης του κλειδιού ενός group .....	102
Συμπεράσματα .....	103
Ενότητα 3 <sup>η</sup> : Συστήματα ασφαλούς επικοινωνίας ομάδας .....	105
Antigone .....	106
Αρχιτεκτονική του Antigone .....	108
Υλοποίηση πολιτικών στο Antigone .....	111
Secure Spread .....	112
Αρχιτεκτονική του Secure Spread .....	112
Αυθεντικοποίηση και έλεγχος πρόσβασης στο Secure Spread .....	112
Διαχείριση κλειδιού συνόδου .....	114
Σύγκριση IA - LA .....	118
Ensemble .....	118
Ιδιότητες του Ensemble .....	118
Συνένωση υπο – groups .....	120
Διανομή του κλειδιού συνόδου .....	121
Secure Group Layer .....	122
Συστήματα επικοινωνίας ομάδας που αντιμετωπίζουν τις εκ των έσω απειλές – Το σύστημα Secure Ring .....	125
Θεώρηση του συστήματος .....	125
Πρωτόκολλα του Secure Ring .....	127
Κριτήρια επιλογής ενός συστήματος – Σύγκριση συστημάτων .....	134
Ενότητα 4 <sup>η</sup> : Ενοποίηση Secure Spread – Bouncer .....	137
Αρχιτεκτονική ενοποίησης .....	137



Εγκατάσταση και λειτουργία.....	139
Εγκατάσταση .....	139
Παραμετροποίηση συστήματος.....	140
Σύνδεση μέλουνς.....	141
Ανασκόπηση .....	142
Περιληψη .....	143
Executive Summary.....	146
Βιβλιογραφία .....	149

## Εισαγωγή



Ένας από τους ποίο διαδεδομένους τρόπους επικοινωνίας στις μέρες μας είναι το διαδίκτυο. Το διαδίκτυο σήμερα χρησιμοποιείται σαν κανάλι επικοινωνίας για πληθώρα δραστηριοτήτων, όπως είναι εμπορικές συναλλαγές, ανταλλαγή αρχείων, ηλεκτρονικό ταχυδρομείο, ηλεκτρονική διακυβέρνηση και άλλων εφαρμογών για επιστημονικούς, ψυχαγωγικούς και εμπορικούς σκοπούς.

Είναι εμφανές ότι πολλές από τις παραπάνω εφαρμογές απαιτούν ασφάλεια στην επικοινωνία προκειμένου να είναι αποτελεσματικές. Δυστυχώς όμως το διαδίκτυο είναι ένα ανασφαλές μέσο επικοινωνίας. Ο σχεδιασμός του έγινε με γνώμονα την αποτελεσματικότητα, την ταχύτητα και την απλότητα του δικτύου. Αρχικά σκοπός του διαδικτύου ήταν η δημιουργία ενός αξιόπιστου διαύλου επικοινωνίας που θα χρησιμοποιούνταν από μία μικρή ομάδα ατόμων για επιστημονικούς σκοπούς, η ασφάλεια δεν ήταν ζητούμενη και δεν αποτέλεσε στόχο των κατασκευαστών. Η φιλοσοφία του διαδικτύου εξάλλου είναι να διατηρηθεί το δίκτυο όσο το δυνατών ποιο απλό και να μεταφερθεί η πολυπλοκότητα στα άκρα.

Τα τελευταία χρόνια έχουν αναπτυχθεί αρκετές τεχνικές που επιτυγχάνουν ασφάλεια στην επικοινωνία μεταξύ δύο οντοτήτων στο διαδίκτυο, οι τεχνικές αυτές όμως δεν μπορούν να καλύψουν τις απαιτήσεις ασφάλειας κατανεμημένων συστημάτων που διαρκώς αναπτύσσονται και εξαπλώνονται μεταξύ των χρηστών του διαδικτύου. Παράδειγμα τέτοιων συστημάτων είναι τα peer to peer συστήματα και συστήματα επικοινωνίας ομάδας (group communication systems)

Σκοπός της παρούσας εργασίας είναι να θίξει τα βασικά ζητήματα ασφάλειας των group communication συστημάτων, να παρουσιάσει τις τεχνικές που έχουν αναπτυχθεί για την αντιμετώπιση των ζητημάτων ασφάλειας των παραπάνω συστημάτων, να αναδείξει τα πλεονεκτήματα και τα μειονεκτήματα της κάθε τεχνικής και να εντοπίσει τις δυνατότητες υλοποίησης τους. Ήα επικεντρωθούμε κυρίως σε κατανεμημένες αρχιτεκτονικές και μηχανισμούς που δεν θα παραβιάζουν την κατανεμημένη φύση των group communication συστημάτων.

Στην εργασία αυτή θα χρησιμοποιούνται συχνά όροι δανεισμένοι από την αγγλική γλώσσα αφού προηγουμένως έχουν ερμηνευθεί. Η χρήση αγγλικής ορολογίας είναι κατά την γνώμη μας ποίο συνοπτική και περιεκτική και δίνει την δυνατότητα σε άτομα με εμπειρία στο χώρο της ασφάλειας και των group communication

συστημάτων να διαβάσουν με μεγαλύτερη ευκολία την παρούσα εργασία. Επιπλέον η ακριβή μετάφραση όρων είναι ιδιαίτερα δύσκολη και επικίνδυνη, απαιτεί πολύ προσπάθεια και ξεφεύγει από τους στόχους της παρούσας εργασίας.

## Η δομή της εργασίας

Η εργασία χωρίζεται σε τέσσερις βασικές ενότητες στην πρώτη γίνεται μία συνοπτική ανάλυση των αρχών των group communication συστημάτων, στην δεύτερη παρατίθενται και αναλύονται τα βασικά ζητήματα ασφάλειας των group communication συστημάτων και παρουσιάζονται τεχνικές που έχουν προταθεί για την επίλυση τους, στην τρίτη ενότητα παρουσιάζονται οι μηχανισμοί ασφάλειας κάποιων γνωστών συστημάτων για group communication και στην τελευταία ενότητα σχολιάζεται το σύστημα “Secure Spread” ενοποιημένο με τον “Bouncer”. Το Secure Spread εξασφαλίζει αποτελεσματικούς μηχανισμούς διαχείρισης του κλειδιού συνόδου ενός group, ενώ ο Bouncer παρέχει στο Secure Spread αποτελεσματικούς μηχανισμούς ελέγχου πρόσβασης κάποιου νέου μέλους στο group.

## Ενότητα 1<sup>η</sup>: Συστήματα επικοινωνίας ομάδας

Τα group communication συστήματα μπορούν να διαχωριστούν σε δύο βασικές κατηγορίες. Οι κατηγορίες αυτές είναι τα:

- Σύγχρονα συστήματα. Το ιδιαίτερο χαρακτηριστικό των σύγχρονων συστημάτων είναι η ύπαρξη χρονικών περιορισμών για την επικοινωνία και την επεξεργασία. Αν θεωρήσουμε ένα αξιόπιστο επικοινωνιακό μέσο και επεξεργαστές με την ιδιότητα να σταματούν την επικοινωνία σε περίπτωση αποτυχίας, ο αποστολέας ενός μηνύματος μπορεί να γνωρίζει την αποτυχία ή επιτυχία λήψης του μηνύματος που έστειλε. Όταν ο αποστολέας δεν λάβει ένα μήνυμα επιβεβαίωσης (acknowledgement) από τον παραλήπτη στα επιτρεπόμενα χρονικά όρια τότε είναι σίγουρος ότι ο παραλήπτης έχει αποτύχει.
- Ασύγχρονα συστήματα. Στα ασύγχρονα συστήματα η καθυστέρηση στην επικοινωνία και την επεξεργασία είναι αυθαίρετη ακόμα και στην περίπτωση που δεν προέρχεται από κάποια αποτυχία στο σύστημα. Στα

συστήματα αυτά δεν υπάρχει τρόπος διαχωρισμού ενός πολύ αργού επεξεργαστή και ενός επεξεργαστή που έχει αποτύχει.

Για να είναι δυνατή η κατανόηση όσων θα ειπωθούν κατά την εξέλιξη της εργασίας θα δώσουμε τους ορισμούς ορισμένων βασικών εννοιών που αφορούν την επικοινωνία σε ένα group communication σύστημα.

- Best effort delivery: Τα συστήματα με αυτή την ιδιότητα δεν δίνουν εγγυήσεις για την μετάδοση των μηνυμάτων και επιπλέον δεν δίνουν εγγυήσεις για την σειρά μετάδοσης των μηνυμάτων.
- Reliable message delivery: Ένα group communication σύστημα με αυτή την ιδιότητα εγγυάται:
  - Αν μία σωστή διαδικασία (process) κάνει broadcast αποστολή ενός μηνύματος  $m$  τότε όλες οι σωστές διαδικασίες θα παραδώσουν τελικά το μήνυμα  $m$ .
  - Αν μία σωστή διαδικασία παραδώσει ένα μήνυμα  $m$  τότε όλες οι σωστές διαδικασίες θα παραδώσουν το μήνυμα  $m$ .
  - Κάθε σωστή διαδικασία παραδίδει το μήνυμα  $m$  το πολύ μία φορά αν και μόνο αν κάποια άλλη διαδικασία είχε κάνει broadcast αποστολή του μηνύματος.
- FIFO message delivery: Η ιδιότητα αυτή εγγυάται ότι και η reliable message delivery αλλά επιπλέον εξασφαλίζει ότι μηνύματα που προέρχονται από την ίδια πηγή θα διανέμονται με την σειρά που στάλθηκαν. Για τα μηνύματα που προέρχονται από διαφορετικές πηγές δεν δίνεται αυτή η εγγύηση (Αν υποθέσουμε για παράδειγμα υποθέσουμε ότι η διαδικασία A απαντά σε μήνυμα της διαδικασίας B τότε τα μηνύματα αυτό μπορεί να διανέμονται με διαφορετική σειρά στις διαδικασίες C και D που ακούνε την επικοινωνία).
- Causal message delivery: Η ιδιότητα αυτή γενικεύει την FIFO message delivery κατά ένα τρόπο που τα μηνύματα διανέμονται κατά σειρά αιτιότητας. Αν για παράδειγμα το μήνυμα  $m$  είναι απάντηση του μηνύματος  $m'$  τότε όλες οι σωστές διαδικασίες πρέπει να διανείμουν το  $m$  πριν το  $m'$ .
- Totally ordered message delivery: Η ιδιότητα αυτή εξασφαλίζει ότι όλα τα μηνύματα θα διανεμηθούν με την ίδια σειρά προς όλες τις κατευθύνσεις.

Στο σημείο αυτό πρέπει να ορίσουμε δυο αρχές των group communication συστημάτων που έχουν να κάνουν με το συγχρονισμό του group κατά την αλλαγή του view. Σαν view ενός group ορίζουμε το σύνολο των μελών του group μία δεδομένη χρονική στιγμή. Οι αρχές αυτές είναι οι ακόλουθες:

- Extended Virtual Synchrony. Η αρχή αυτή εγγυάται ότι ένα μήνυμα θα διανεμηθεί στα μέλη που κατά την ώρα παράδοσης ανήκουν στο ίδιο view. Το view αυτό όμως μπορεί να διαφέρει από το view του group κατά την ώρα αποστολής.
- Virtual Synchrony ή View Synchrony. Η αρχή αυτή εγγυάται ότι ένα μήνυμα θα διανεμηθεί στα μέλη του group view του αποστολέα κατά την ώρα αποστολής (sending view delivery).

Όταν προσπαθούμε να διασφαλίσουμε ένα group communication σύστημα πρέπει να είμαστε ιδιαίτερα προσεκτικοί με τις παραπάνω δύο αρχές. Τα περισσότερα κατανεμημένα συνεργατικά πρωτόκολλα διαχείρισης του μυστικού κλειδιού συνόδου απαιτούν το υποκείμενο επικοινωνιακό σύστημα να υιοθετεί την αρχή του Virtual Synchrony.

Στην παρούσα εργασία όποτε χρησιμοποιείται ο όρος σύστημα επικοινωνίας ομάδας (group communication system) θα εννοούμε ένα σύγχρονο σύστημα επικοινωνίας μεταξύ οντοτήτων διεσπαρμένων στο διαδίκτυο. Τα group communication συστήματα που θα μας απασχολήσουν θα πρέπει να παρέχουν δυνατότητες επικοινωνίας σε wide area networks (WANs) και δεν θα πρέπει να εξαρτώνται από κάποια τεχνολογία μετάδοσης όπως είναι το IP multicast.

## Ενότητα 2<sup>η</sup>: Βασικά ζητήματα ασφάλειας των Group Communication συστημάτων

Η κρυπτογραφία είναι η πλέον διαδεδομένη τεχνική για την αντιμετώπιση των προβλημάτων ασφάλειας στην επικοινωνία μέσω του διαδικτύου αλλά και γενικότερα. Η μορφή της κρυπτογραφίας που χρησιμοποιείται στο διαδίκτυο στηρίζεται στην ύπαρξη ενός δημόσια γνωστού αλγορίθμου κρυπτογράφησης και ενός μυστικού κλειδιού με το οποίο κρυπτογραφείται και αποκρυπτογραφείται το

περιεχόμενο που θέλουμε να αποστείλουμε (συμμετρική κρυπτογραφία) ή την ύπαρξη ενός ζεύγους δημοσίου - ιδιωτικού κλειδιού (ασύμμετρη κρυπτογραφία ή κρυπτογραφία δημοσίου κλειδιού). Στην ασύμμετρη κρυπτογραφία ο αποστολέας κρυπτογραφεί με το δημόσιο κλειδί του παραλήπτη και ο παραλήπτης αποκρυπτογραφεί με το δικό του μυστικό ιδιωτικό κλειδί. Τόσο η συμμετρική όσο και η ασύμμετρη κρυπτογραφία μπορούν να χρησιμοποιηθούν για την δημιουργία ψηφιακών υπογραφών και να εξασφαλίσουν υποστήριξη σε μηχανισμούς αυθεντικοποίησης.

Τα βασικότερα ζητήματα για την επίτευξη ασφαλούς επικοινωνίας μιας ομάδας οντοτήτων στο διαδίκτυο είναι:

- Η αυθεντικοποίηση της κάθε οντότητας
- Ο έλεγχος πρόσβασης κάθε αυθεντικοποιημένης οντότητας στην ομάδα.
- Η ανταλλαγή ενός κοινού μυστικού κλειδιού συνόδου μεταξύ των μελών που έχουν εξασφαλίσει το δικαίωμα πρόσβασης στην ομάδα.

Λέγοντας αυθεντικοποίηση εννοούμε την διαδικασία εκείνη κατά την οποία μία οντότητα αποδεικνύει την ταυτότητά της.

Κάθε ομάδα πρέπει να έχει ένα μηχανισμό με τον οποίο θα ελέγχει ποιες οντότητες έχουν δικαίωμα εισόδου στην ομάδα. Ό μηχανισμός αυτός θα υλοποιεί με άλλα λόγια τον έλεγχο πρόσβασης στην ομάδα.

Η ανταλλαγή ενός κοινού κλειδιού συνόδου γίνεται μεταξύ δύο ή περισσοτέρων αυθεντικοποιημένων οντοτήτων. Το κλειδί είναι προσωρινό, για την διάρκεια μιας συνόδου και σκοπός του είναι να χρησιμοποιηθεί για την κρυπτογράφηση και την αποκρυπτογράφηση των μηνυμάτων που ανταλλάσσονται κατά την διάρκεια της συνόδου.

## Αυθεντικοποίηση

Στην αυθεντικοποίηση δύο είναι τα βασικά ζητήματα που μας απασχολούν

- Η μορφή και το είδος των διαπιστευτηρίων που είναι απαραίτητα προκειμένου μία οντότητα να μπορεί να αποδείξει την ταυτότητά της και ο τρόπος που θα τα αποκτήσει.
- Η διαδικασία ανταλλαγής των διαπιστευτηρίων με ένα τρόπο ώστε οποιοσδήποτε επίβουλος που παρακολουθεί και μπορεί να τροποποιεί τα

μηνύματα που ανταλλάσσονται οι δύο οντότητες να μην μπορεί να παραβιάσει την διαδικασία αυθεντικοποίησης.

Η μορφή το είδος και η διαχείριση των απαιτούμενων διαπιστευτηρίων είναι ένα από τα μεγαλύτερα προβλήματα που έχουν να αντιμετωπίσουν οι ειδικοί που ασχολούνται με την ασφάλεια της επικοινωνίας μέσο του διαδικτύου.

Όταν δύο οντότητες θέλουν να επικοινωνήσουν πρέπει να αποδείξουν την ταυτότητά τους. Ένας τρόπος που μπορεί κάπι τέτοιο να επιτευχθεί είναι με την επίδειξη ενός μυστικού κλειδιού που γνωρίζουν από κοινού και οι δύο. Τίθεται όμως το ερώτημα πως θα αποκτήσουν το κλειδί. Ένας τρόπος είναι να ορίσουν ένα κοινό φυσικό σημείο συνάντησης, να επιδείξουν τις αστυνομικές τους ταυτότητες και να συμφωνήσουν το κοινό μυστικό κλειδί. Ο τρόπος αυτός θα ήταν ίσως αποτελεσματικός στην περίπτωση που ζούσαν στην ίδια πόλη, θα ήταν όμως αδύνατο αν η μία οντότητα βρίσκεται στην Αθήνα και η άλλη στο Λονδίνο. Ακόμα αν απαιτούνταν ένα κλειδί για κάθε οντότητα με την οποία κάποιος επικοινωνεί η διαχείριση ενός τόσο μεγάλου αριθμού κλειδιών θα ήταν προβληματική. Για να λυθούν τα προβλήματα που προαναφέραμε προτάθηκε η ύπαρξη μιας έμπιστης τρίτης οντότητας (TPP) η οποία θα έχει το ρόλο του κέντρου διαμοιρασμού κλειδιών (KDC). Για τον λεπτομερή τρόπο με τον οποίο κάπι τέτοιο μπορεί να επιτευχθεί έχουν προταθεί διάφορες αρχιτεκτονικές, μερικές από τις οποίες θα παρουσιάσουμε στην συνέχεια. Η γενική ιδέα είναι ότι κάθε οντότητα θα διαμοιράζεται με το KDC ένα μυστικό κλειδί. Όταν η οντότητα A επιθυμεί να επικοινωνήσει με την B θα στέλνει μία αίτηση στο KDC που θα δηλώνει την επιθυμία της για επικοινωνία με τον B. Το KDC γνωρίζει ότι ο A είναι αυτός που δηλώνει αφού τον έχει αυθεντικοποίησει μέσο του κοινού τους κλειδιού, στην συνέχεια το KDC στέλνει στον B την αίτηση του A για επικοινωνία. Αν ο B αποδεχτεί την αίτηση το KDC στέλνει κρυπτογραφημένα στον A και τον B ένα μυστικό κλειδί συνόδου και η επικοινωνία μπορεί να ξεκινήσει. Το βασικότερο πρόβλημα της μεθόδου αυτής είναι ότι απαιτείται η ύπαρξη μιας έμπιστης τρίτης οντότητας που την καθιστά μία κεντρικοποιημένη μέθοδο με όλες τις συνέπειες που αυτό συνεπάγεται δηλ. ένα σημείο συμφόρησης που σε περίπτωση αποτυχίας (πρόβλημα στον εξυπηρετητή ή στην γραμμή του δικτύου) έχει σαν αποτέλεσμα την κατάρρευση ολόκληρου του συστήματος.

Η ασύμμετρη κρυπτογραφία μπορεί να χρησιμοποιηθεί για την αυθεντικοποίηση μίας οντότητας. Στην ασύμμετρη κρυπτογραφία σε κάθε οντότητα αντιστοιχίζεται ένα

μοναδικό ζεύγος δημοσίου - ιδιωτικού κλειδιού. Το δημόσιο κλειδί είναι συνεπώς ένα μοναδικό αναγνωριστικό κάθε οντότητας και μπορεί να χρησιμοποιηθεί σαν διαπιστευτήριο. Το σημαντικότερο πρόβλημα που καλούνται να αντιμετωπίσουν οι ειδικοί της ασφάλειας είναι το πώς θα εξασφαλίσουν την αντιστοίχιση μεταξύ μίας οντότητας και ενός δημοσίου κλειδιού με μοναδικό τρόπο παγκοσμίως έτσι ώστε να γνωρίζουν οι πάντες ότι ένα δημόσιο κλειδί χαρακτηρίζει μία οντότητα και μόνο αυτή. Για την επίλυση του παραπάνω προβλήματος υιοθετείται η χρήση μιας έμπιστης τρίτης οντότητας (Trusted Third Party – TTP) και η χρήση πιστοποιητικών. Το πιστοποιητικό είναι ένα έγγραφο που περιέχει το δημόσιο κλειδί, τα στοιχεία της οντότητας, την ημερομηνία λήξης και άλλες πληροφορίες. Το πιστοποιητικό είναι ψηφιακά υπογεγραμμένο από την έμπιστη τρίτη οντότητα που εδώ επιτελεί το ρόλο της αρχής πιστοποίησης (Certification Authority – CA). Βασικό μειονέκτημα και αυτής της μεθόδου είναι η κεντρικοποιημένη φύσης της.

Μία άλλη ιδέα για αυθεντικοποίηση στηρίζεται στην ύπαρξη μίας έμπιστης τρίτης οντότητας που σκοπός της είναι να αποδώσει σε μία οντότητα τα απαιτούμενα διαπιστευτήρια με τα οποία οι άλλες οντότητες θα μπορούν να την αυθεντικοποίησουν. Άλλα μετά την απόδοση των διαπιστευτηρίων δεν συμμετέχει στην διαδικασία αυθεντικοποίησης και μπορεί να πάψει να υφίσταται. Η ανεργοποιηθεί ξανά μόνο κατά την εγγραφή νέων οντοτήτων.

Η α παρουσιάσουμε τώρα ορισμένα πρωτόκολλα και τεχνικές που υλοποιούν τις βασικές ιδέες αυθεντικοποίησης που αναφέρθησαν παραπάνω. Τα πρωτόκολλα θα ομαδοποιηθούν σε τρεις κατηγορίες:

- Πρωτόκολλα που στηρίζονται στην χρήση μίας έμπιστης τρίτης οντότητας
- Πρωτόκολλα που απαιτούν την ύπαρξη μίας έμπιστης τρίτης οντότητας μόνο κατά την φάση εγγραφής νέου χρήστη
- Πρωτόκολλα που δεν χρησιμοποιούν έμπιστη τρίτη οντότητα.

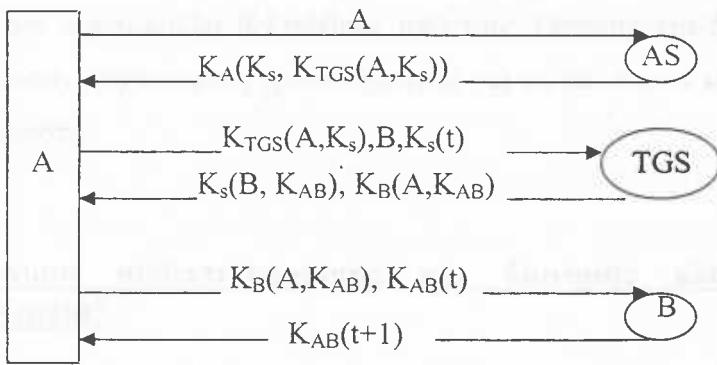
# Πρωτόκολλα που στηρίζονται στην χρήση μίας έμπιστης τρίτης οντότητας

## Το πρωτόκολλο “ΚΕΡΒΕΡΟΣ”

Το πρωτόκολλο Κέρβερος [22] αναπτύχθηκε από το MIT, στηρίζεται στην συμμετρική κρυπτογραφία και απαιτεί την ύπαρξη τριών εξυπηρετητών. Ενός Εξυπηρετητή Πιστοποίησης Αυθεντικότητας (AS) που επιβεβαιώνει την ταυτότητα των χρηστών κατά την είσοδό τους στο σύστημα, ενός Εξυπηρετητή Παραχώρησης Εισιτηρίων (TGS) σκοπός του οποίου είναι η έκδοση εισιτηρίων που αποτελούν αποδείξεις ταυτότητας της οντότητας και τέλος ενός Εξυπηρετητή που προσφέρει την παρεχόμενη υπηρεσία.

### **Περιγραφή Λειτουργίας**

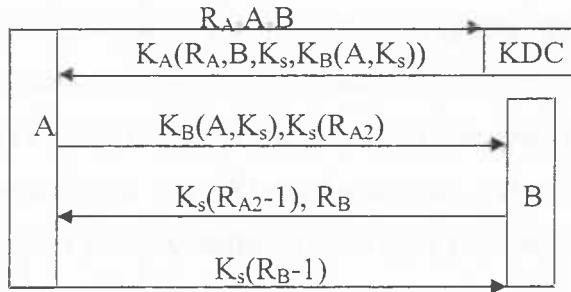
Αρχικά η προς αυθεντικοποίηση οντότητα έστω Α στέλνει το όνομά της στον AS χωρίς κρυπτογράφηση, στην συνέχεια ο AS επιστρέφει στην οντότητα Α ένα κλειδί συνόδου  $K_s$  και ένα εισιτήριο  $K_{TGS}(A, K_s)$  κρυπτογραφημένο με το μυστικό κλειδί της Α το  $K_A$ . Όταν το μήνυμα φτάσει στην Α το σύστημα ζητά από την Α να δώσει την συνθηματική της λέξη (password) για το σύστημα, από την λέξη αυτή δημιουργεί το μυστικό κλειδί  $K_A$  της Α και αποκρυπτογραφεί το μήνυμα . Η οντότητα Α αφού έχει πλέον ένα εισιτήριο μπορεί να ζητήσει από τον TGS να αρχίσει μία επικοινωνία με την οντότητα Β. Για να επιτευχθεί κάτι τέτοιο η Α στέλνει στον TGS ένα μήνυμα που περιέχει το εισιτήριο  $K_{TGS}(A, K_s)$ , το όνομα της οντότητας με την οποία θέλει να επικοινωνήσει έστω Β και μία σφραγίδα χρόνου  $K_s(t)$ . Τότε ο TGS αφού ελέγξει το μήνυμα του Α του επιστρέφει ένα κλειδί συνόδου με τον Β το  $K_{AB}$  κρυπτογραφημένο με το  $K_s$  δηλαδή  $K_s(B, K_{AB})$  και το ίδιο κλειδί κρυπτογραφημένο με το  $K_B$  δηλαδή  $K_B(A, K_{AB})$ . Στο σημείο αυτό η οντότητα Α μπορεί να στείλει το κλειδί συνόδου στον Β μαζί με μία σφραγίδα χρόνου  $K_{AB}(t)$  και η επικοινωνία τους μπορεί να αρχίσει με ασφάλεια μέσο του κοινού τους κλειδιού  $K_{AB}$ . Τα μηνύματα που ανταλλάσσονται στην παραπάνω διαδικασία φαίνονται αναλυτικά στο σχήμα που ακολουθεί.



### Το πρωτόκολλο πιστοποίησης αυθεντικότητας των Needham-Schoeder

Στο πρωτόκολλο αυτό γίνεται η χρήση ενός κέντρου διανομής κλειδιών (KDC). Έστω Α μία οντότητα που επιθυμεί να επικοινωνήσει με την οντότητα Β. Αρχικά η Α στέλνει στο KDC ένα μήνυμα που περιέχει το όνομά της δηλ. Α, το όνομα της οντότητας με την οποία θέλει να επικοινωνήσει δηλ. την Β και έναν τυχαίο μεγάλο αριθμό (nonce) τον  $R_A$  που χρησιμοποιείται για να διαβεβαιώσει ότι πρόκειται για ένα νέο μήνυμα και όχι επανάληψη κάποιου παλαιότερου. Στην συνέχεια το KDC στέλνει κρυπτογραφημένα με το κλειδί της Α ( $K_A$ ) ένα κλειδί συνόδου ( $K_s$ ) το οποίο θα χρησιμοποιήσει η Α στην επικοινωνία της με την οντότητα Β, τον αριθμό  $R_A$  και τέλος ένα εισιτήριο  $K_B(A, K_s)$  που θα αποσταλεί στον Β. Στο επόμενο βήμα η οντότητα Α στέλνει το εισιτήριο και έναν τυχαίο αριθμό  $R_{A2}$ , τον οποίο τον κρυπτογραφεί με το  $K_s$  στον Β. Όταν ο Β λάβει το μήνυμα στέλνει στον Α το  $K_s(R_{A2}-1)$ ,  $R_B$  και όταν λάβει το μήνυμα αυτό ο Α στέλνει το μήνυμα  $K_s(R_B-1)$ . Τα δύο τελευταία μηνύματα με τους  $R_A$ ,  $R_B$  σκοπό έχουν να πειστούν οι Α και Β ότι μιλάνε μεταξύ τους και ότι κάποιος τρίτος δεν έχει παραβιάσει την διαδικασία αυθεντικοποίησης.

Παρακάτω παρατίθεται η σειρά των μηνυμάτων που περιγράψαμε.



Το παραπάνω πρωτόκολλο βελτιώθηκε από τους Denning και Sacco[1], οι οποίοι χρησιμοποίησαν μηχανισμούς χρονοσήμανσης για να επιλύσουν κάποια προβλήματα προσωποποίησης.

## Το σύστημα αυθεντικοποίησης και διανομής κλειδιών συνόδου “KryptoKnight”

Το σύστημα KryptoKnight [23] αποτελεί μια παραλλαγή του συστήματος “Κέρβερος”. Οι οντότητες που εμπλέκονται στο πρωτόκολλο είναι οι χρήστες, τα προγράμματα και οι εξυπηρετητές αυθεντικοποίησης. Λέγοντας χρήστες εννοούμε όλες εκείνες τις δικτυακές οντότητες όπως εξυπηρετητές ή άτομα ή εφαρμογές που χρίζουν ανάγκη αυθεντικοποίησης. Με τον όρο προγράμματα εννοούμε όλες τις διαδικασίες που λαμβάνουν χώρα κατά την διάρκεια της αυθεντικοποίησης. Τέλος οι εξυπηρετητές αυθεντικοποίησης είναι έμπιστες τρίτες οντότητες σκοπός των οποίων είναι η έκδοση των διαπιστευτηρίων και ο έλεγχος αυτών κατά την διαδικασία αυθεντικοποίησης, προκειμένου δυο οντότητες να αυθεντικοποιηθούν αμοιβαία και να αποκτήσουν ένα κοινό μυστικό κλειδί συνόδου το οποίο θα χρησιμοποιήσουν στην μεταξύ τους επικοινωνία.

Το KryptoKnight αποτελείται από μία ομάδα πρωτοκόλλων που παρέχουν τις εξής υπηρεσίες:

- Αυθεντικοποίηση αμοιβαία ή όχι
- Διανομή μυστικού κλειδιού συνόδου μεταξύ δυο οντοτήτων (χρηστών)
- Αυθεντικοποίηση δεδομένων

Δεν θα αναλύσουμε τον τρόπο με τον οποίο προσφέρονται οι παραπάνω υπηρεσίες απλά θα σταθούμε σε ορισμένα πλεονεκτήματα της οικογένειας πρωτοκόλλων KryptoKnight σε αντιπαράθεση με το πρωτόκολλο Κέρβερος.

- Δεν απαιτεί την ύπαρξη μηχανισμών χρονοσήμανσης και καλά συντονισμένων ρολογιών γιατί γίνεται η χρήση nonces (ψευδοτυχαίων αριθμών) που αποδεικνύουν την μοναδικότητα ενός μηνύματος.
- Χρησιμοποιεί μηνύματα μικρότερου μήκους χωρίς περιττές πληροφορίες.

Παρά τα πλεονεκτήματά του KryptoKnight σε σχέση με το πρωτόκολλο Κέρβερος παραμένει ένα κεντρικοποιημένο σύστημα με όλα τα μειονεκτήματα που αυτό συνεπάγεται.



## Πρωτόκολλο αυθεντικοποίησης με την χρήση απιστοποίητων κλειδιών

Όπως ήδη έχουμε αναφέρει η χρήση ενός εξυπηρετητή αυθεντικοποίησης επιφορτισμένο με τον ρόλο της έμπιστης τρίτης οντότητας αποτελεί μία αξιόπιστη λύση για το πρόβλημα της αυθεντικοποίησης με τα γνωστά όμως πρόβλημα, δηλαδή ότι αποτελεί ένα κοινό σημείο συμφόρησης και αποτυχίας. Μία πρόταση που θα μετριάσει τα παραπάνω προβλήματα είναι των I-Lung Kao και Randy Chow [3]. Σύμφωνα με αυτή μπορεί να χρησιμοποιηθεί μία κλασική αρχιτεκτονική αυθεντικοποίησης με χρήση μίας έμπιστης τρίτης οντότητας. Όταν δύο οντότητες ξένες μεταξύ τους θέλουν να επικοινωνήσουν για πρώτη φορά τότε η αυθεντικοποίηση γίνεται με τον κλασικό τρόπο μέσω μίας έμπιστης τρίτης οντότητας. Με την μέθοδο που παρουσιάζουμε μετά την πρώτη επικοινωνία οι δύο οντότητες συμφωνούν ένα μυστικό κλειδί για επόμενη επικοινωνία το οποίο θα χρησιμοποιήσουν στην διαδικασία αυθεντικοποίησης.

Η μέθοδος αυτή δίνει μία λύση στα προβλήματα των κεντρικοποιημένων αρχιτεκτονικών που έχουμε αναφέρει παραπάνω, αφήνει όμως άλυτα προβλήματα όπως το κόστος για την δημιουργία ενός δικτύου εμπιστοσύνης μέσο έμπιστων τρίτων οντοτήτων σε παγκόσμια κλίμακα. Τέλος εισάγει το πρόβλημα διαχείρισης των μυστικών κλειδιών που έχουν συμφωνήσει οι οντότητες για μελλοντική επικοινωνία.

## Αυθεντικοποίηση με χρήση κρυπτογραφίας δημοσίου κλειδιού

Η αυθεντικοποίηση που στηρίζεται στην χρήση κρυπτογραφίας δημοσίου κλειδιού πλεονεκτεί σε σχέση με τις άλλες μεθόδους αυθεντικοποίησης επειδή δεν χρειάζεται την ανταλλαγή κάποιας μυστικής πληροφορίας μεταξύ των οντοτήτων που εμπλέκονται στην αυθεντικοποίηση. Μία οντότητα που προσπαθεί να αυθεντικοποιηθεί πρέπει να υπογράψει ψηφιακά με την χρήση του ιδιωτικού της κλειδιού μία συμβολοσειρά που επέλεξε τυχαία η οντότητα που επιθυμεί να την αυθεντικοποιήσει. Αν η τελευταία καταφέρει να επαληθεύσει την ψηφιακά υπογεγραμμένη απάντηση με το δημόσιο κλειδί της προς αυθεντικοποίηση οντότητας τότε η αυθεντικοποίηση έχει επιτυχώς ολοκληρωθεί.

Από την περιγραφή της παραπάνω διαδικασίας γίνεται αντιληπτή η ανάγκη να αντιστοιχηθεί με κάποιον τρόπο ένα δημόσιο κλειδί σε μία και μόνο οντότητα και η

ανάγκη να κρατηθεί το ιδιωτικό κλειδί της κάθε οντότητας μυστικό. Μία οντότητα μπορεί να είναι ένας άνθρωπος ή μια διαδικασία.

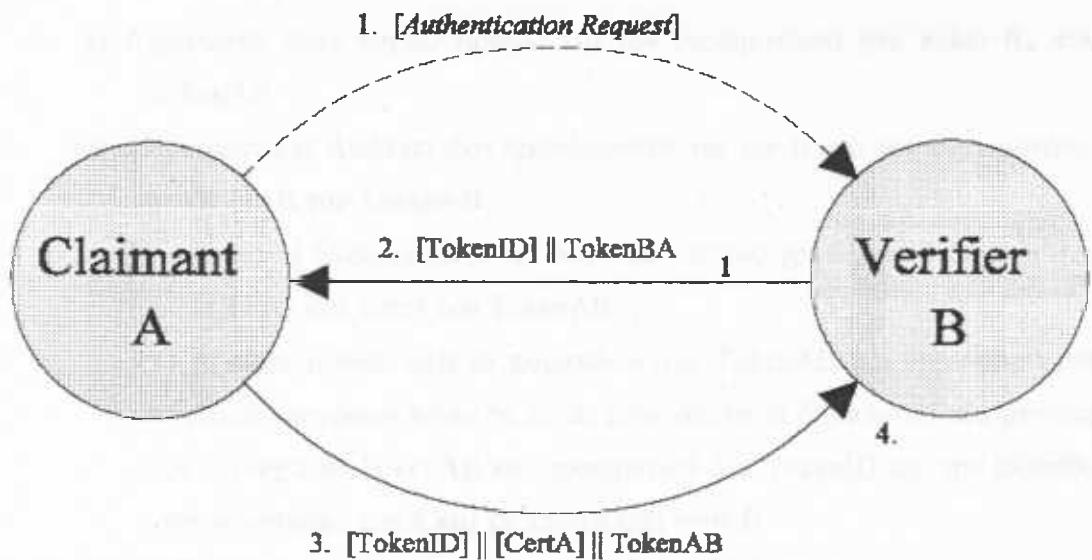
Η αντιστοίχηση ενός κλειδιού σε μία οντότητα μπορεί να γίνει με την βοήθεια μίας έμπιστης τρίτης οντότητας (TPP). Η υπηρεσία που θα παρέχει η TPP θα είναι η έκδοση ψηφιακών πιστοποιητικών ή η αντιστοίχιση του κλειδιού με την οντότητα μέσω κάποιου άλλου τρόπου για παράδειγμα την αποστολή του δημοσίου κλειδιού κάποιας προς αυθεντικοποίηση οντότητας μέσω ενός ασφαλούς καναλιού στην οντότητα που θέλει να την αυθεντικοποιήσει.

Στην περίπτωση που χρησιμοποιηθούν ψηφιακά πιστοποιητικά απαιτείται η ύπαρξη υποδομής δημοσίου κλειδιού (PKI). Λέγοντας υποδομή δημοσίου κλειδιού εννοούμε ένα σύνολο από έμπιστες τρίτες οντότητες που ονομάζονται αρχές πιστοποίησης (Certification Authorities- CA) οι οποίες αληλοπιστοποιούνται είτε αμοιβαία είτε με κάποια δενδρική μορφή π.χ. αν η CA A πιστοποιεί την B και την C, τότε αν κάποια οντότητα X έχει ένα πιστοποιητικό από την B και θέλει να αυθεντικοποιηθεί από την C πρέπει να στείλει μια αλυσίδα πιστοποιητικών που περιέχει το πιστοποιητικό που έχει λάβει από την B το πιστοποιητικό της B και το πιστοποιητικό της A. Οι CAs εκτός από την έκδοση πιστοποιητικών αναλαμβάνουν την ευθύνη για την διαχείριση των πιστοποιητικών που εκδίδουν καθ' όλο τον κύκλο ζωής τους. Αν για οποιοδήποτε λόγο ένα πιστοποιητικό πάψει να ισχύει τότε τοποθετείται σε μία λίστα ανακληθέντων πιστοποιητικών η οποία δημοσιεύεται. Κάθε φορά που λαμβάνεται μία αλυσίδα πιστοποιητικών απαιτείται ο έλεγχος εγκυρότητας του κάθε πιστοποιητικού. Ο έλεγχος ενός πιστοποιητικού περιλαμβάνει τον έλεγχο την ημερομηνία λήξης του, τις δυνατότητες χρήσεις του, έλεγχο αν έχει τοποθετηθεί στην λίστα ανακληθέντων πιστοποιητικών και αν έχει εκδοθεί από μία έγκυρη αρχή πιστοποίησης.

Θα παρουσιάσουμε στην συνέχεια δυο πρωτόκολλα που στηρίζονται στην χρήση κρυπτογραφίας δημόσιου κλειδιού. Το πρώτο εξασφαλίζει αυθεντικοποίηση της μιας από τις δυο εμπλεκόμενες οντότητες ενώ το άλλο αμοιβαία αυθεντικοποίηση μεταξύ των εμπλεκόμενων οντοτήτων. Τα πρωτόκολλα αυτά [24] έχουν δημοσιευθεί από την NIST (National Institute of Standards and Technology) που είναι εθνική αρχή των Ηνωμένων πολιτειών της Αμερικής, υπεύθυνη για πρότυπα και τεχνολογίες.

## Πρωτόκολλο μονομερούς αυθεντικοποίησης

Αρχικά θα παραθέσουμε σχηματικά τα μηνύματα που ανταλλάσσονται μεταξύ της οντότητας που προσπαθεί να αυθεντικοποιηθεί (claimant) έστω Α και της οντότητας που προσπαθεί να την αυθεντικοποιήσει (verifier) έστω Β.



Το πρωτόκολλο που περιγράφουμε δίνει την δυνατότητα παραμετροποίησεων. Ορισμένα από τα μηνύματα ή μέρος των μηνυμάτων μπορεί να είναι προαιρετικά, όταν συμβαίνει κάτι τέτοιο θα υπάρχει σχετική αναφορά.

Θα αναλύσουμε στην συνέχεια την ροή των μηνυμάτων.

1. [Προαιρετικό] Ο Α διαλέγει την οντότητα στην οποία θέλει να αποδείξει την ταυτότητά του, έστω ότι η οντότητα αυτή είναι η Β και κάνει μία αίτηση αυθεντικοποίησης.
2. Ο Β αποφασίζει αν θα συνεχίσει την διαδικασία αυθεντικοποίησης. Αν επιλέξει να συνεχίσει τότε:
  - α) Δημιουργεί ένα τυχαίο αριθμό και τον ενσωματώνει στο πεδίο  $R_B$  του  $TokenBA_1$ , και αποθηκεύει τον αριθμό αυτό για μελλοντική χρήση.
  - β) [Προαιρετικό] Δημιουργεί άλλα δεδομένα που περικλείονται στο πεδίο  $Text1$  του  $TokenBA_1$ , τα δεδομένα αυτά μπορεί να εξαρτώνται από την εφαρμογή που χρησιμοποιεί το πρωτόκολλο.

Ο Β στέλνει στον Α το  $TokenBA_1$  και προαιρετικά ενσωματώνει στο μήνυμα και ένα  $TokenID$ . Το  $TokenID$  μπορεί να είναι ένας προσδιοριστής

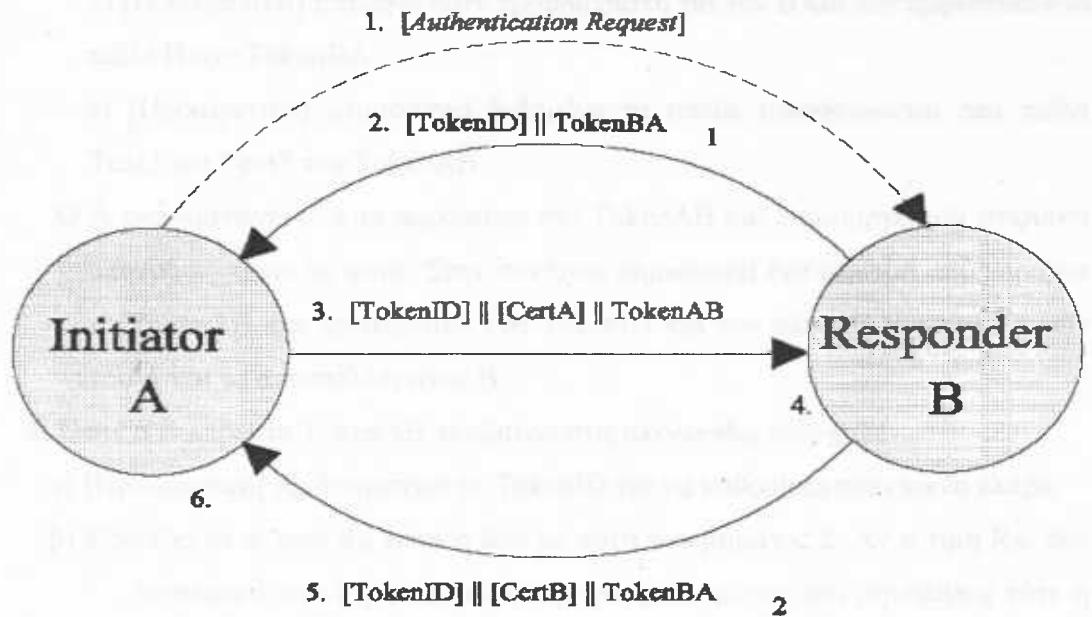
του token ή να ενσωματώνει πληροφορία για τον τύπο του πρωτοκόλλου που χρησιμοποιείται.

3. Λαμβάνοντας ο Α το μήνυμα που περιέχει το TokenBA<sub>1</sub> από τον Β
  - α) [Προαιρετικό] Χρησιμοποιεί το TokenID για να δει ποιό token έλαβε
  - β) [Προαιρετικό] Ανακτά την πληροφορία που περιέχει το Text1 του TokenBA<sub>1</sub>
  - γ) Δημιουργεί έναν τυχαίο αριθμό και τον ενσωματώνει στο πεδίο R<sub>A</sub> του TokenAB
  - δ) [Προαιρετικό] Διαλέγει ένα προσδιοριστή για τον Β και τον ενσωματώνει στο πεδίο B του TokenAB.
  - ε) [Προαιρετικό] Ενσωματώνει εφ' όσον κάτι τέτοιο χρειάζεται δεδομένα στα πεδία Text2 και Text3 του TokenAB  
Ο Α ενσωματώνει όλα τα παραπάνω στο TokenAB και δημιουργεί μία ψηφιακή υπογραφή πάνω σε αυτά. Στην συνέχεια δημιουργεί ένα μήνυμα που περιέχει το TokenAB και προαιρετικά ένα TokenID και την αλινσίδα πιστοποιητικών του Α και τα αποστέλλει στον Β
4. Όταν ο Β λάβει το μήνυμα που περιέχει το TokenAB τότε:
  - α) [Προαιρετικά] εξετάζει το TokenID
  - β) Εξετάζει αν η τιμή R<sub>B</sub> είναι η ίδια με αυτή του βήματος 2. Αν η τιμή R<sub>B</sub> δεν περιέχεται στο μη ψηφιακά υπογεγραμμένο μέρος του μηνύματος τότε η επαλήθευση γίνεται στο βήμα 4ε.
  - γ) Ο Β ανακτά το πιστοποιητικό του Α. Το πιστοποιητικό μπορεί να περιέχεται στις προαιρετικές πληροφορίες σε ένα από τα παραπάνω μηνύματα. Στην περίπτωση που δεν γίνεται χρήση πιστοποιητικών πρέπει ο Β με κάποιο άλλο τρόπο να αποκτήσει το δημόσιο κλειδί που αντιστοιχεί στον Α.
  - δ) Εφ' όσον γίνεται χρήση πιστοποιητικών ελέγχεται η εγκυρότητά τους.
  - ε) Γίνεται επαλήθευση της ψηφιακής υπογραφής του Α
  - στ) [Προαιρετικό] Ανακτά τα στοιχεία που περιέχονται στα πεδία Text2 και Text3

Η επιτυχής ολοκλήρωση των 4β και 4ε σημαίνει επιτυχή αυθεντικοποίηση του Α από τον Β.

## Πρωτόκολλο αμοιβαίας αυθεντικοποίησης

Μια σχηματική αναπαράσταση των μηνυμάτων που ανταλλάσσονται κατά την διάρκεια αυτού του πρωτοκόλλου είναι η παρακάτω.



Σε αυτό το πρωτόκολλο θεωρούμε την οντότητα Α που άρχισε την διαδικασία αυθεντικοποίησης (initiator) και την οντότητα Β (responder).

Η ανταλλαγή των μηνυμάτων του παραπάνω σχήματος θα περιγραφεί στην συνέχεια.

- 1) [Προαιρετικό] Η οντότητα Α διαλέγει την οντότητα Β με την οποία επιθυμεί αμοιβαία αυθεντικοποίηση και της αποστέλλει μία αίτηση αυθεντικοποίησης.
- 2) Η Β αποφασίζει αν επιθυμεί να συνεχίσει την αμοιβαία αυθεντικοποίηση ή όχι. Αν αποφασίσει να λάβει μέρος στην διαδικασία τότε:
  - α) Δημιουργεί ένα τυχαίο αριθμό και τον εμφυτεύει στο πεδίο  $R_B$  του  $\text{TokenBA}_1$ .
  - β) [Προαιρετικό] Δημιουργεί δεδομένα και τα εμφυτεύει στο πεδίο  $\text{Text1}$  του  $\text{TokenBA}_1$ .

Στην συνέχεια η οντότητα Β αποστέλλει στην Α ένα μήνυμα που περιέχει το  $\text{TokenBA}_1$  και ένα προαιρετικό  $\text{TokenID}$ .

- 3) Με την λήψη του μηνύματος του βήματος 2 ο Α προβαίνει στις ακόλουθες ενέργειες.

- α) [Προαιρετικό] Χρησιμοποιεί το TokenID για να καθορίσει ποιο token έλαβε.
- β) [Προαιρετικό] Ανακτά τις πληροφορίες από το Text1 του TokenBA<sub>1</sub>
- γ) Δημιουργεί ένα τυχαίο αριθμό και τον εμφυτεύει στο πεδίο R<sub>A</sub> του TokenAB. Η τιμή του R<sub>A</sub> αποθηκεύεται για μελλοντική χρήση.
- δ) [Προαιρετικό] Επιλέγει έναν προσδιοριστή για τον B και τον εμφυτεύει στο πεδίο B του TokenBA.
- ε) [Προαιρετικό] Δημιουργεί δεδομένα τα οποία τοποθετούνται στα πεδία Text2 και Text3 του TokenAB

Ο A ενσωματώνει όλα τα παραπάνω στο TokenAB και δημιουργεί μία ψηφιακή υπογραφή πάνω σε αυτά. Στην συνέχεια δημιουργεί ένα μήνυμα που περιέχει το TokenAB και προαιρετικά ένα TokenID και την αλυσίδα πιστοποιητικών του A και τα αποστέλλει στον B.

- 4) Όταν ο B λάβει το TokenAB προβαίνει στις ακόλουθες ενέργειες
- α) [Προαιρετικό] Χρησιμοποιεί το TokenID για να καθορίσει ποιο token έλαβε.
  - β) Εξετάζει αν η τιμή R<sub>B</sub> είναι η ίδια με αυτή του βήματος 2. Αν η τιμή R<sub>B</sub> δεν περιέχεται στο μη ψηφιακά υπογεγραμμένο μέρος του μηνύματος τότε η επαλήθευση γίνεται στο βήμα 4ε.
  - γ) Ο B ανακτά το πιστοποιητικό του A. Το πιστοποιητικό μπορεί να περιέχεται στις προαιρετικές πληροφορίες σε ένα από τα παραπάνω μηνύματα. Στην περίπτωση που δεν γίνεται χρήση πιστοποιητικών πρέπει ο B με κάποιο άλλο τρόπο να αποκτήσει το δημόσιο κλειδί που αντιστοιχεί στον A.
  - δ) Εφ' όσον γίνεται χρήση πιστοποιητικών ελέγχεται η εγκυρότητά τους.
  - ε) Γίνεται επαλήθευση της ψηφιακής υπογραφής του A
  - στ) [Προαιρετικό] Ανακτά τα στοιχεία που περιέχονται στο πεδία Text2 και Text3

Η επιτυχής ολοκλήρωση των 4β και 4ε σημαίνει επιτυχή αυθεντικοποίηση του A από τον B.

- 5) Στην συνέχεια ο B
- α) [Προαιρετικό] Διαλέγει ένα προσδιοριστή για τον A και τον εμφυτεύει στο πεδίο A του TokenBA<sub>2</sub>.
  - β) [Προαιρετικό] Ο B δημιουργεί το περιεχόμενο των πεδίων Text4 και Text5

Ο B ενσωματώνει όλα τα παραπάνω στο TokenBA<sub>2</sub> και δημιουργεί μία ψηφιακή υπογραφή πάνω σε αυτά. Στην συνέχεια δημιουργεί ένα μήνυμα που

περιέχει το TokenBA<sub>2</sub> και προαιρετικά ένα TokenID και την αλυσίδα πιστοποιητικών του B και τα αποστέλλει στον A.

- 6) Με την λήψη του TokenBA<sub>2</sub> ο A προβαίνει στις ακόλουθες ενέργειες.
- α) [Προαιρετικό] Χρησιμοποιεί το TokenID για να καθορίσει ποιο token έλαβε.
  - β) Εξετάζει αν η τιμή R<sub>A</sub> είναι η ίδια με αυτή του βήματος 3. Αν η τιμή R<sub>A</sub> δεν περιέχεται στο μη ψηφιακά υπογεγραμμένο μέρος του μηνύματος τότε η επαλήθευση γίνεται στο βήμα 6στ.
  - γ) Εξετάζει αν η τιμή R<sub>B</sub> είναι η ίδια με αυτή του πεδίου R<sub>B</sub> στο TokenBA<sub>2</sub>. Αν η τιμή R<sub>B</sub> δεν περιέχεται στο μη ψηφιακά υπογεγραμμένο μέρος του μηνύματος τότε η επαλήθευση γίνεται στο βήμα 6στ.
  - δ) Ο A ανακτά το πιστοποιητικό του B. Το πιστοποιητικό μπορεί να περιέχεται στις προαιρετικές πληροφορίες σε ένα από τα παραπάνω μηνύματα. Στην περίπτωση που δεν γίνεται χρήση πιστοποιητικών πρέπει ο A με κάποιο άλλο τρόπο να αποκτήσει το δημόσιο κλειδί που αντιστοιχεί στον B.
  - ε) Εφ' όσον γίνεται χρήση πιστοποιητικών ελέγχεται η εγκυρότητά τους.
  - στ) Γίνεται επαλήθευση της ψηφιακής υπογραφής του B
- ζ) [Προαιρετικό] Ανάκτηση των δεδομένων των πεδίων Text4 και Text5.  
Η επιτυχής ολοκλήρωση των 6β και 6στ σημαίνει ο B κατάφερε να αυθεντικοποιηθεί επιτυχώς από τον A και κατά συνέπεια οι οντότητες A και B έχουν επιτύχει αμοιβαία αυθεντικοποίηση.

Τα κρυπτοσυστήματα και συστήματα ψηφιακών υπογραφών που στηρίζονται στην υποδομή δημοσίου κλειδιού έχουν το μειονέκτημα ότι απαιτούν την ύπαρξη μιας online έμπιστης τρίτης οντότητας. Πλεονεκτούν όμως σε σχέση με τα πρωτόκολλα που στηρίζονται στην χρήση συμμετρικής κρυπτογραφίας γιατί δεν απαιτούν την ανταλλαγή κάποιας μυστικής πληροφορίας μεταξύ των εμπλεκομένων οντότητων.

# Πρωτόκολλα που απαιτούν την ύπαρξη μίας έμπιστης τρίτης οντότητας μόνο κατά την φάση εγγραφής νέου χρήστη

## Πρωτόκολλο αυθεντικοποίησης με χρήση πολλών διαχειριστών ασφάλειας

Περιγράψαμε πρωτόκολλα στα οποία τη συνολική ευθύνη για την αυθεντικοποίηση μίας οντότητας την αναλάμβανε μία έμπιστη τρίτη οντότητα. Σε περίπτωση μη διαθεσιμότητας της έμπιστης τρίτης οντότητας το όλο σύστημα ήταν καταδικασμένο σε αποτυχία.

Θα παρουσιάσουμε τώρα ένα πρωτόκολλο των Woei-Jiunn, Tsaurf Shi-Jinn Horngt, Chia-Ho Chen[2] στο οποίο η ευθύνη αυθεντικοποίησης κατανέμεται σε πολλούς διαχειριστές ασφάλειας. Το πρωτόκολλο αυτό περιέχει τις ακόλουθες φάσεις:

- Εγγραφή χρήστη
- Αυθεντικοποίηση χρήστη
- Πρωτόκολλο ελέγχου πρόσβασης

Εστω ότι ένας χρήστης επιθυμεί να έχει πρόσβαση στα αρχεία ενός εξυπηρετητή αρχείων(S).

Κατά την φάση εγγραφής ενός χρήστη, ο χρήστης υποβάλει τα πιστοποιητικά του στον S και σε συνεργασία με αυτόν δημιουργεί ένα συμμετρικό κλειδί το οποίο αποθηκεύεται σε μία έξυπνη κάρτα.

Στην συνέχεια ο S σπάει το κλειδί με κατάλληλες μαθηματικές τεχνικές σε τόσο μέρη όσοι είναι και οι διαχειριστές ασφάλειας (έστω n) και τους μοιράζει τα κομμάτια του κλειδιού.

Κατά την αυθεντικοποίηση, ο χρήστης με το κατάλληλο λογισμικό σπάει το κλειδί που έχει αποθηκευμένο στην έξυπνη κάρτα σε n μέρη και στέλνει κάθε κομμάτι σε καθέναν από τους διαχειριστές ασφάλειας. Αυτοί ελέγχουν αν το κομμάτι που τους έχει σταλεί είναι το ίδιο με αυτό που είχαν λάβει κατά την φάση της εγγραφής από τον εξυπηρετητή S και σε περίπτωση που κάποιος αριθμός από αυτούς έστω m < n απαντήσουν θετικά ως προς την επαλήθευση, τότε ο χρήστης εισέρχεται στο σύστημα. Με τον τρόπο αυθεντικοποίησης που μόλις περιγράψαμε αν n-m διαχειριστές ασφάλειας δεν είναι διαθέσιμοι τότε το σύστημα θα συνεχίσει να λειτουργεί απρόσκοπτα. Εφόσον ολοκληρωθεί με επιτυχία η διαδικασία της

αυθεντικοποίησης τότε ο χρήστης και ο S λαμβάνουν από τους διαχειριστές ασφάλειας ένα κοινό μυστικό κλειδί συνόδου της μορφής  $K_{si} = (K_{si,1}, K_{si,2}, \dots, K_{si,m})$ .

Αφού ολοκληρωθεί η διαδικασία αυθεντικοποίησης και αποκτηθεί από τον εξυπηρετητή αρχείων και από το χρήστη το κοινό κλειδί συνόδου τότε ο εξυπηρετητής κρυπτογραφεί με το μυστικό κλειδί τις πληροφορίες που θα του ζητήσει ο χρήστης και τις αποστέλλει σ' αυτόν.

Η διαδικασία πρόσβασης μπορεί να τροποποιηθεί ώστε να υπάρχει η δυνατότητα για διανομή αρχείων σε μία ομάδα χρηστών.

### Πρωτόκολλα αυθεντικοποίησης που στηρίζονται στην ταυτότητα του χρήστη

Η λογική των πρωτοκόλλων που στηρίζονται στην ταυτότητα του χρήστη είναι να δημιουργήσουν κλειδιά που ενσωματώνουν την ταυτότητα του χρήστη. Η χρήση ενός τέτοιου κλειδιού μπορεί να αυθεντικοποίησει άμεσα τον κάτοχο του χωρίς την μεσολάβηση μίας έμπιστης τρίτης οντότητας. Παρακάτω δίνουμε την περιγραφή δύο τέτοιων σχημάτων.

#### Σχήμα 1<sup>o</sup>

Ένα σχήμα που στηρίζεται στην ύπαρξη μίας έμπιστης τρίτης οντότητας μοναδικός σκοπός της οποίας είναι η παραγωγή ιδιωτικών κλειδιών που θα αποθηκεύονται σε μία έξυπνη κάρτα (smart card) προτάθηκε από τον Adi Shamir[25]. Με χρήση ασύμμετρης κρυπτογραφίας και του ιδιωτικού κλειδιού ο κάτοχος μιας έξυπνης κάρτας μπορεί να υπογράψει ψηφιακά ή να κρυπτογραφήσει μηνύματα. Η σημαντικότερη διαφορά της εν' λόγο αρχιτεκτονικής και των αρχιτεκτονικών που στηρίζονται στην υποδομή δημοσίου κλειδιού είναι ότι το δημόσιο κλειδί στο σχήμα που περιγράφουμε είναι η ταυτότητα της οντότητας. Η ταυτότητα μπορεί να είναι το όνομά και η διεύθυνσή της οντότητας ή ο αριθμός φορολογικού μητρώου της (εάν πρόκειται για άνθρωπο). Το ιδιωτικό κλειδί που αντιστοιχεί στο δημόσιο (ταυτότητα) που η οντότητα έχει επιλέξει υπολογίζεται από την έμπιστη τρίτη οντότητα και αποθηκεύεται στην έξυπνη κάρτα.

Η ασφάλεια της αρχιτεκτονικής έγκειται σε τέσσερις παράγοντες:

- Στην ασφάλεια των συναρτήσεων κρυπτογράφησης
- Στην εχεμύθεια της έμπιστης τρίτης οντότητας που έχει αναλάβει το ρόλο δημιουργίας κλειδιών
- Στην αξιοπιστία των έλεγχων της έμπιστης τρίτης οντότητας κατά την φάση της εγγραφής ενός νέου χρήστη ώστε να διαπιστώσει αν η οντότητα είναι όντως αυτή που υπαινίσσεται.
- Στα μέτρα που λαμβάνει κάθε οντότητα προκειμένου να προστατέψει την έξυπνη κάρτα της.

Η αρχιτεκτονική αυτή δίνει εξαιρετική λύση στο πρόβλημα της αυθεντικοποίησης στην περίπτωση κλειστών groups για παράδειγμα μεταξύ των εργαζομένων μίας εταιρίας ή των server της.

Το σημαντικότερο πρόβλημα της παραπάνω αρχιτεκτονικής είναι ότι απαιτεί ένα σχήμα δημοσίου κλειδιού με δυο επιπλέον δυνατότητες.

- Όταν ένας αριθμός έστω k ο οποίος χρησιμοποιείται για την δημιουργία του ιδιωτικού κλειδιού είναι γνωστός τότε πρέπει τα ιδιωτικά κλειδιά να μπορούν να υπολογιστούν εύκολα για έναν όχι αμελητέο αριθμό από πιθανά δημόσια κλειδιά.
- Να είναι αδύνατος ο υπολογισμός του k αν κάποιος γνωρίζει ζεύγη δημόσιων – ιδιωτικών κλειδιών που δημιουργήθηκαν με βάση το k.

Ο δυστυχώς ο RSA δεν υποστηρίζει τις παραπάνω δύο ιδιότητες και δεν μπορεί να χρησιμοποιηθεί για το σκοπό αυτό. Οι δημιουργοί του σχήματος έχουν αναπτύξει ένα κρυπτοσύστημα το οποίο υποστηρίζει ψηφιακές υπογραφές δεν υποστηρίζει όμως κρυπτογράφηση.

## Σχήμα 2<sup>o</sup>

Την ιδέα της αυθεντικοποίησης με χρήση μίας έμπιστης τρίτης οντότητας που συμμετέχει μόνο κατά την φάση εγγραφής ενός νέου χρήστη υλοποιεί ένα πρωτόκολλο που έχει δημιουργηθεί από τους Wen – Her Yang και Hun – Min Sun [26]. Το πρωτόκολλο αυτό έχει το πλεονέκτημα ότι η αμοιβαία αυθεντικοποίηση και η ανταλλαγή ενός μυστικού κλειδιού συνόδου μεταξύ δύο οντοτήτων μπορεί να ολοκληρωθεί μόνο με την ανταλλαγή δύο μηνυμάτων. Το πρωτόκολλο κάνει χρήση ενός σχήματος που στηρίζεται στην ταυτότητα της προς αυθεντικοποίηση οντότητας (ID – Based), για την φάση της αυθεντικοποίησης και την συμφωνία ενός μυστικού κλειδιού συνόδου μεταξύ των εμπλεκόμενων οντοτήτων ενώ χρησιμοποιεί

συμμετρική κρυπτογραφία για να διασφαλίσει την επικοινωνία μεταξύ των οντοτήτων.

Το πρωτόκολλο αποτελείται από δύο φάσεις:.

- Την αρχική φάση (initial phase)
- Την φάση αυθεντικοποίησης (authentication phase)

Κατά την αρχική φάση μία έμπιστη τρίτη οντότητα (information center – IC) δημιουργεί τα διαπιστευτήρια των οντοτήτων που επιθυμούν να εμπλακούν στην διαδικασία αυθεντικοποίησης. Τα βήματα για την δημιουργία ενός διαπιστευτηρίου είναι τα ακόλουθα.

1. Το IC διαλέγει δύο μεγάλους πρώτους αριθμούς  $p$ ,  $q$  και υπολογίζει τον  $n = p * q$ .
2. Το IC ανακτά την μυστική πληροφορία  $d$  την οποίο γνωρίζει το IC και μόνον αυτό μέσο του υπολογισμού

$$3 * d \pmod{(p-1)(q-1)} = 1$$

3. Το IC υπολογίζει έναν ακέραιο  $g$  που είναι ένα πρωταρχικό στοιχείο και στις συναρτήσεις  $GF(p)$  και  $GF(q)$ . Όπου  $g$  είναι η δημόσια πληροφορία του IC.
4. Έστω ότι  $ID_i$  είναι η ταυτότητα του χρήστη  $i$
5. Το IC επιλέγει μία συνάρτηση σύνοψης και υπολογίζει την εκτεταμένη ταυτότητα (Extended identity –  $EID_i$ ) με τον ακόλουθο τρόπο.

$$EID_i \equiv f(ID_i) \pmod{2^N} \equiv (EID_{i1}, EID_{i2}, \dots, EID_{iN})$$

όπου  $N$  το μήκος σε bit του  $EID$ .

6. Μετά τον υπολογισμό του  $EID_i$  υπολογίζεται από τον IC η μυστική πληροφορία  $S_i$  που αντιστοιχεί στο χρήστη  $i$  ως εξής:

$$S_i \equiv EID_i^d \pmod{n}$$

7. Η initial φάση ολοκληρώνεται με την αποστολή των  $(n, g, f(x), S_i)$  πίσω στον χρήστη  $i$  μέσω ενός ασφαλούς καναλιού. Με την λήψη της παραπάνω πληροφορίας ο χρήστης  $i$  πρέπει να κρατήσει μυστικό το  $S_i$  και να αποθηκεύσει την δημόσια πληροφορία  $(n, g, f(x))$ .

Μετά το τέλος της παραπάνω διαδικασίας το IC πρέπει να αποθηκεύσει την μυστική του πληροφορία  $d$  και μπορεί να σταματήσει την λειτουργία του. Οι ακέραιοι  $p, q$  δεν χρειάζονται από εδώ και στο εξής και πρέπει να καταστραφούν με κάποιο ασφαλή τρόπο. Όταν ένας νέος χρήστης ζητά διαπιστευτήριο για την είσοδο του στο

σύστημα το IC ενεργοποιείται ξανά και τα βήματα 5 – 7 της παραπάνω διαδικασίας επαναλαμβάνονται.

Κατά την φάση της αυθεντικοποίησης χρειάζεται η ανταλλαγή μόνο δύο μηνυμάτων για να ολοκληρωθεί η διαδικασία. Τα βήματα της διαδικασίας αυθεντικοποίησης παρατίθενται παρακάτω.

1. Όταν ένας χρήστης  $i$  επιθυμεί να επικοινωνήσει με τον χρήστη  $j$  τότε δημιουργεί ένα τυχαίο αριθμό  $r_i$  και υπολογίζει τους αριθμούς  $X_i$  και  $Y_i$

$$X_i \equiv g^{3^*r_i} \pmod{n}$$

$$Y_i \equiv S_i * \text{time}_i * g^{2^*r_i} \pmod{n}$$

Όπου  $\text{time}_i$  είναι η ώρα που υπολόγισε ο  $i$  τους δύο ακέραιους.

2. Ο χρήστης  $i$  στέλνει τους παραπάνω δύο αριθμούς, το  $ID_i$  και το  $\text{time}_i$  στον  $j$
3. Με την λήψη του παραπάνω μηνύματος ο χρήστης  $j$  συγκρίνει την τιμή  $\text{time}_i$  με την τρέχουσα τοπική του ώρα. Αν η διαφορά είναι μικρότερη μίας προκαθορισμένης διάρκειας το μήνυμα θεωρείται έγκυρο. Στην περίπτωση ενός δικτύου που τα ρολόγια δεν είναι συγχρονισμένα, για να μην απορριφθεί ένα έγκυρο μήνυμα η σύγκριση ώρας μπορεί να παραληφθεί. Στην συνέχεια ο χρήστης  $j$  υπολογίζει το  $EID_i = f(ID_i)$  και ελέγχει αν ισχύει η εξίσωση.

$$EID_i * \text{time}_i^3 = Y_i^3 / X_i^2$$

4. Αν η εξίσωση ισχύει τότε ο χρήστης  $j$  πείθεται ότι το μήνυμα έχει σταλεί από τον  $i$  και κρατά την ποσότητα  $X_i$  για τον υπολογισμό του μυστικού κλειδιού συνόδου. Στην συνέχεια ο  $j$  δημιουργεί έναν τυχαίο αριθμό  $r_j$  και υπολογίζει τους ακέραιους  $X_j$  και  $Y_j$

$$X_j \equiv g^{3^*r_j} \pmod{n}$$

$$Y_j \equiv S_j * \text{time}_j * g^{2^*r_j} \pmod{n}$$

5. Ο χρήστης  $j$  στέλνει στον  $i$  τους δύο παραπάνω αριθμούς μαζί με την ταυτότητα του  $ID_j$  και τον χρόνο  $\text{time}_j$ .
6. Όταν ο χρήστης  $i$  λάβει το παραπάνω μήνυμα ελέγχει αν η τιμή του  $\text{time}_j$  είναι ίδια με αυτή που είχε αποστείλει ο ίδιος. Αν είναι υπολογίζει την ποσότητα  $EID_j = f(ID_j)$  και ελέγχει αν ισχύει η εξίσωση.

$$EID_j * \text{time}_j = Y_j^3 / X_j^2$$

7. Αν η εξίσωση ισχύει τότε ο  $i$  υπολογίζει το μυστικό κλειδί συνόδου ως εξής.

$$K_{ij} = X_j^{r_i} = g^{3^*r_i * r_j}$$



8. Ομοίως ο  $j$  υπολογίζει το κλειδί συνόδου.

$$K_{ji} = X_j^{ij} = g^{3^*i^*j}$$

9. Με το κοινό μυστικό κλειδί συνόδου οι χρήστες  $i$  και  $j$  μπορούν πλέον να κρυπτογραφούν και να αποκρυπτογραφούν τα μηνύματα που ανταλλάσσουν.

Τα σχήματα κρυπτογράφησης που στηρίζονται στην χρήση μίας έμπιστης τρίτης οντότητας μόνο κατά την φάση εγγραφής νέου χρήστη είναι κατάλληλα για αυθεντικοποίηση οντοτήτων στα κλειστά πλαίσια μίας εταιρίας ή ενός οργανισμού. Μέχρι στιγμής δεν έχει υλοποιηθεί ένα ID-based σχήμα που να επιτρέπει συνεργασία μεταξύ έμπιστων τρίτων οντοτήτων αυτού του τύπου, ώστε να προκύψει μία αρχιτεκτονική με δυνατότητες υλοποίησης σε παγκόσμια κλίμακα με χαρακτηριστικά παρόμοια με αυτά της αυθεντικοποίησης με χρήση κρυπτογραφίας δημόσιου κλειδιού. Ένα επιπλέον σημείο που παραμένει αδιευκρίνιστο είναι το τι γίνεται στην περίπτωση που τα διαπιστευτήρια που παράγονται με αρχιτεκτονικές αυτού του τύπου πρέπει να αλλαχθούν π.χ. στην περίπτωση κλοπής ή απώλεια της μυστικής πληροφορίας που κατέχει κάθε οντότητα.

## Πρωτόκολλα που δεν χρησιμοποιούν έμπιστη τρίτη οντότητα

### Το πρωτόκολλο PGP

Στα συστήματα αυθεντικοποίησης που γίνεται χρήση ασύμμετρης κρυπτογραφίας το σημαντικότερο πρόβλημα είναι η αντιστοίχηση ενός δημοσίου κλειδιού σε μία οντότητα. Μία λύση στο πρόβλημα αυτό όπως έχουμε ήδη αναφέρει καλείται να δώσει η χρήση υποδομής δημοσίου κλειδιού μία άλλη ιδέα είναι η δημιουργία ενός ιστού εμπιστοσύνης μεταξύ των χρηστών. Η ιδέα αυτή υλοποιείται στο σύστημα PGP (Pretty Good Privacy) [27][28].

Σκοπός του PGP είναι να κάνει την κρυπτογραφία διαθέσιμη στις πλατειές μάζες, έχοντας την αρχή αυτή κατά νου το PGP “σπάει” την παραδοσιακή ιεραρχική αρχιτεκτονική εμπιστοσύνης. Στο PGP δεν υπάρχει μία κεντρική οντότητα που εμπιστεύονται όλοι, αλλά οι ίδιοι οι χρήστες υπογράφουν τα πιστοποιητικά άλλων χρηστών ανάλογα με το αν τους εμπιστεύονται ή όχι. Για παράδειγμα αν η Alice



υπογράφει ψηφιακά το πιστοποιητικό του Bob, τότε ο Bob προωθεί το πιστοποιητικό στην Carol η οποία γνωρίζει την Alice και εμπιστεύεται τα πιστοποιητικά που αυτή υπογράφει. Το πιστοποιητικό του Bob θα μπορούσε να έχει πολλές υπογραφές από άλλους χρήστες και με τον τρόπο αυτό να είναι ευρύτερα αποδεκτό. Στην περίπτωση που η Carol, δεν έβρισκε μεταξύ των ψηφιακών υπογραφών του πιστοποιητικού του Bob, την υπογραφή ενός χρήστη, τον οποίο εμπιστεύεται να υπογράφει πιστοποιητικά άλλων (introducer), τότε δεν θα έκανε αποδεκτό το πιστοποιητικό του Bob.

Θα παρουσιάσουμε παρακάτω ποιό αναλυτικά το μοντέλο εμπιστοσύνης του PGP. Στο PGP όπως έχουμε ήδη αναφέρει γίνεται χρήση πιστοποιητικών.

Ένα PGP πιστοποιητικό περιέχει:

- Την ταυτότητα (ID) του χρήστη, σαν ID χρησιμοποιείται συνήθως η διεύθυνση ηλεκτρονικού ταχυδρομείου του χρήστη
- Το δημόσιο κλειδί του χρήστη
- Το ID του δημόσιου κλειδιού
- Η ημερομηνία δημιουργίας του πιστοποιητικού

Πρέπει να σημειώσουμε ότι στο δίκτυο εμπιστοσύνης του PGP κάθε χρήστης προσδιορίζεται από το ID του δημόσιου κλειδιού και όχι από το προσωπικό του ID. Αυτό γίνεται γιατί η αντιστοίχηση μεταξύ ID ή ονόματος χρήστη γίνεται αυθαίρετα και το όνομα ή το ID μπορεί να μην είναι τα πραγματικά στοιχεία του χρήστη.

Η εμπιστοσύνη στο PGP έχει δυο παραμέτρους

- Αξιοπιστία πιστοποιητικού
- Εμπιστοσύνη σε μία οντότητα να υπογράφει πιστοποιητικά άλλων.

### *Αξιοπιστία πιστοποιητικού*

Η αξιοπιστία ενός πιστοποιητικού προσδιορίζει το κατά πόσο είμαστε σίγουροι ότι η αντιστοίχηση μεταξύ του ID και του δημοσίου κλειδιού είναι σωστή. Στο PGP υπάρχουν τρία επίπεδα αξιοπιστίας

- Απροσδιόριστο (undefined). Στο επίπεδο αυτό δεν μπορούμε να ξέρουμε αν το δημόσιο κλειδί είναι έγκυρο ή όχι.
- Οριακό (marginal). Σε αυτό το επίπεδο το δημόσιο κλειδί πρέπει να είναι έγκυρο άλλα δεν υπάρχει απόλυτη βεβαιότητα.



- Πλήρης (complete). Στο επίπεδο αυτό είμαστε σίγουροι ότι το δημόσιο κλειδί είναι έγκυρο.

### **Εμπιστοσύνη σε μία οντότητα να υπογράφει πιστοποιητικά άλλων**

Μία οντότητα με πλήρες αξιόπιστο δημόσιο κλειδί δεν σημαίνει ότι είναι κατάλληλη να υπογράφει άλλα δημόσια κλειδιά. Η καταλληλότητα ενός κλειδιού να υπογράφει κλειδιά ταξινομείται στο PGP σε τέσσερα επίπεδα.

- Πλήρης (full). Το δημόσιο κλειδί απολαμβάνει πλήρη εμπιστοσύνη να υπογράφει άλλα κλειδιά.
- Οριακό (marginal). Το δημόσιο κλειδί μπορεί να υπογράψει άλλα δημόσια κλειδιά αλλά δεν απολαμβάνει πλήρους εμπιστοσύνης.
- Αναξιόπιστο (untrustworthy). Το δημόσιο κλειδί δεν θεωρείται αξιόπιστο να υπογράφει άλλα κλειδιά
- Άγνοια (don't know). Δεν υπάρχει πληροφορία σχετικά με την αξιοπιστία του κλειδιού να υπογράφει άλλα δημόσια κλειδιά.

Τα παραπάνω επίπεδα δεν είναι ρητά, απλά αποτελούν έναν οδηγό για το πόσο εμπιστοσύνη μπορεί να έχει κάποιος σε κάθε δημόσιο κλειδί. Το PGP χρησιμοποιεί δύο μεταβλητές τις **COMPLETES\_NEEDED** και **MARGINALS\_NEEDED**. Η πρώτη καθορίζει των αριθμό των πλήρως αξιόπιστων ψηφιακών υπογραφών που απαιτούνται για είναι ένα πιστοποιητικό πλήρως αξιόπιστο ενώ η δεύτερη των αριθμό των οριακά αξιόπιστων υπογραφών που απαιτούνται για να είναι ένα πιστοποιητικό πλήρως αξιόπιστο. Στην περίπτωση που σε ένα πιστοποιητικό κανένα από τα δύο όρια δεν ικανοποιείται αλλά το πιστοποιητικό έχει τουλάχιστον μία οριακά ή πλήρη αξιόπιστη υπογραφή τότε θεωρείται οριακά αξιόπιστο πιστοποιητικό.

Πρέπει να σημειώσουμε ότι δύο πιστοποιητικά που τοποθετούνται στο ίδιο επίπεδο ασφάλειας δεν είναι και το ίδιο αξιόπιστα. Επίσης οι χρήστες που υπογράφουν πιστοποιητικά και τοποθετούνται στο ίδιο επίπεδο αξιοπιστίας δεν είναι το ίδιο αξιόπιστοι. Ένας μηχανισμός που θα μπορούσε να εξασφαλίσει έναν ποιο αντικειμενικό διαχωρισμό μεταξύ των πιστοποιητικών θα αντιστοιχούσε μονάδες αξιοπιστίας σε κάθε χρήστη που υπογράφει πιστοποιητικά και η αξιοπιστία του πιστοποιητικού θα ήταν το άθροισμα των μονάδων αξιοπιστίας των χρηστών που το υπέγραψαν.

Το PGP ακολουθεί ένα κατανεμημένο μοντέλο εμπιστοσύνης. Το γεγονός αυτό απαλλάσσει το PGP από τα προβλήματα των συστημάτων αυθεντικοποίησης που στηρίζονται στην υποδομή δημοσίου κλειδιού (ένα σημείο συμφόρησης, ένα σημείο αποτυχίας). Η προσέγγιση του PGP παρά τα όποια πλεονεκτήματα, δεν μπορεί να δώσει μία αξιόπιστη λύση στο πρόβλημα της αυθεντικοποίησης σε συστήματα με ισχυρές απαιτήσεις ασφάλειας. Τα επίπεδα εμπιστοσύνης και αξιοπιστίας που ορίζει δεν είναι αυστηρά ορισμένα ενώ ο κίνδυνος από επιθέσεις συνομωσίας είναι υπαρκτός. Το PGP αποτελεί μία καλή λύση για συστήματα με ποιο χαλαρές απαιτήσεις ασφάλειας π.χ. ηλεκτρονικό ταχυδρομείο.

## Συμπεράσματα

Τα πρωτόκολλα αυθεντικοποίησης που παρουσιάσαμε σ' αυτή την ενότητα σκοπό έχουν την αμοιβαία αυθεντικοποίηση μεταξύ δύο οντοτήτων, το γεγονός αυτό δεν αποτελεί περιοριστικό παράγοντα για να χρησιμοποιηθούν τα παραπάνω πρωτόκολλα για την αμοιβαία αυθεντικοποίηση των μελών ενός group μίας και όπως θα δούμε στην συνέχεια η αυθεντικοποίηση των μελών του group γίνεται είτε από μία οντότητα (συνήθως τον δημιουργό του group) που είναι επιφορτισμένη με αυτή την λειτουργία, είτε όλα τα μέλη του group αυθεντικοποιούνται αμοιβαία ανά δύο.

Η επιλογή ενός συστήματος αυθεντικοποίησης στην περίπτωση των group communication συστημάτων δεν είναι ιδιαίτερα απλή. Πριν επιλέξουμε ένα σύστημα αυθεντικοποίησης πρέπει να γνωρίζουμε τον σκοπό του group communication συστήματος, τον τύπο των εφαρμογών που θα υποστηρίζει, τον αριθμό των χρηστών που θα εξυπηρετεί και την διασπορά των χρηστών στο διαδίκτυο.

Για συστήματα που σκοπό έχουν την διασύνδεση των υπαλλήλων ή των servers μίας επιχείρησης ή ενός οργανισμού τα ID-based σχήματα φαίνεται να πλεονεκτούν μίας και δεν απαιτούν την ύπαρξη μίας on – line έμπιστης τρίτης οντότητας, ενώ η δημιουργία των διαπιστευτηρίων μπορεί να αποτελεί μία δραστηριότητα του τμήματος μηχανοργάνωσης της εταιρίας. Τον ρόλο της έμπιστης τρίτης οντότητας που θα κατέχει την μυστική πληροφορία για την δημιουργία των διαπιστευτηρίων μπορεί να παίξει η διοίκηση της εταιρίας ή του οργανισμού.

Για συστήματα που η διασπορά των χρηστών τους στο διαδίκτυο και η γεωγραφική απόσταση που τα χωρίζει είναι μεγάλη π.χ. οι υπάλληλοι των υπουργείων της Ευρωπαϊκής Ένωσης, η χρήση αυθεντικοποίησης που στηρίζεται

στην ασύμμετρη κρυπτογραφία και την υποδομή δημοσίου κλειδιού φαίνεται να πλεονεκτεί.

Τα πρωτόκολλα που στηρίζονται στην συμμετρική κρυπτογραφία συγκεντρώνουν τα ίδια μειονεκτήματα με αυτά της ασύμμετρης που βασίζεται στην υποδομή δημοσίου κλειδιού (ένα σημείο συμφόρησης, ένα σημείο αποτυχίας) ενώ επιπλέον απαιτούν την ανταλλαγή μυστικής πληροφορίας.

Το σύστημα PGP στην σημερινή του μορφή δεν είναι ικανό να δώσει αξιόπιστη λύση στο πρόβλημα της αυθεντικοποίησης σε εφαρμογές με μεγάλες απαιτήσεις ασφάλειας. Ήα μπορούσε όμως να αποτελέσει ικανοποιητική λύση σε συστήματα με ποιο χαλαρές απαιτήσεις ασφάλειας π.χ. chat rooms.

## Έλεγχος πρόσβασης σε μία ομάδα

Ο μηχανισμός ελέγχου πρόσβασης υποψήφιων μελών σε ένα σύστημα ασφαλούς επικοινωνίας ομάδας αποτελεί τον ακρογωνιαίο λίθο για την ασφάλεια ολόκληρου του συστήματος. Οι μηχανισμοί αυθεντικοποίησης που περιγράψαμε παραπάνω μπορούν να αξιοποιηθούν μόνο εφ' όσον υπάρχει ο κατάλληλος μηχανισμός που θα εξετάζει τα διαπιστευτήρια των υποψηφίων μελών και θα αποφασίζει αν πρέπει να εισέλθουν ή όχι στο group. Επιπλέον η εγκαθίδρυση ενός μυστικού κλειδιού συνόδου μεταξύ των μελών του group υποβαθμίζεται αν δεν μπορούμε να καθορίσουμε με ακρίβεια τα μέλη που θα διαμοιράζονται το μυστικό κλειδί.

Παρ' όλη την σημασία του ελέγχου πρόσβασης για την ασφάλεια ενός group communication συστήματος η επιστημονική κοινότητα δεν έχει ασχοληθεί εκτενώς με αυτό το θέμα αλλά έχει επικεντρωθεί κυρίως σε έρευνα γύρω από μηχανισμούς αυθεντικοποίησης και διαχείρισης του μυστικού κλειδιού συνόδου του group.

## Αρχιτεκτονικές ελέγχου πρόσβασης

Τα περισσότερα group communication συστήματα όπως θα δούμε παρακάτω περιορίζουν τον έλεγχο πρόσβασης σε μία στατική λίστα πρόσβασης (Access Control List –ACL). Η ACL μπορεί να περιέχει τα ονόματα, τα ID ή τα δημόσια κλειδιά των μελών που θα συμμετέχουν σε ένα group. Η διαχείριση μίας ACL μπορεί να γίνεται από μία έμπιστη τρίτη οντότητα ή κάποιο μέλος του group. Το μέλος αυτό είναι συνήθως ο ιδρυτής του group. Είναι φανερό ότι η χρήση ACLs παραβιάζει την

κατανεμημένη φύση των group communication συστημάτων αφού την συνολική ευθύνη για το τον έλεγχο πρόσβασης αναλαμβάνει μόνο μία οντότητα.

Το Antigone είναι το μόνο σύστημα ασφαλούς επικοινωνίας ομάδας από τα ευρέως διαδεδομένα που υιοθετεί ένα ολοκληρωμένο μηχανισμό ελέγχου πρόσβασης. Στο Antigone ο έλεγχος πρόσβασης γίνεται μέσο μίας έμπιστης τρίτης οντότητας, όμως όλα τα μέλη έχουν γνώση των μελών που είναι κάθε στιγμή συνδεδεμένα (current membership), αν κάποιο μέλος δεν επιθυμεί να είναι στο ίδιο group με κάποιο άλλο μπορεί να ζητήσει την αποβολή του δεύτερου από το group και την αναθεώρηση του μυστικού κλειδιού συνόδου.

Μία άλλη προσέγγιση στον έλεγχο πρόσβασης υιοθετεί το σύστημα Secure Group Layer (SGL). Στο σύστημα αυτό ένας εξυπηρετητής συγκεντρώνει τις πολιτικές πρόσβασης των group που εξυπηρετεί και με βάση αυτές εκδίδει πιστοποιητικά πρόσβασης στα μέλη που επιθυμούν και βάση της πολιτικής πρόσβασης έχουν δικαίωμα τα εισέλθουν στο group. Το πιστοποιητικό περιέχει το όνομα του group, το όνομα του μέλους και το δημόσιο κλειδί του και υπογράφεται ψηφιακά από τον εξυπηρετητή. Ο εξυπηρετητής λειτουργεί παρόμοια με μία αρχή πιστοποίησης όπως την περιγράψαμε στην υποδομή δημοσίου κλειδιού, έτσι έκτος από την έκδοση των πιστοποιητικών αναλαμβάνει και την διαχείριση αυτών καθ' όλο τον κύκλο ζωής τους.

Εκτός από τους παραπάνω μηχανισμούς ελέγχου πρόσβασης που αποτελούν μέρος των group communication συστημάτων και στηρίζονται στις ιδιότητες των υποκείμενων συστημάτων, θα παρουσιάσουμε δύο αρχιτεκτονικές που υιοθετούν την λογική της ψηφοφορίας μεταξύ των μελών προκειμένου να αποφασισθεί η είσοδος ενός νέου μέλους στο group. Η μία από τις αρχιτεκτονικές κάνει χρήση πιστοποιητικών δημοσίου κλειδιού, ενώ η άλλη ακολουθεί μία ID-based προσέγγιση.

## ID-based αρχιτεκτονική

Η ID-based αρχιτεκτονική [73] στηρίζεται στην threshold έκδοση του BLS σχήματος ψηφιακών υπογραφών [68, 69]. Στην συνέχεια θα παραθέσουμε το σύνολο των συμβόλων που θα χρησιμοποιηθούν στην περιγραφή του σχήματος.

$M_i$	το μέλος $i$
$t$	ο αριθμός των μελών που πρέπει να δώσουν την συγκατάθεση τους για την είσοδο ενός νέου μέλους στο group (threshold)
$n$	ο αριθμός των μελών του group
$A$	ο αριθμός που χρησιμοποιείτε σαν γεννήτορας μίας ομάδας $G_1$
$\hat{e}$	αντιστοιχία π.χ. $\hat{e} : G_1 \times G_1 \rightarrow G_2$
$SK_i$	το μυστικό κλειδί του μέλους $M_i$
$S_i(m)$	υπογραφή του μηνύματος $m$
$SL_i$	η λίστα των υπογραφόντων για το μέλος $M_i$
$ss_i$	το μυστικό κομμάτι (secret share) του μέλους $M_i$
$pss_j$	το μερικό κομμάτι (partial share) του $M_i$ από τον $M_j$
$id_i$	ο προσδιοριστή (ID) του μέλους $M_i$
$t_r$	το όριο για την ανάκληση ενός μέλους από το group (revocation threshold)
$G_1, G_2$	κυκλικά GDH groups της τάξης $q$
$B$	το δημόσιο κλειδί του group
$T_i$	το token του μέλους $M_i$
$PK_i$	το δημόσιο κλειδί του μέλους $M_i$
$MRL$	η λίστα ανακληθέντων μελών
$H$	συνάρτηση σύνοψης όπως η SHA-1 και η MD5
$H_1$	συνάρτηση σύνοψης της μορφής $\{0,1\}^* \rightarrow G_1^*$
$H_2$	συνάρτηση σύνοψης της μορφής $\{0,1\}^* \times G_1 \rightarrow Z_q^*$

Η αρχικοποίηση της διαδικασίας ελέγχου πρόσβασης μπορεί να γίνει είτε με την ανάμειξη μίας έμπιστης οντότητας (trusted dealer = TD) ή από  $t - 1$  (ή και περισσότερα) ιδρυτικά μέλη του group. Ανεξάρτητα του τρόπου με τον οποίο θα γίνει η αρχικοποίηση της διαδικασίας το πρώτο βήμα είναι η αρχικοποίηση και η δημιουργία από τον dealer των παραμέτρων  $(p, F_p, a, b, A, q)$  της ελλειπτικής καμπύλης. Η καμπύλη δίνεται από την ισότητα  $y^2 = x^3 + ax + b$ . Το  $G_1$  είναι μία ομάδα της τάξης  $q$  με γεννήτορα τον  $A$ . Το  $G_2$  είναι μία υποομάδα του  $F_{p^2}^*$  ενώ το  $\hat{e} : G_1 \times G_2 \rightarrow G_2$  είναι μία δημόσια διγραμμική αντιστοίχιση. Επίσης  $H_1 : \{0,1\}^* \rightarrow G_1^*$  είναι μία συνάρτηση σύνοψης που αντιστοιχεί δυαδικές συμβολοσειρές σε μη

μηδενικά στοιχεία στο  $G_1$ . Όλες οι παραπάνω πληροφορίες είναι δημόσια γνωστές σε όλα τα τρέχοντα και υποψήφια μέλη.

### **Αρχικοποίηση από τον dealer**

Ο TD διαλέγει τυχαία ένα πολυώνυμο  $f(z) = f_0 + f_1z + \dots + f_{t-1}z^{t-1}$  βαθμού  $t-1$  έτσι ώστε το μυστικό του group να δίνεται από την σχέση  $f(0) = f_0 = x$ . Προκειμένου να παρέχεται επαληθευσιμότητα των μυστικών κομματιών (verifiable secret sharing – VSS), ο TD υπολογίζει και δημοσιεύει τις ποσότητες  $W_i = f_i A$  ( $i = 0, \dots, t-1$ ) καθώς και την ποσότητα  $W_0 = xA = B$ . Τα  $W_i$  είναι κατά κάποιον τρόπο τα δημόσια κλειδιά των μελών που συμμετέχουν στην διαδικασία ελέγχου πρόσβασης ενώ το  $W_0 = B$  είναι το δημόσιο κλειδί του group. Στην συνέχεια για κάθε μέλος  $M_i$ , ο TD υπολογίζει το μυστικό κομμάτι (secret share) και το identity based token  $T_i$ . Το  $T_i$  είναι ουσιαστικά το “διαβατήριο” ενός μέλους για την είσοδο στο group, με την κατοχή του ο  $M_i$  αποδεικνύει ότι είναι μέλος του group. Ο υπολογισμός του  $ss_i$  και του  $T_i$  γίνεται με χρήση των έξης συναρτήσεων :  $ss_i = f(id_i)$  και  $T_i = xH_1(id_i||exp)$ . Όπου  $exp$  είναι η ημερομηνία λήξης του token. Η ημερομηνία αυτή είναι κοινή για όλα τα μέλη.

### **Αρχικοποίηση από τα ιδρυτικά μέλη του group**

Για να γίνει αρχικοποίηση από τα μέλη του group απαιτείται η παρουσία  $t - 1$  ή περισσοτέρων μελών. Κάθε μέλος  $M_i$  επιλέγει ένα πολυώνυμο  $f_i(z)$  βαθμού  $t - 1$ , τα πολυώνυμα των  $i$  μελών είναι ανεξάρτητα μεταξύ τους. Στην συνέχεια με χρήση του DKG [70] πρωτοκόλλου κάθε μέλος  $M_i$  υπολογίζει το προσωπικό του secret share έτσι ώστε  $ss_i = \sum_{j=1}^l f_j(id_i)(mod q)$  ( $l \geq 2t - 1$ ). Με τον υπολογισμό του share κάθε  $M_i$  μπορεί να ανακτήσει το μυστικό του group με την χρήση Lagrange. Η παραπάνω διαδικασία υποστηρίζει την αρχή του VVS. Στην συνέχεια προκειμένου το κάθε μέλος να αποκτήσει ένα membership token κάθε σύνολο από τα  $t$  ιδρυτικά μέλη πρέπει να συνεργαστεί. Για παράδειγμα τα μέλη του group  $M_2, M_3, \dots, M_{t+1}$  πρέπει να συνεργάστουν για την δημιουργία του membership token του  $M_1$ . Το token  $T_1$  υπολογίζεται ως έξης  $T_1 = \sum_{j=2}^{t+1} (ss_j)^l (0)) H_1(id_1 || exp) [= xH_1(id_1 || exp)]$ .

## Διαδικασία εισόδου νέου μέλους

Έστω  $n \geq t$  ο αριθμός των τρέχοντων μελών του group. Προκειμένου ένα μέλος να αποκτήσει δικαιώμα πρόσβασης στο group, πρέπει να συγκεντρώσει τουλάχιστον  $t$  ψήφους από τα τρέχοντα μέλη του group. Στο σχήμα που ακολουθεί απεικονίζονται τα μηνύματα που ανταλλάσσονται κατά την διαδικασία εισόδου νέου μέλους.

1: $M_{new}$	$REQ = id_{new}, m, S_{new}(id_{new}, m)$	$M_i$
2: $M_{new}$	$(id_i, S_i(id_i, H(REQ)))$	$M_i$
3: $M_{new}$	$SL_{new}, S_{new}(SL_{new})$	$M_j$
4: $M_{new}$	$T_j(new), [pss_j(new)], S_j(T_j(new), SL_{new}, [pss_j(new)])$	$M_j$

Στην διαδικασία αυτή στόχος του  $M_{new}$  είναι η απόκτηση ενός token  $T_{new}$  με το οποίο θα εξασφαλίσει την είσοδο του στο group. Τα βήματα της διαδικασίας πρόσβασης είναι τα ακόλουθα:

- Ο  $M_{new}$  στέλνει ένα υπογεγραμμένο μήνυμα – αίτηση πρόσβασης  $m$  μαζί με το αναγνωριστικό του  $id_{new}$  σε τουλάχιστον  $t$  μέλη από τα τρέχοντα μέλη του group.
- Τα μέλη του group που επιθυμούν να συμμετάσχουν στην διαδικασία πρόσβασης απαντούν με ένα μήνυμα που περιέχει το  $id$  τους και την σύνοψη του μηνύματος αίτησης που έλαβαν και το υπογράφουν ψηφιακά.
- Ο  $M_{new}$  επιλέγει τα μέλη (sponsors)  $M_j$ -s από τα μέλη που απάντησαν στο μήνυμα – αίτηση του και καταρτίζει την λίστα υπογραφόντων  $SL_{new}$  η οποία περιέχει τα  $id$ -s των επιλεχθέντων μελών. Η λίστα υπογράφεται ψηφιακά από τον  $M_{new}$  και αποστέλλεται στους  $M_j$ -s.
- Κάθε μέλος από τα επιλεχθέντα  $M_j$  στέλνει στον  $M_{new}$  ένα μερικό (partial) membership token  $T_j(new)$  και ένα μερικό κομμάτι του μυστικού του group  $pss_j(new)$  έτσι ώστε :

$$T_j(new) = (ss_j \cdot l_j(0))H_1(id_{new} || exp)$$

$$pss_j(new) = ss_j \cdot l_j(id_{new}) + r_j \pmod{q}$$

$$\text{όπου } l_j(x) = \prod_{i=1}^t \frac{x - id_i}{id_j - id_i}$$

Ο  $M_{new}$  προμηθεύεται επίσης την λίστα ανακληθέντων μελών MRL. Οι συντελεστές Lagrange  $l_j(id_{new})$  είναι δημόσια γνωστοί και έτσι ο  $M_{new}$

μπορεί να υπολογίσει το secret share  $ss_j$ -s των μελών από τα partial secret shares τους  $pss_j$ -s. Αυτό μπορεί να αποφευχθεί αν χρησιμοποιηθεί η shuffling τεχνική. Στην τεχνική αυτή σε κάθε share προστίθεται μία τιμή  $r_j$ -s που κάνει αδύνατο τον share μόνο με την χρήση των συντελεστών Lagrange, όταν όμως όλα τα shares προστεθούν το σύνολο των  $r_j$ -s είναι μηδέν με αποτέλεσμα να μην επηρεάζεται το αποτέλεσμα της πρόσθεσης (το μυστικό του group στην παρούσα περίπτωση)

5. Τελικά ο  $M_{new}$  υπολογίζει το  $ss_{new}$  και το membership token  $T_{new}$  αθροίζοντας  $pss_j(new)$ -s και  $T_j(new)$ -s αντίστοιχα που έλαβε στο βήμα 4

### *Μελέτη ασφάλειας*

Η καταλληλότητα του πρωτοκόλλου για τον έλεγχο πρόσβασης σε ένα group μπορεί να θεωρηθεί συνάρτηση τριών παραμέτρων.

1. *An μπορεί να δώσει μία ολοκληρωμένη λύση στο πρόβλημα του ελέγχου πρόσβασης.* Η διαδικασία που περιγράψαμε παραπάνω δίνει μία ολοκληρωμένη λύση στο πρόβλημα ελέγχου πρόσβασης μίας και μετά το τέλος της διαδικασίας το νέο μέλος λαμβάνει ένα membership token η ορθότητα του οποίου μπορεί να ελεγχθεί με την επαλήθευση της ισότητας  $\hat{e}(B, H_1(id_{new}||exp)) = \hat{e}(A, T_{new})$ . Με την χρήση του  $T_{new}$  ο  $M_{new}$  μπορεί να αποδείξει την ιδιότητα μέλους. Επίσης λαμβάνει ένα secret share η ορθότητα του οποίου μπορεί να ελεγχθεί με την επαλήθευση της ισότητας  $ss_{new}A = \sum_{i=0}^{t-1} id_{new}^i W^i$ . Με την χρήση του  $ss_{new}$  ο  $M_{new}$  μπορεί να λάβει μέρος σε μελλοντικές αποφάσεις πρόσβασης.
2. *Anιχνευσιμότητα.* Στην περίπτωση που η επαλήθευση της ορθότητας των  $T_{new}$  ή  $ss_{new}$  αποτύχει ο  $M_{new}$  πρέπει να εντοπίσει το εντοπίσει το μέλος που έστειλε λανθασμένο partial token ή partial secret share αντίστοιχα. Για την επαλήθευση κάθε partial secret share ο  $M_{new}$  μπορεί να χρησιμοποιήσει την διαδικασία VSS. Ο έλεγχος της ορθότητας κάθε partial token μπορεί να γίνει μέσω της επαλήθευσης της παρακάτω ισότητας

$$\hat{e}(T_j(new), A) = \hat{e}(H_1(id_{new}||exp), l_j(0) \sum_{i=0}^{t-1} id_{new}^i W^i)$$

3. Αντίσταση σε απειλές πλαστοπροσωπίας. Προκειμένου να βεβαιωθεί ο  $M_{new}$  ότι επικοινωνεί μόνο με γνήσια μέλη του group, μπορεί να επαληθεύσει τα partial credentials με διαδικασία ίδια με αυτή της ανιχνευσιμότητας.

Όπως παρατηρούμαι η παραπάνω διαδικασία ικανοποιεί τις τρις παραμέτρους και αποτελεί μία αξιόπιστη λύση για το πρόβλημα του ελέγχου πρόσβασης.

### Απόδειξη ιδιότητας μέλους

Ένα μέλος μπορεί να αποδείξει ότι είναι μέλος ενός group, μέσω δύο σχημάτων. Το ένα σχήμα χρησιμοποιείται προκειμένου να αποδείξει ένα μέλος την ιδιότητα μέλους που κατέχει σε ένα άλλο μέλος του group (internal membership proof – IMP). Ενώ το άλλο για να αποδείξει την ιδιότητα μέλους σε μέλη που δεν ανήκουν στο group (external membership proof – EMP). Το IMP σχήμα είναι ένα pairing – based secret handshake σχήμα [71], ενώ το EMP είναι ένα identity based σχήμα ψηφιακών υπογραφών σε Gup Diffie Hellman groups – GDH groups [72].

### Ανάκληση membership token

Η εισαγωγή του πεδίου exp στο membership token αποτελεί ένα είδος μηχανισμό ανάκλησης δικαιώματος πρόσβασης σε ένα group. Τα μέλη του group κατά την αναθεώρηση του δικαιώματος πρόσβασης μπορούν να μην αναθεωρήσουν το δικαίωμα πρόσβασης κάποιων από τα μέλη του group. Επιπλέον τα εναπομείναντα μέλη πρέπει να ανανεώσουν τα secret share με χρήση της proactive secret sharing τεχνικής [74] (η proactive secret sharing τεχνική επιτρέπει σε τ μέλη να αλλάξουν τα secret share τους χωρίς να απαιτείται αλλαγή του μυστικού που συνθέτουν τα shares) τους στην περίπτωση που τα μέλη που αποβλήθηκαν από το group κατείχαν κάποιο secret share. Η παραπάνω πολιτική ανάκλησης της ιδιότητας μέλους, αν και δίνει μία λύση στο πρόβλημα είναι ιδιαίτερα αναποτελεσματική, για το λόγο αυτό το ID – based σχήμα ελέγχου πρόσβασης υλοποιεί ένα μηχανισμό ανάκλησης που στηρίζεται σε λίστες ανάκλησης ιδιότητας μέλους (membership revocation lists – MRLs) που είναι αντίστοιχες των certificate revocation lists που χρησιμοποιούνται για την ανάκληση πιστοποιητικών δημοσίου κλειδιού.

Τα θέματα που θα αναλυθούν στην παρουσίαση του μηχανισμού ανάκλησης είναι τα ακόλουθα:

- Επικύρωση ιδιότητας μέλους (Membership Validation)

## ➤ Διαδικασία ανάκλησης (Revocation Process)

### *Membership Validation*

Όταν ένας χρήστης V λαμβάνει ένα υπογεγραμμένο μήνυμα από κάποιον άλλο χρήστη έστω  $M_u$  ο οποίος ενστερνίζεται την ιδιότητα του μέλους ενός group με δημόσιο κλειδί B, τότε ο V πρέπει αρχικά να ελέγξει αν ο  $M_u$  είναι πραγματικά ένα τρέχον μέλος του group (membership validation) και στην συνέχεια να επαληθεύσει την ψηφιακή υπογραφή του μηνύματος χρησιμοποιώντας το EMP σχήμα επαλήθευσης ψηφιακών υπογραφών. Η διαδικασία του membership validation διαφέρει ανάλογα με το αν ο V είναι μέλος του group ή όχι.

Στην περίπτωση που ο χρήστης V είναι μέλος του group (Internal Membership Validation – IMP), ελέγχει στην MRL λίστα του την κατάσταση του  $M_u$ . Στην περίπτωση που η MRL δεν περιέχει εγγραφή που να αντιστοιχεί στο  $id_u$ , ο V θεωρεί των  $M_u$  σαν ένα έγκυρο μέλος του group.

Αν ο V δεν είναι μέλος του group (External Membership Validation – EMV) τότε η membership validation διαδικασία του  $M_u$  από τον V περιέχει τα παρακάτω βήματα:

1. Ο V στέλνει στο group μια αίτηση ελέγχου της ιδιότητας μέλους (membership validation request) του  $M_u$ .
2. Τα μέλη του group που κατέχουν ένα share και ενδιαφέρονται να απαντήσουν στέλνουν στον V τα ID τους.
3. Ο V περιμένει μέχρι την συγκέντρωση των IDs. Όταν τα συγκεντρώσει καταρτίζει την λίστα υπογραφόντων  $SL_u$  με βάση τα IDs που έλαβε και την αποστέλλει στους κατόχους των IDs που περιέχει η λίστα.
4. Με την λήψη της λίστας τα μέλη ελέγχουν την κατάσταση του  $M_u$  βάση των προσωπικών τους MRLs και στέλνουν στον V την απάντηση τους ψηφιακά υπογεγραμμένη.
5. Ο V επαληθεύει την υπογραφή των μηνυμάτων απάντησης του βήματος 4.

### *Revocation Process*

Η διαδικασία ανάκλησης ενός μέλους μπορεί να αρχικοποιηθεί από κάποιο τρέχων μέλος  $M_a$  του group, τα βήματα της διαδικασίας είναι τα παρακάτω:

1. Ο  $M_a$  στέλνει σε όλα τα μέλη του group (broadcast) ένα μήνυμα – αίτηση ανάκλησης της ιδιότητας μέλους κάποιου μέλους  $M_r$ , χρησιμοποιώντας το EMP σχήμα υπογραφών.
2. Όλα τα υπόλοιπα μέλη του group  $M_j$ -s ( $j \neq r$ ) εκτελούν την IMP διαδικασία για τον  $M_a$ . Στην συνέχεια ανανεώνουν τις MRL-s τους προσθέτοντας τον  $M_r$  στην λίστα ανακληθέντων και προσθέτουν τον  $M_a$  στους ανακαλούντες για τον  $M_r$  ενώ θέτουν την κατάσταση (status) του  $M_r$  σε “υπό-αναθεώρηση” (under-review). Όταν ο αριθμός των ανακαλούντων του  $M_r$  φτάσει το όριο ανάκλησης (revocation threshold)  $t_r$ , τότε το status του  $M_r$  ανανεώνεται σε “ανακληθέν” (revoked).
3. Αν ο  $M_r$  είναι το  $t$  μέλος με  $t \leq t-1$  που ανακαλείται (όπου  $t$  ο αριθμός ανακληθέντων μελών που προκαλεί ανανέωση των secret shares και καθορίζεται από την πολιτική του group), τότε οι  $t_r$  ανακαλούντες συνεργάζονται και ανανεώνουν τα secret shares με την proactive μέθοδο. Στην περίπτωση που υπάρχουν λιγότεροι από  $t$  από τους  $t_r$  ανακαλούντες που κατέχουν secret shares τότε τα ιδρυτικά μέλη του group πρέπει να ενεργοποιήσουν την διαδικασία ανανέωσης των shares.
4. Όλα τα μέλη του group  $M_i$ -s ( $i \neq r$ ) που έχουν δικαίωμα ψήφου επικοινωνούν με τους ανακαλούντες ή με τα ιδρυτικά μέλη και ανανεώνουν τα share τους.

## Αρχιτεκτονική που κάνει χρήση πιστοποιητικών δημοσίου κλειδιού

Μία αξιόλογη δουλεία γύρο από τον έλεγχο πρόσβασης σε ένα group communication σύστημα έγινε από τους Yordan Kim, Daniele Mazzocchi και Gene Tsudik [29].

Βασικές έννοιες στην προσέγγιση αυτή είναι:

- Ο κανονισμός του group (Group Charter – GC)
- Η αρχή του group (Group Authority – GAUTH)

Το **group charter** είναι ένα ηλεκτρονικό έγγραφο στο οποίο κωδικοποιούνται οι κανόνες πρόσβασης σε ένα group. Σκοπός του εγγράφου είναι να πληροφορήσει τα μελλοντικά μέλη σχετικά με την διαδικασία ελέγχου πρόσβασης στο group. Το group

charter υπογράφεται ψηφιακά από την αρχή έκδοσης. Η αρχή έκδοσης μπορεί να είναι μία αρχή πιστοποίησης ή ο ιδρυτής του group.

Ένα group charter πρέπει να περιέχει

1. Το όνομα του group (Group Name – GN)
2. Το όνομα της αρχής (Group Authority Name – GAUTH Name)
3. Τον τύπο της πολιτικής πρόσβασης (Admission Policy Type – APT)
4. Το όνομα της αρχής έκδοσης και την ψηφιακή της υπογραφή

Οι πολιτικές πρόσβασης σε ένα group καθορίζουν τον μηχανισμό έλεγχου πρόσβασης που κάθε group υιοθετεί. Οι τύποι πολιτικών που μπορεί να περιέχονται στο group charter είναι οι ακόλουθοι

- APT\_ACL : Ο τύπος αυτός δηλώνει ότι η έλεγχος πρόσβασης γίνεται μέσω μίας ACL
- APT\_GAUTH: Η πρόσβαση ενός μέλους στο group είναι αποκλειστική ευθύνη του GAUTH
- APT\_GROUP: Η απόφαση για την πρόσβαση ενός νέου μέλους στο group λαμβάνεται από τα μέλη του group. Εδώ μπορούμε να διακρίνουμε τρις υποκατηγορίες
  - APT\_Group.Static: Η απόφαση για την είσοδο ενός νέου μέλους στο group απαιτεί την συναίνεση κάποιου σταθερού αριθμού μελών (κάτω όριο), αν ο αριθμός των μελών του group πέσει κάτω από το όριο αυτό απαιτείται ειδική πολιτική.
  - APT\_Group.Dynamic: Η απόφαση για την είσοδο νέου μέλους στο group απαιτεί την συναίνεση ενός ποσοστού των μελών του group. Σε αυτήν την περίπτωση απαιτείτε η ύπαρξη πολιτικής που θα ενεργοποιείται όταν δημιουργείται το group και δεν εμπεριέχει ακόμα κάποιο μέλος.
  - APT\_Group.Hybrid: Είναι μία μίξη των πολιτικών APT\_Group.Static και APT\_Group.Dynamic.

Αναφέραμε το **GAUTH** σαν την αρχή ενός group. Το GAUTH δεν είναι κάποια συγκεκριμένη οντότητα, τον ρόλο του μπορεί να διατελέσει κάποια έμπιστη τρίτη οντότητα όπως μία αρχή πιστοποίησης ή το μέλος που ξεκινά την δημιουργία ενός group ή ακόμα και το ίδιο το group. Η σημαντικότερη εργασία του GAUTH είναι η

δημιουργία των πιστοποιητικών των νέων μελών (Group Membership Certificate – GMC). Τα GMCs είναι ουσιαστικά τα “διαβατήρια” των νέων μελών για την είσοδό τους στο group. Η περίοδος ισχύος ενός τέτοιου τύπου πιστοποιητικού είναι περιορισμένη και μπορεί να χρησιμοποιηθεί από κάποιο νέο μέλος για την είσοδο του σε ένα και μόνο group. Ένα GMC πρέπει να περιέχει:

- Το όνομα του group
- Ημερομηνία έκδοσης
- Διάρκεια ισχύος
- Το δημόσιο κλειδί που αντιστοιχεί στο χρήστη ή αναφορά στο πιστοποιητικό δημοσίου κλειδιού του χρήστη (PKC)



### **Διαδικασία ελέγχου πρόσβασης**

Η διαδικασία ελέγχου πρόσβασης μπορεί να χωριστεί σε τρία στάδια

- Το **πρώτο στάδιο** περιλαμβάνει την δημιουργία του group charter. Μπορούμε να θεωρήσουμε για λόγους απλότητας ότι το group charter είναι πάντα υπογεγραμμένο από μία off – line αρχή. Το group charter δημιουργείται κατά την φάση δημιουργίας του group.
- Το **δεύτερο στάδιο** περιλαμβάνει την αλληλεπίδραση του υποψηφίου μέλους με τα τρέχοντα μέλη του group. Πριν προσπαθήσει να εισέλθει στο group το νέο μέλος πρέπει να γνωρίζει το όνομα του group. Το επόμενο βήμα είναι να αποκτήσει το group charter. Αυτό μπορεί να γίνει είτε λαμβάνοντας το group charter από το ίδιο το group είτε από κάποιο on – line εξυπηρετητή.
- Το **τρίτο βήμα** περιλαμβάνει την επικοινωνία μεταξύ υποψηφίου μέλους και GAUTH. Ο GAUTH ελέγχει αν το μέλος ικανοποιεί όλες τις προδιαγραφές πρόσβασης. Αν τις ικανοποιεί δημιουργεί το GMC του νέου μέλους.

Στα τρία παραπάνω βήματα δεν εξετάσαμε εσκεμμένα την ίδια την διαδικασία ελέγχου πρόσβασης αυτό έγινε γιατί η διαδικασία μπορεί να ποικίλει ανάλογα με τα χαρακτηρίστικα και τις απαιτήσεις ασφάλειας του κάθε group και για αυτό θα αναλυθεί διεξοδικά στην συνέχεια .

Το πλαίσιο ελέγχου πρόσβασης που περιγράφουμε υποστηρίζει πολιτικές τόσο για στατικά όσο και για δυναμικά groups. Στα στατικά groups όλα τα πιθανά μέλη είναι από πριν γνωστά, έτσι μπορούμε να καταρτίσουμε λίστες πρόσβασης με τα ονόματα των μελών που έχουν δικαίωμα πρόσβασης στο group ή να δημιουργήσουμε



κανόνες πρόσβασης π.χ. τα μέλη ενός group G πρέπει να κατέχουν ένα PKC από κάποια συγκεκριμένη CA. Στα δυναμικά group η γνώση από πριν των μελών του είναι πρακτικά αδύνατη και πρέπει να υιοθετηθούν άλλες τεχνικές ελέγχου πρόσβασης.

Οι αρχιτεκτονικές ελέγχου πρόσβασης που θα παρουσιάσουμε στην συνέχεια διαφοροποιούνται ως προς τον ρόλο του GAUTH.

Διακρίνουμε τις ακόλουθες αρχιτεκτονικές:

- Είσοδος μέσω δημόσιας ACL
- Είσοδος μέσω του GAUTH
- Είσοδος μέσω των μελών του group
- Είσοδος χωρίς χρήση GAUTH

### ***Eίσοδος μέσω μίας δημόσιας ACL***

Η αρχιτεκτονική αυτή είναι το απλούστερο σενάριο ελέγχου πρόσβασης όλα τα πιθανά μέλη απαριθμούνται σε μία λίστα. Τα μέλη είναι γνωστά κατά την φάση δημιουργίας του group. Στην αρχιτεκτονική αυτή το group charter είναι ουσιαστικά μία υπογεγραμμένη ACL. Η αρχιτεκτονική δεν απαιτεί την ύπαρξη GAUTH μίας και ο έλεγχος πρόσβασης στηρίζεται στην ACL.

Όταν δύο ή περισσότερα μέλη επικοινωνούν μπορούν απλά να υπογράφουν με τα δημόσια κλειδιά τους όλα τα μηνύματα που ανταλλάσσουν. Κάθε μήνυμα που είναι υπογεγραμμένο με ένα δημόσιο κλειδί που περιέχεται στην ACL λαμβάνεται σαν μήνυμα που έχει αποσταλεί από έγκυρο μέλος του group.

Το βασικότερο πρόβλημα της παραπάνω μεθόδου είναι η έλλειψη ευελιξίας λόγο του στατικού membership (σύνολο των μελών ενός group).

### ***Eίσοδος μέσω του GAUTH***

Σε αυτή την αρχιτεκτονική ο έλεγχος πρόσβασης γίνεται αποκλειστικά και μόνο από τον GAUTH. Ο GAUTH αναλαμβάνει την δημιουργία των GMCs των νέων μελών η απόφαση αν ένα νέο μέλος θα εισέλθει στο group είναι αποκλειστικά δική του. Ο GAUTH σε αυτή την αρχιτεκτονική πρέπει να χαίρει εμπιστοσύνης από όλα τα τρέχοντα μέλη του group επιπλέον πρέπει να είναι on – line και να διαθέτει ισχυρούς μηχανισμούς αντιμετώπισης επιθέσεων. Εξαιτίας του γεγονότος ότι η αρχιτεκτονική αυτή απαιτεί την ύπαρξη ενός on – line GAUTH είναι ελκυστική για

σύγχρονα groups όπου το ρόλο του GAUTH μπορεί να παίζει ο ιδρυτής του group ή κάποιο άλλο μέλος που έχει εξουσιοδοτηθεί από το group για αυτή την λειτουργία.

Μία παραλλαγή της αρχιτεκτονικής μπορεί να περιέχει πολλούς GAUTHs. Τα ονόματα των GAUTHs θα αναφέρονται στο group charter. Η χρήση ενός off – line GAUTH θα επιβραδύνει την διαδικασία ελέγχου πρόσβασης. Μία τέτοια επιλογή πρέπει να υιοθετηθεί μόνο στην περίπτωση ασύγχρονων group communication συστημάτων.

### **Είσοδος μέσω των μελών του group**

Όταν ο έλεγχος πρόσβασης γίνεται από τα μέλη του group (APT\_GROUP) όπως ήδη έχουμε αναφέρει η διαδικασία πρόσβασης τροποποιείται ανάλογα με το αν θα χρησιμοποιηθούν στατικά ή δυναμικά όρια. Όταν γίνει χρήση στατικών ορίων πρέπει τουλάχιστον  $t$  (όπου  $t$  το όριο) από τα  $n$  τρέχοντα μέλη του group να εγκρίνουν την είσοδο του υποψήφιου μέλους. Στην περίπτωση που  $n < t$  πρέπει να υπάρχει ιδική πολιτική που θα χειρίζεται την εξαίρεση. Στα δυναμικά όρια πρέπει ένα ποσοστό επί του συνόλου των τρεχόντων μελών να εγκρίνουν την είσοδο του υποψήφιου μέλους. Κατά την φάση δημιουργίας του group τον έλεγχο πρόσβασης θα χειρίζεται ειδική πολιτική.

Ανεξάρτητα όμως από τον τύπο των ορίων που θα χρησιμοποιηθούν τα κοινά βήματα και των δύο πολιτικών είναι τα ακόλουθα.

**Βήμα 1<sup>o</sup>. Αίτηση σύνδεσης (Join Request):** Αρχικά ένα υποψήφιο μέλος έστω  $M_{new}$  στέλνει μία αίτηση σύνδεσης (join\_request). Το μήνυμα της αίτησης είναι ψηφιακά υπογεγραμμένο από τον  $M_{new}$  και περιέχει το πιστοποιητικό του  $M_{new}$  ( $Cert_{new}$ ) και το όνομα του group στο οποίο θέλει να συνδεθεί π.χ. AUEB. Ο τρόπος με τον οποίο διανέμεται το μήνυμα εξαρτάται από την εφαρμογή.

**Βήμα 2<sup>o</sup>. Ψηφοφορία (Voting):** Με την λήψη του μηνύματος που περιέχει την αίτηση σύνδεσης κάθε μέλος που ανήκει στο group, που το μήνυμα απευθύνεται εξάγει το PKC του υποψήφιου μέλους και ελέγχει την ψηφιακή υπογραφή. Αν τα μέλη που συμμετέχουν στην ψηφοφορία αποδέχονται την είσοδο του υποψήφιου μέλους τότε απαντούν στον  $M_{new}$  με ένα υπογεγραμμένο μήνυμα. Ο τύπος της ψηφιακής υπογραφής που θα χρησιμοποιηθεί στην διαδικασία ψηφοφορίας αναγράφεται στο group charter μερικές από τις πιθανές επιλογές είναι:

- threshold signatures
- group signatures
- subgroup multisignatures
- plain individual signatures

Οι τεχνικές αυτές θα αναλυθούν στην συνέχεια.

**Βήμα 3<sup>o</sup>. Αίτηση για GMC (GMC request):** Με την λήψη του απαραίτητου αριθμού υπογραφών από τα μέλη του group (σύμφωνα με το group charter -  $GC^{AUEB}$ ) ο  $M_{new}$  στέλνει στον GAUTH ένα μήνυμα με το οποίο ζητά ένα GMC για την είσοδό του στο group. Το μήνυμα αυτό πρέπει να περιέχει το PKC του  $M_{new}$  και το σύνολο των ψήφων που έχει συλλέξει από τα μέλη του group.

**Βήμα 4<sup>o</sup>. Έκδοση του GMC (GMC issuance):** Με την λήψη της αίτησης για έκδοση πιστοποιητικού ο  $GAUTH^{AUEB}$  ελέγχει αν ο  $M_{new}$  έχει συμπληρώσει των απαιτούμενο αριθμό υπογραφών που αναφέρεται στο group charter και αν οι υπογραφές, είναι έγκυρες υπογραφές των μελών του group. Αν οι παραπάνω έλεγχοι ολοκληρωθούν με επιτυχία τότε ο  $GAUTH$  δημιουργεί ένα νέο  $GMC_{new}^{AUEB}$  για τον  $M_{new}$ .

Οπλισμένος με το GMC ο  $M_{new}$  μπορεί να δράσει σαν αξιόπιστο μέλος στο group. Για να αποδείξει την ταυτότητά του σε κάποιο άλλο μέλος του group αρκεί να υπογράψει ένα μήνυμα και να το αποστείλει στο μέλος που επιθυμεί να τον αυθεντικοποιήσει.

### **Είσοδος χωρίς χρήση GAUTH**

Στην παραπάνω αρχιτεκτονική αν και η απόφαση πρόσβασης ενός νέου μέλους στο group λαμβάνονταν συλλογικά από όλα τα τρέχοντα μέλη του, υπήρχε η απαίτηση της ύπαρξης ενός GAUTH που θα δημιουργούσε τα GMCs. Μία ιδέα προκειμένου να αποφευχθεί η χρήση του GAUTH είναι κάθε μέλος να ψηφίζει αν επιθυμεί την είσοδο του  $M_{new}$ , το τελευταίο μέλος που θα συμμετέχει στην ψηφοφορία θα ελέγχει αν έχει συμπληρωθεί ο αριθμός των ψήφων που απαιτεί το group charter και εφ' όσον κάτι τέτοιο συμβαίνει να το χρησιμοποιείται σαν απόδειξη για την πρόσβαση του  $M_{new}$  στο group. Το σχήμα αυτό μπορεί να είναι μη λειτουργικό για μεγάλα groups, επίσης πρέπει να διαθέτει ένα μηχανισμό που θα ελέγχει αν οι υπογραφές που έχει λάβει ένα υποψήφιο μέλος είναι από έγκυρα μέλη του group.

Σχολιάζοντας τις παραπάνω αρχιτεκτονικές πρέπει να πούμε ότι η πρώτη (χρήση ACLs) είναι μακράν η απλούστερη αλλά τυγχάνει έλλειψης ευελιξίας. Η δεύτερη (χρήση GAUTH) είναι ποιό ευέλικτη αλλά στηρίζεται στην ύπαρξη μίας έμπιστης τρίτης οντότητας με τα γνωστά προβλήματα ασφάλειας και διαθεσιμότητας που αυτό συνεπάγεται. Επιπλέον η χρήση μίας έμπιστης τρίτης οντότητας παραβιάζει την κατανεμημένη φύση των group communication συστημάτων. Οι αρχιτεκτονικές στις οποίες ο έλεγχος πρόσβασης βασίζεται στα μέλη του group είναι οι πλέον κατάλληλες για κατανεμημένα συστήματα αλλά απαιτούν πρωτόκολλα με μεγάλο υπολογιστικό και επικοινωνιακό κόστος.

### Ψηφιακές υπογραφές

Ανάλογα με την πολιτική πρόσβασης που θα ακολουθήσει κάθε group μπορούν να χρησιμοποιηθούν διάφορα σχήματα ψηφιακών υπογραφών. Θα παρουσιάσουμε στην συνέχεια ορισμένα σχήματα υπογραφών και θα αναλύσουμε τα ιδιαίτερα χαρακτηριστικά κάθε σχήματος.

#### *Απλές Ψηφιακές Υπογραφές (Plain Digital Signatures)*

Οι απλές ψηφιακές υπογραφές μπορούν να χρησιμοποιηθούν για όλες τις πολιτικές ελέγχου πρόσβασης. Στην περίπτωση της APT\_ACL πολιτικής ένα πιθανό μέλος αποδεικνύει ότι είναι κάτοχος ενός PKC που απαριθμείται σε μια ACL με την ψηφιακή υπογραφή ενός μηνύματος.

Αν ο έλεγχος πρόσβασης γίνεται μέσω ενός GAUTH (APT\_GAUTH) το μέλος (ή ο GAUTH ή και οι δύο) δημιουργεί ένα ζεύγος κλειδιών που χρησιμοποιείται για την δημιουργία ψηφιακών υπογραφών. Το ζεύγος κλειδιών σε αυτήν την πολιτική αποδεικνύει ότι ο κάτοχος του είναι μέλος του group. Αν ο GAUTH δεν αναφέρει στο GMC το όνομα του μέλους τότε το μέλος παραμένει ανώνυμο για όλους έκτος από τον GAUTH.

Όταν υιοθετείται από το group η APT\_GROUP πολιτική, η χρήση απλών ψηφιακών υπογραφών είναι ιδιαίτερα απλή τόσο σε δυναμικά όσο και σε στατικά groups. Ένα μέλος του group απλά υπογράφει μία αίτηση πρόσβασης ενός υποψηφίου μέλους. Ένα σημαντικό πλεονέκτημα των απλών ψηφιακών υπογραφών είναι ότι κάθε μέλος μπορεί να υπογράψει ασύγχρονα μία αίτηση πρόσβασης. Το κυριότερο

μειονέκτημα της χρήσης απλών ψηφιακών υπογραφών σε αυτήν την πολιτική είναι ότι το υποψήφιο μέλος πρέπει να συγκεντρώσει έναν ελάχιστο αριθμό υπογραφών πριν απευθύνθει στον GAUTH. Επιπρόσθετα το γεγονός ότι πρέπει να ελεγχθούν όλες οι ψηφιακές υπογραφές μπορεί να δημιουργήσει σημαντική επιβάρυνση.

### *Υπογραφές που απαιτούν την συμμέτοχή ενός αριθμού μελών (Threshold Signatures)*

Οι threshold signatures είναι μία πολλά υποσχόμενη κατεύθυνση για την επίλυση του προβλήματος έλεγχου πρόσβασης σε ένα group communication σύστημα. Η threshold κρυπτογραφία αναπτύχθηκε αρχικά από τους Desmet και Frankel [30].

Σε ένα  $(k,n)$  threshold σχήμα, ένα μυστικό (κλειδί) σπάει σε  $n$  μέρη και καθένα από αυτά μοιράζεται σε ένα μέλη. Για την ανακατασκευή του κλειδιού απαιτείται η συμμετοχή τουλάχιστον  $k$  μελών.

Αρχικά τα threshold σχήματα χρησιμοποιήθηκαν για την προστασία του κλειδιού μίας αρχής πιστοποίησης. Στα μοντέρνα threshold σχήματα ψηφιακών υπογραφών δεν απαιτείται η ανακατασκευή του μυστικού κλειδιού αντί αυτού γίνεται χρήση διαχωρισμού των ενεργειών (function sharing) για τον υπολογισμό της ψηφιακής υπογραφής.

Τα threshold σχήματα έχουν νόημα μόνο όταν υιοθετείται για τον έλεγχο πρόσβασης η APT\_GROUP πολιτική. Υπάρχουν αρκετές παραλλαγές των threshold υπογραφών, καθένα από αυτά τα σχήματα υπογραφών δεν θα εξεταστεί αναλυτικά. Ζητούμενο στα threshold σχήματα είναι η δυνατότητα δυναμικής αλλαγής του αριθμού των μελών που συμμετέχουν ώστε να μπορούν να χρησιμοποιηθούν σε APT\_GROUP.Dynamic πολιτική. Στην συνέχεια θα δούμε κάποια σχήματα που στηρίζονται σε σταθερά (fixed) και σε δυναμικά (dynamic) thresholds.

#### **Fixed Threshold:**

Όσον αφορά τα σταθερά thresholds θα επικεντρωθούμε στο RSA σχήμα υπογραφών. Στο threshold RSA σχήμα κάθε μέλος κατέχει ένα μυστικό μερίδιο (secret share) που αντιστοιχήθηκε σε αυτόν από μία οντότητα που είναι επιφορτισμένη με αυτή την λειτουργία (dealer). Όταν ο  $M_{new}$  ζητά πρόσβαση στο group κάθε διαθέσιμο μέλος  $M_i$  στέλνει το κομμάτι της υπογραφής που του αντιστοιχεί στον  $M_{new}$  εφ' όσον επιθυμεί την είσοδό του στο σύστημα. Ο  $M_{new}$  δεν έχει την δυνατότητα να συνθέσει την υπογραφή όταν λάβει τους απαιτούμενους  $k$  ψήφους (κομμάτια) από τα τρέχοντα μέλη του group, μίας και δεν μπορεί να υπολογίσει τους συντελεστές Lagrange. Μία

λύση στο πρόβλημα είναι η προώθηση των ψήφων που έλαβε ο  $M_{new}$  σε ένα έμπιστο dealer ο οποίος θα δημιουργήσει και θα επιστρέψει στον  $M_{new}$  την threshold υπογραφή. Αν τυχάνει ο dealer να είναι ταυτόχρονα και ο GAUTH τότε με την αποστολή των ψήφων μπορεί να δημιουργήσει ταυτόχρονα το GMC για τον  $M_{new}$ .

Ένα εναλλακτικό σχήμα προτάθηκε από τους Frankel κ. α. [31]. Στο σχήμα αυτό οι απαιτούμενοι συντελεστές Lagrange εμπεριέχονται στους ψήφους. Το πρόβλημα σ' αυτό το σχήμα είναι ότι απαιτείται ο  $M_{new}$  να γνωρίζει από πριν τα μέλη που θα συμμετάσχουν στην ψηφοφορία. Προκειμένου να αποκτηθεί αυτή η πληροφορία απαιτούνται δύο επιπλέον επικοινωνιακοί γύροι.

### **Dynamic Threshold:**

Η ανάπτυξη αποτελεσματικών threshold σχημάτων ψηφιακών υπογραφών όταν το threshold είναι δυναμικό αποτελεί μία πρόκληση για την επιστημονική κοινότητα της κρυπτογραφίας.

Μία προσπάθεια για την δημιουργία ενός τέτοιου σχήματος ψηφιακών υπογραφών [31] κάνει χρήση προδραστικής κρυπτογραφίας (proactive cryptography) για να υπολογίσει δυναμικά το threshold ( $t$ ). Το σχήμα αυτό απαιτεί πολλούς επικοινωνιακούς γύρους και την on – line κατάσταση όλων των μελών (κάτι που δεν αποτελεί πρόβλημα στα σύγχρονα group communication συστήματα που μας ενδιαφέρουν).

Μία ποιο πρόσφατη εργασία του Kong, κ.α. [32] προτείνει ένα πλήρως κατανεμημένο  $(k,n)$  σύστημα ψηφιακών υπογραφών στο οποίο ο ρόλος του dealer κατανέμεται μεταξύ των μελών του group όπου το  $k$  είναι σταθερό. Αρχικά ένας έμπιστος dealer εμπλέκεται για την δημιουργία ενός RSA modulus  $n$  και μία μυστική συνάρτηση  $f(x)$  όπου το  $f(ID)$  κατανέμεται μεταξύ των  $k$  μελών. Ο τρόπος με τον οποίο τα πρώτα  $k$  μέλη έλαβαν το δικαίωμα πρόσβασης στο group δεν διευκρινίζεται σε αυτό το σχήμα. Όταν ένας υποψήφιος χρήστης ζητά δικαίωμα πρόσβασης στο group κάθε μέλος υπογράφει ένα μερικό πιστοποιητικό χρησιμοποιώντας τον προσδιοριστή Lagrange που του αντιστοιχεί και ένα μυστικό μερίδιο (κομμάτι ενός μυστικού αριθμού). Το υποψήφιο μέλος αφού λάβει τα παραπάνω μηνύματα από τα μέλη του group υπολογίζει το GMC και ανακτά το μυστικό του μερίδιο. Το σχήμα αυτό φαντάζει ιδανική λύση για το πρόβλημα έλεγχου πρόσβασης σε ένα group communication σύστημα γιατί απαιτεί ελάχιστη ανάμιξη μίας έμπιστης τρίτης οντότητας ενώ τον ρόλο του GAUTH διατελεί το ίδιο το group. Παρ' όλα αυτά

πρέπει να σημειώσουμε ότι είναι ένα σχήμα που απαιτεί μεγάλο επικοινωνιακό και υπολογιστικό κόστος.

### **Accountable Subgroup Multisignatures**

Ένας άλλος τύπος ψηφιακών υπογραφών που μπορεί να χρησιμοποιηθεί για τον έλεγχο πρόσβασης σε ένα group καλείται “Accountable Subgroup Multisignatures ASM” και έχει προταθεί από Ohta, κ.α. [33]. Το ASM σχήμα δίνει την δυνατότητα σε οποιαδήποτε υποομάδα (subgrop),  $S$ , μίας ομάδας (group),  $G$ , να υπογράφει ψηφιακά εκ μέρους της ομάδας κατά τέτοιο τρόπο ώστε οποιαδήποτε οντότητα θέλει να επαληθεύσει την υπογραφή να μπορεί να αποκαλύψει την ταυτότητα των μελών που συμμετείχαν στην δημιουργία της.

Το σχήμα ASM εκμεταλλεύεται τα πλεονεκτήματα της ομομορφικής ιδιότητας των ψηφιακών υπογραφών του Schonorr [34]. Στο ASM για να υπογράφει ψηφιακά ένα μήνυμα (μία αίτηση πρόσβασης στην περίπτωση μας) πρέπει κάθε μέλος του group να στείλει στο υποψήφιο μέλος (verifier) ένα μήνυμα δέσμευσης (partial commitment). Όταν ο verifier λάβει  $k$  commitments τις πολλαπλασιάζει για να αποκτήσει την δέσμευση εισόδου (join commitment). Η join commitment αποστέλλεται στην συνέχεια στα  $k$  μέλη. Η join commitment περιέχει τα ονόματα όλων των μελών που έλαβαν μέρος στην παραπάνω διαδικασία. Κάθε μέλος υπολογίζει μία μερική υπογραφή (partial signature) και την αποστέλλει στο υποψήφιο μέλος. Όταν το υποψήφιο μέλος συγκεντρώσει τις  $k$  μερικές υπογραφές τις αθροίζει προκειμένου να δημιουργήσει ολόκληρη την υπογραφή.

Το ASM σχήμα απαιτεί δύο modular εκθετοποιήσεις και  $k$  modular πολλαπλασιασμούς. Αυτό είναι πολύ αποτελεσματικό μίας και σε διαφορετική περίπτωση (απλές υπογραφές) θα χρειάζονταν  $k$  επαληθεύσεις ψηφιακών υπογραφών. Ένα άλλο πλεονέκτημα του ASM σχήματος είναι η μη άρνηση αποδοχής ευθύνης από τα μέλη που συμμετέχουν. Επιπλέον σε αντίθεση με τις threshold υπογραφές δεν απαιτείται η ύπαρξη dealer, παρ' όλα αυτά η ύπαρξη ενός GAUTH κρίνεται σκόπιμη για την δημιουργία των GMCs. Αν δεν χρησιμοποιηθεί ένας GAUTH τότε απαιτητέ η επίδειξη των πιστοποιητικών όλων των μελών που υπέγραψαν για την είσοδο του νέου μέλους κάτι που είναι ιδιαίτερα πολύπλοκο ιδικά σε δυναμικά groups.

## Ομαδικές υπογραφές (Group Signatures)

Σε ένα σχήμα ομαδικής υπογραφής (π.χ. [35,36]) όλα τα μέλη είναι ισότιμα (peers) και μπορούν να υπογράψουν ανώνυμα χωρίς να μπορεί να διασυνδεθεί η υπογραφή τους με την ταυτότητά τους (unlinkability). Η ιδιότητα αυτή δεν μπορεί να εξασφαλίσει την αρχή της μη αποποίησης ευθύνης των μελών. Τα περισσότερα συστήματα ομαδικών υπογραφών προκειμένου να αντιμετωπιστεί το παραπάνω πρόβλημα δίνουν την δυνατότητα σε ένα εξουσιοδοτημένο μέλος του group (Group Manager) να ανοίγει τις ψηφιακές υπογραφές των υπόλοιπων μελών και να προσδιορίζει την ταυτότητα του υπογράφοντα.

Ουσιαστικά το μόνο πρόβλημα των ομαδικών υπογραφών προκειμένου να χρησιμοποιηθούν για τον έλεγχο πρόσβασης είναι η αδυναμία να συσχετιστεί μία υπογραφή με μία οντότητα. Η ανάπτυξη ορισμένων απλών επιπρόσθετων τεχνικών μπορεί να λύσει αποτελεσματικά αυτό το πρόβλημα. Μία τέτοια τεχνική [37] αναπτύχθηκε από τους Ateniese, κ.α. και επιτρέπει την δημιουργία ενός σχήματος υπογραφών υποομάδας από κάποιο σχήμα υπογραφών ομάδας.

Συνοψίζοντας τα παραπάνω σχήματα υπογραφών παρατηρούμε ότι οι απλές ψηφιακές υπογραφές και το σχήμα ASM μπορούν να χρησιμοποιηθούν για να αποδειχτεί το σύνολο των μελών της ομάδας (membership) αντίθετα οι threshold υπογραφές δεν παρέχουν τέτοια δυνατότητα. Τα σχήματα των απλών και ομαδικών υπογραφών είναι τα γενικότερα και μπορούν να χρησιμοποιηθούν στην περίπτωση που δεν υπάρχει γνώση του membership και on-line παρουσία των μελών του group (ασύγχρονα συστήματα). Τέλος αναφέρουμε ότι μόνο το ASM σχήμα και αυτό των απλών υπογραφών αποδεικνύουν την ταυτότητα του μέλους που υπογράφει για την είσοδο ενός νέου μέλους. Οι ιδιότητες των παραπάνω σχημάτων ψηφιακών υπογραφών παρουσιάζονται συνοπτικά στον παρακάτω πίνακα.

Key Features

Signature Type	Prove membership	On-line presence	Membership awareness	Accountability	Anonymity	Unlinkability
Plain	YES	NO	NO	YES	YES	NO
ASM-s	YES	YES	YES	YES	YES	NO
Threshold fixed	NO	YES	NO	NO	YES	YES
Threshold dynamic	NO	YES	YES	NO	YES	YES
Group	YES	NO	NO	NO*	YES	YES*

## **“Bouncer” – Ένα εργαλείο για τον έλεγχο πρόσβασης σε ένα group communication σύστημα**

Ο Bouncer [55, 60] είναι ένα εργαλείο που παρέχει υπηρεσίες ελέγχου πρόσβασης σε σύγχρονα και ασύγχρονα peer to peer groups. Η υλοποίηση του στηρίζεται στο πλαίσιο εργασίας που περιγράψαμε παραπάνω. Ο Bouncer υλοποιεί τέσσερις πολιτικές πρόσβασης:

- Πολιτική που στηρίζεται σε απλές ψηφιακές υπογραφές
- Πολιτική που στηρίζεται στο σχήμα υπογραφών TS-RSA
- Πολιτική που στηρίζεται στο σχήμα υπογραφών TS-DSA
- Πολιτική που στηρίζεται στο σχήμα υπογραφών ASM

Ένα βασικό πρόβλημα που προκύπτει στην προσπάθεια υλοποίησης των παραπάνω πολιτικών είναι ότι ορισμένες από αυτές (TS-RSA, TS-DSA, ASM) απαιτούν τα μέλη του group να έχουν γνώση του συνόλου των μελών που είναι την τρέχουσα στιγμή συνδεδεμένα (current membership), προκειμένου να είναι δυνατός ο υπολογισμός του ορίου (threshold). Στα σύγχρονα groups communication συστήματα η γνώση του current membership από όλα τα μέλη είναι εύκολο να επιτευχθεί ενώ τα περισσότερα σύγχρονα group communication συστήματα δίνουν την παραπάνω δυνατότητα στα μέλη τους χωρίς την ανάγκη τροποποιήσεων. Η γνώση όμως του current membership είναι δύσκολο να επιτευχθεί στα ασύγχρονα peer groups. Ένας τρόπος για να επιλυθεί το πρόβλημα γνώσης του current membership στα ασύγχρονα συστήματα είναι η χρήση μίας έμπιστης οντότητας που θα αναφέρεται σαν GAUTH. Όλα τα μέλη του group περιοδικά στέλνουν μηνύματα με τα οποία γνωστοποιούν στον GAUTH ότι παραμένουν συνδεδεμένα στο group.

Όπως έχει ήδη αναφερθεί τα ασύγχρονα peer groups δεν αποτελούν αντικείμενο μελέτης αυτής της εργασίας. Η αναφορά τους στην παραπάνω παράγραφο έγινε μόνο για λόγους πληρότητας της περιγραφής του Bouncer. Κάτι ακόμα που πρέπει να σημειωθεί είναι ότι ο ρόλος του GAUTH στον Bouncer είναι πολλαπλός. Η κυριότερη λειτουργία του στα σύγχρονα group communication συστήματα είναι η δημιουργία των group membership certificates (GMSs). Για τα GMSs και τον ρόλο του GAUTH στα ασύγχρονα group communication συστήματα θα γίνει αναλυτική αναφορά στην συνέχεια.

## **Πιστοποιητικά**

Στα περισσότερα group communication συστήματα ο προσδιοριστής με τον οποίο θα αναγνωρίζεται μοναδικά ένα μέλος από το group είναι συνήθως η IP διεύθυνση του ή το DNS name του. Επειδή οι προσδιοριστές τέτοιου τύπου μπορούν να υπολογιστούν με ευκολία από επίδοξους εισβολείς το bouncer κάνει χρήση των πιστοποιητικών δημοσίου κλειδιού (PKC) κάθε μέλους για τον υπολογιστή του προσδιοριστή που θα τους αντιστοιχηθεί. Το μορφότυπο των πιστοποιητικών ακολουθεί το X.509v3 πρότυπο ενώ τα πιστοποιητικά πρέπει να είναι υπογεγραμμένα από μία έγκυρη αρχή πιστοποίησης (CA).

Ο Bouncer δεν κάνει χρήση σταθερών μακράς διάρκειας πιστοποιητικών για την ταυτοποίηση των νεοεισελθέντων μελών, από τα άλλα μέλη του group άλλα χρησιμοποιεί ένα ειδικού τύπου πιστοποιητικό που θα αναφέρεται ως (Group Membership Certificate – GMC). Το GMC αποδίδεται σε ένα μέλος αφού έχει ολοκληρώσει με επιτυχία την διαδικασία ελέγχου πρόσβασης και αποτελεί ουσιαστικά την απόδειξη ότι έχει γίνει αποδεκτό στο group. Η δομή του GMC είναι παρόμοια με αυτή του PKC. Το GMC περιέχει το όνομα του group και πεδία για την εισαγωγή της ώρας πιστοποίησης και την περίοδο ισχύος του. Επιπλέον κάθε GMC πρέπει να περιέχει το PKC της οντότητας στην οποία έχει αποδοθεί ή κάποιο άλλο δημόσιο κλειδί το οποίο θα προσδιορίζει με μοναδικό τρόπο τον κάτοχό του. Ένα μέλος μπορεί να αποδείξει ότι ανήκει σε ένα group με την επίδειξη του GMC και αποδεικνύοντας ότι γνωρίζει το μυστικό κλειδί που αντιστοιχεί στο δημόσιο που περιέχει το πιστοποιητικό, υπογράφοντας για παράδειγμα ένα μήνυμα.

Ένα πρόβλημα του Bouncer σχετικά με τα πιστοποιητικά είναι ότι δεν υλοποιεί κάποια πολιτική για την ανάκληση GMCs.

## **Έλεγχος Πρόσβασης**

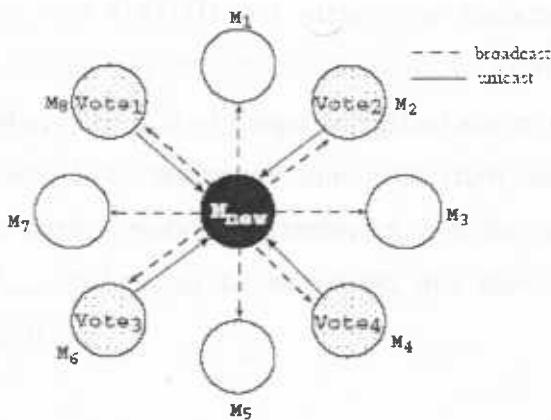
Προκειμένου να γίνει ευκολότερη και ποιο κατανοητή η περιγραφή της διαδικασίας ελέγχου πρόσβασης θα παρατεθεί η σημειολογία που θα χρησιμοποιηθεί.

Πριν προχωρήσουμε στην περιγραφή των πρωτοκόλλων του Bouncer θα δώσουμε μία συνοπτική εικόνα του πλαισίου εργασίας που υιοθετείται. Ορισμένες έννοιες έχουν ήδη αναφερθεί στην προηγούμενη ενότητα. Στην παρούσα περιγραφή θα αναφερθούν συνοπτικά και θα εξεταστούν από μία ποίο τεχνική σκοπιά.

GAUTH	Αρχή του group (Group Authority)
n	ο συνολικός αριθμός των μελών του group
i, j	δείκτες που προσδιορίζουν τα μέλη $0 < i, j < n$
$M_i, M_j$	το i και το j μέλος αντίστοιχα
$id_i$	ο προσδιοριστής του μέλους $M_i$
$PKC_i$	το πιστοποιητικό δημόσιου κλειδιού του μέλους $M_i$
$GMC_i$	το πιστοποιητικό του μέλους i που του εξασφαλίζει είσοδο στο group
$SK_i, PK_i$	το μυστικό και το δημόσιο κλειδί του μέλους i αντίστοιχα
GSK	το μυστικό κλειδί του group
$S_i(x)$	η ψηφιακή υπογραφή του μηνύματος x που δημιουργήθηκε με το $SK_i$
$ss_i$	το μυστικό κομμάτι (share) του μέλους $M_i$
$pss_j(i)$	το μερικό μυστικό κομμάτι (share) του $M_i$ από τον $M_j$
SL <sub>i</sub>	λίστα που περιέχει τα μέλη που υπογράφουν

*Σημειολογία*

Η διαδικασία πρόσβασης ενός νέου μέλους έστω  $M_{new}$  σε ένα group απεικονίζεται αφαιρετικά στο παρακάτω σχήμα.



Τα βήματα που απαιτούνται για την ολοκλήρωση της διαδικασίας έλεγχου πρόσβασης είναι τα ακόλουθα.

- Εκκίνηση:** Το υποψήφιο μέλος  $M_{new}$  αποκτά το group charter του group που επιθυμεί να συνδεθεί και μαθαίνει το μέγεθος του group είτε από τον GAUTH είτε από κάποιο μέλος που είναι επιφορτισμένο με αυτό το καθήκον (συνήθως είναι ο δημιουργός του group). Το group charter έχει την δομή και

τις πληροφορίες που αναφέραμε στην προηγούμενη ενότητα του ελέγχου πρόσβασης.

2. **Αίτηση Σύνδεσης:** Το πρωτόκολλο αρχίζει με την αποστολή από τον  $M_{new}$  ενός μηνύματος αίτησης πρόσβασης (JOIN\_REQ) στο group. Το μήνυμα αυτό είναι υπογεγραμμένο ψηφιακά από τον  $M_{new}$  και περιέχει το όνομα του group στο οποίο επιθυμεί να συνδεθεί και το PKC του ή το δημόσιο κλειδί του.
3. **Λήψη απόφασης εισόδου του  $M_{new}$ :** Με την λήψη του μηνύματος σύνδεσης κάθε μέλος του group εξάγει αρχικά το PKC του  $M_{new}$  και επαληθεύει την ψηφιακή υπογραφή του μηνύματος (JOIN\_REQ). Αν το μέλος αποδέχεται την είσοδο του  $M_{new}$  στο group, του απαντά με ένα υπογεγραμμένο μήνυμα (JOIN REP). Στην συνέχεια ο  $M_{new}$  συλλέγει τόσες υπογραφές όσες το όριο (threshold) που αναφέρεται στο group charter και επαληθεύει τις ψηφιακές υπογραφές των μηνυμάτων που έλαβε.
4. **Δημιουργία του GMC<sub>new</sub>:** Το ποίος είναι υπεύθυνος για την πιστοποίηση του GMC<sub>new</sub> του μέλουν  $M_{new}$  εξαρτάται από την πολιτική ελέγχου πρόσβασης του group. Αν στην πολιτική αναφέρεται η χρήση κάποιου προϋπάρχοντος GAUTH, τότε με την συλλογή επαρκούς αριθμού υπογραφών ο  $M_{new}$  στέλνει στο GAUTH μία αίτηση για απόκτηση πιστοποιητικού (GMC\_REQ). Το μήνυμα της αίτησης περιέχει το πιστοποιητικό δημόσιου κλειδιού του  $M_{new}$  (PKC<sub>new</sub>) το όνομα του group και το σύνολο των ψήφων που συγκέντρωσε. Στην περίπτωση που η πολιτική ασφάλειας του group ακολουθεί μία κατανεμημένη αρχιτεκτονική που δεν υιοθετεί την χρήση GAUTH, ο  $M_{new}$  επαληθεύει τις υπογραφές που συνέλεξε και υπολογίζει μόνος του το GMC<sub>new</sub>.

### Σχεδιασμός των συστήματος

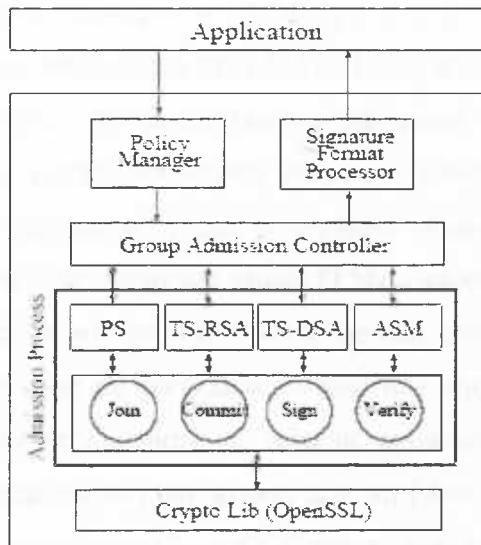
Το σύστημα έλεγχου πρόσβασης Bouncer αποτελείται από τέσσερα τμήματα (components). Τα τμήματα αυτά είναι τα:

- Policy manager: Ελέγχει αν το υποψήφιο μέλος συμμορφώνεται στην πολιτική που ορίζεται στο group charter. Αρχικά ο policy manager ελέγχει τον τύπο του ορίου (threshold), αν ο τύπος είναι στατικός ελέγχει αν ο αριθμός των συνδεδεμένων μελών είναι ίσος ή μεγαλύτερος του threshold. Στην περίπτωση που είναι κάτω από το threshold ενεργοποιεί την κατάλληλη

πολιτική που μπορεί να είναι η προώθηση του JOIN\_REQ σε έναν GAUTH ή η αλλαγή του threshold. Ο Bouncer υλοποιεί ένα μηχανισμό παραθύρου για την αλλαγή του threshold. Κάθε μέλος κρατά σε μία μεταβλητή  $n_{old}$  το μέγεθος του group κατά την τελευταία αλλαγή του threshold. Ο μηχανισμός αναθεώρησης του threshold ενεργοποιείται όταν η διαφορά μεταξύ του τρέχοντος αριθμού μελών του group  $n_{cur}$  και του  $n_{old}$  ξεπεράσει το μήκος του παραθύρου.

- Group admission controller: Περιέχει συναρτήσεις που χρησιμοποιούνται για τον προσδιορισμό του πρωτοκόλλου πρόσβασης και την διανομή των μηνυμάτων στις κατάλληλες βιβλιοθήκες.
- Βιβλιοθήκες που περιέχουν τα σχήματα ψηφιακών υπογραφών
- Signature format processor: Σκοπός του είναι η μετάφραση όλων ψηφιακών υπογραφών στο κατάλληλο μορφότυπο (format). Το format των GMC είναι το X.509v3 ενώ των ψηφιακών υπογραφών των μηνυμάτων το PKCS7.

Ο τρόπος με τον οποίο ενοποιούνται τα παραπάνω τμήματα στον Bouncer απεικονίζεται στο σχήμα που ακολουθεί.



## Πρωτόκολλα ελέγχου πρόσβασης

Στην ενότητα αυτή θα περιγράψουμε τα πρωτόκολλα έλεγχου πρόσβασης του Bouncer ακολουθώντας την κεντρικοποιημένη (χρήση μιας έμπιστης οντότητας σαν GAUTH) και την κατανεμημένη προσέγγιση.

### **Απλές ψηφιακές υπογραφές (Plain Signatures-PS)**

Τα σχήματα απλών ψηφιακών υπογραφών υιοθετούνται συχνά στις πολιτικές ελέγχου πρόσβασης. Αν και ο Bouncer ακολουθεί το RSA σχήμα υπογραφών οποιοδήποτε παρόμοιο σχήμα μπορεί να χρησιμοποιηθεί.

### Κεντρικοποιημένο RSA

Τα βήματα του κεντρικοποιημένου πρωτοκόλλου RSA είναι τα ακόλουθα:

1. Ένα πιθανό μέλος  $M_{new}$  επιδεικνύει το πιστοποιητικό δημοσίου κλειδιού που κατέχει  $PKC_{new}$  και υπογράφει με το αντίστοιχο ιδιωτικό κλειδί το  $JOIN\_REQ$  μήνυμα ( $m_1$ ) για να αποδείξει ότι το γνωρίζει. Στην συνέχεια ο  $M_{new}$  περιμένει για την συγκέντρωση του απαιτούμενου αριθμού (έστω  $t$ ) ψηφιακών υπογραφών από τα τρέχοντα μέλη του group.
2. Το τρέχον μέλος  $M_i$  υπογράφει το μήνυμα - ψήφο  $(m_1||id_{new})$  χρησιμοποιώντας το σχήμα απλών ψηφιακών υπογραφών RSA και το αποστέλλει περικλείοντας το  $GMC$  του ( $GMC_i$ ) στον  $M_{new}$ .
3. Ο  $M_{new}$  επαληθεύει τουλάχιστον  $t$   $GMC_i$  ( $j \in R i, |j| = t$ ), συλλέγει τις ψήφους και στην συνέχεια αποστέλλει στον GAUTH τις ψήφους που συνέλεξε μαζί με το  $GMC\_REQ_{new}$  μηνύματος και την υπογραφή του. Το  $GMC\_REQ_{new}$  είναι μία αίτηση για την απόκτηση ενός πιστοποιητικού τύπου X.509 που περιέχει την ταυτότητα του  $M_{new}$ , το δημόσιο κλειδί του  $M_{new}$  ( $PK_{new}$ ) και άλλα στοιχεία όπως το όνομα του group. Ο  $M_{new}$  αποστέλλει επίσης την λίστα ( $SL_{new}$ ) όσον υπέγραψαν για την είσοδό του στο group ώστε, ο GAUTH να μπορεί να επαληθεύσει με ευκολία το σύνολο των ψήφων.
4. Ο GAUTH αφού επαληθεύσει τις ψήφους δημιουργεί το  $GMC_{new}$  για το μέλος  $M_{new}$ . Η επαλήθευση των ψήφων περιλαμβάνει και έλεγχο αν οι ψήφοι αυτοί αντιστοιχούν στον  $M_{new}$ . Ο έλεγχος αυτός επιτυγχάνεται με την διασταύρωση του  $id_{new}$  που περιέχει η κάθε ψήφος και του  $id_{new}$  του αιτούντος.

### Κατανεμημένο RSA

Τα βήματα του κατανεμημένου πρωτοκόλλου RSA είναι τα ακόλουθα:

1. Ένα πιθανό μέλος  $M_{new}$  επιδεικνύει το πιστοποιητικό δημοσίου κλειδιού που κατέχει  $PKC_{new}$  και υπογράφει με το αντίστοιχο ιδιωτικό κλειδί το

- GMC\_REQ<sub>new</sub> μήνυμα ( $m$ ) για να αποδείξει ότι το γνωρίζει. Στην συνέχεια ο  $M_{new}$  περιμένει για την συγκέντρωση του απαιτούμενου αριθμού (έστω  $t$ ) ψηφιακών υπογραφών από τα τρέχοντα μέλη του group.
2. Το τρέχον μέλος  $M_i$  υπογράφει το μήνυμα – ψήφο ( $m$ ) χρησιμοποιώντας το σχήμα απλών ψηφιακών υπογραφών RSA και το αποστέλλει περικλείοντας το GMC του ( $GMC_i$ ) στον  $M_{new}$ .
  3. Ο  $M_{new}$  δημιουργεί μόνος του το  $GMC_{new}$  που περιέχει τα  $\{GMC\_REQ_{new} \parallel SL_{new} \parallel S_1 \parallel \dots \parallel S_t\}$ , όπου  $S_i$  είναι το  $(GMC\_REC)^{ski} \text{ mod } n_i$ .

Ένα μειονέκτημα των απλών υπογραφών είναι ότι η επαλήθευση κάθε μίας υπογραφής ανεξάρτητα δημιουργεί επιπρόσθετο κόστος, ένα άλλο μειονέκτημα είναι η ανάγκη συγκέντρωσης από το υποψήφιο μέλος ενός αριθμού υπογραφών πριν προσεγγίσει τον GAUTH.

Η απλότητα του παραπάνω σχήματος και η δυνατότητα να χρησιμοποιηθεί και σε ασύγχρονα συστήματα αποτελούν τα βασικά του πλεονεκτήματα.

### ***Threshold RSA – TS-RSA***

To TS-RSA [56, 57, 58] σχήμα κάνει χρήση μίας έμπιστης οντότητας (trusted dealer). Τον ρόλο του dealer μπορεί να επιτελέσει ο GAUTH ή ο δημιουργός του group. Στον Bouncer επιλέγεται σαν dealer ο δημιουργός του group. Ο dealer εμπλέκεται στην δημιουργία ενός RSA modulus  $N$  που θα είναι κοινό για το group. Επιπλέον χρησιμοποιείται μία μυστική συνάρτηση  $f(x) = a_0 + a_1x + a_2x^2 + \dots + a_{t-1}x^{t-1}$  ( $\text{mod } N$ ) όπου  $a_0$  είναι ένα μυστικό κλειδί του group (GSK). Ο dealer διανέμει ένα μυστικό κομμάτι (secret share)  $ss_i = f(i)$  στα μέλη  $M_i$ s που συμμετείχαν στο group από την δημιουργία του και στην συνέχεια σταματάει την λειτουργία του.

### **Κεντρικοποιημένο TS-RSA**

Τα βήματα του κεντρικοποιημένου TS-RSA πρωτοκόλλου είναι τα ακόλουθα:

1. Ένα πιθανό μέλος  $M_{new}$  επιδεικνύει το πιστοποιητικό δημοσίου κλειδιού που κατέχει  $PKC_{new}$  και υπογράφει με το αντίστοιχο ιδιωτικό κλειδί το JOIN\_REQ μήνυμα ( $m_1$ ) για να αποδείξει ότι το γνωρίζει. Στην συνέχεια ο  $M_{new}$  περιμένει για την συγκέντρωση του απαιτούμενου αριθμού (έστω  $t$ ) ψηφιακών υπογραφών από τα τρέχοντα μέλη του group.

2. Τα τρέχοντα μέλη αφού επαληθεύσουν το πιστοποιητικό του  $M_{new}$  ( $PKC_{new}$ ) απαντούν με την αποστολή των  $GMC_i$  τους στον  $M_{new}$ .
3. Ο  $M_{new}$  διαλέγει  $t$  από τα  $GMC_j$  ( $j \in R$   $i, |j| = t$ ) των μελών που του απάντησαν και καταρτίζει την λίστα των υπογραφόντων ( $SL_{new}$ ) για την είσοδό του στο group. Στην συνέχεια ο  $M_{new}$  στέλνει την λίστα στα μέλη που του έστειλαν το  $GMC$  τους. Ο λόγος που απαιτείται η αποστολή της λίστας είναι για να γνωρίζει κάθε μέλος τους δείκτες των  $t$  μελών που επιλέχθηκαν προκειμένου να είναι δυνατός ο υπολογισμός του συντελεστή Lagrange που δίνεται από τον τύπο  $l_j(x)$  για τα μερικά κομμάτια (partial shares – pss) και τις μερικές ψηφιακές υπογραφές (partial signatures).
4. Κάθε μέλος  $M_j$  δημιουργεί την ψήφο του  $vote_j = (m_1 || id_{new})^{d_j} \pmod{N}$  για το μέλος  $M_{new}$ , όπου  $d_j$  ένα μερικό ιδιωτικό κλειδί με  $d_j = ssj * l_j(0) \pmod{N}$ . Επιπλέον ο  $M_j$  υπολογίζει το partial share  $pss_j(new) = ssj * l_j(new) \pmod{N}$ . Τέλος ο  $M_j$  στέλνει στον  $M_{new}$  τα  $vote_j$  και  $pss_j(new)$ .
5. Ο  $M_{new}$  ανακτά το secret share του προσθέτοντας τα μερικά secret shares  $pss_j(new)$  που έλαβε. Στην συνέχεια ζητά από τον GAUTH ένα GMC αποστέλλοντας τους ψήφους που συνέλεξε και την λίστα των μελών που υπέγραψαν για την είσοδο του στο group.
6. Ο GAUTH πολλαπλασιάζει τις  $t$  ψήφους και με κατάλληλες κρυπτογραφικές τεχνικές [54] ανακτά την υπογραφή. Αν η υπογραφή είναι έγκυρη τότε δημιουργεί το  $GMC_{new}$  του  $M_{new}$ .

### Κατανεμημένο TS-RSA

Τα βήματα του κατανεμημένου TS-RSA πρωτοκόλλου είναι τα ακόλουθα:

1. Ένα πιθανό μέλος  $M_{new}$  επιδεικνύει το πιστοποιητικό δημοσίου κλειδιού που κατέχει  $PKC_{new}$  και υπογράφει με το αντίστοιχο ιδιωτικό κλειδί το  $GMC\_REQ_{new}$  μήνυμα ( $m$ ) για να αποδείξει ότι το γνωρίζει. Στην συνέχεια ο  $M_{new}$  περιμένει για την συγκέντρωση των απαιτούμενου αριθμού (έστω  $t$ ) ψηφιακών υπογραφών από τα τρέχοντα μέλη του group.
2. Τα τρέχοντα μέλη αφού επαληθεύσουν το πιστοποιητικό του  $M_{new}$  ( $PKC_{new}$ ) απαντούν με την αποστολή των  $GMC_i$  τους στον  $M_{new}$ .
3. Ο  $M_{new}$  διαλέγει  $t$  από τα  $GMC_j$  ( $j \in R$   $i, |j| = t$ ) των μελών που του απάντησαν και καταρτίζει την λίστα των υπογραφόντων ( $SL_{new}$ ) για την είσοδό του στο group. Στην συνέχεια ο  $M_{new}$  στέλνει την λίστα στα μέλη που του έστειλαν το

GMC τους. Ο λόγος που απαιτείται η αποστολή της λίστας είναι για να γνωρίζει κάθε μέλος τους δείκτες των t μελών που επιλέχθηκαν προκειμένου να είναι δυνατός ο υπολογισμός του συντελεστή Lagrange που δίνεται από τον τύπο  $l_j(x)$  για τα μερικά κομμάτια (partial shares – pss) και τις μερικές ψηφιακές υπογραφές (partial signatures).

4. Κάθε μέλος  $M_j$  δημιουργεί την ψήφο του  $vote_j = (m_1)^{d_j} \pmod{N}$  για το μέλος  $M_{new}$ , όπου  $d_j$  ένα μερικό ιδιωτικό κλειδί με  $d_j = ssj * l_j(0) \pmod{N}$ . Επιπλέον ο  $M_j$  υπολογίζει το partial share  $pss_j(new) = ssj * l_j(new) \pmod{N}$ . Τέλος ο  $M_j$  στέλνει στον  $M_{new}$  τα  $vote_j$  και  $pss_j(new)$ .
5. Ο  $M_{new}$  πολλαπλασιάζει τις t ψήφους και με κατάλληλες κρυπτογραφικές τεχνικές [55] ανακτά την υπογραφή. Αν η υπογραφή είναι έγκυρη τότε δημιουργεί το  $GMC_{new} = \{GMC\_REQ_{new} || S\}$  όπου  $S = (GMC\_REC_{new})^{GSK} \pmod{N}$ .

Βασικό μειονέκτημα των παραπάνω πρωτοκόλλων είναι ότι δεν εξασφαλίζουν την ορθότητα του secret share μετά το άθροισμα των partial secret shares. Με άλλα λόγια δεν παρέχεται επαληθευσημότητα των secret shares. Το γεγονός αυτό προκαλεί κενά ασφάλειας στο πρωτόκολλο μίας και κακόβουλα ή διεφθαρμένα μέλη μπορεί να στείλουν ψεύτικα partial shares. Επιπλέον η τεχνική του random shuffling [56] απαιτείται για ασφαλή μεταφορά των secret shares.

### ***Threshold DSA – TS-DSA***

Το TS-DSA [59] σχήμα δεν απαιτεί την ύπαρξη dealer, t (threshold) ή περισσότερα μέλη μπορούν να αρχικοποιήσουν την εκτέλεση του πρωτοκόλλου.

### **Κεντρικοποιημένο TS-DSA**

Τα βήματα του κεντρικοποιημένου TS-DSA πρωτοκόλλου είναι τα ακόλουθα:

1. Ένα πιθανό μέλος  $M_{new}$  επιδεικνύει το πιστοποιητικό δημοσίου κλειδιού που κατέχει  $PKC_{new}$  και υπογράφει με το αντίστοιχο ιδιωτικό κλειδί το  $JOIN\_REQ$  μήνυμα  $(m_1)$  για να αποδείξει ότι το γνωρίζει. Στην συνέχεια ο  $M_{new}$  περιμένει για την συγκέντρωση του απαιτούμενου αριθμού (έστω t) ψηφιακών υπογραφών από τα τρέχοντα μέλη του group.
2. Τα τρέχοντα μέλη που επιθυμούν να συμμετέχουν στην διαδικασία ελέγχου πρόσβασης αφού επαληθεύσουν το  $PKC_{new}$  απαντούν στον  $M_{new}$  με την αποστολή των  $GMC_i$  τους.

3. Ο  $M_{new}$  διαλέγει τυχαία t από αυτούς που του απάντησαν ( $M_j$  με  $j \in R$  i,  $|j| = t$ ), συλλέγει τα  $id_j$  τους από τα αντίστοιχα GMC-s και καταρτίζει την λίστα υπογραφόντων  $SL_{new}$  την οποία αποστέλλει σε όλα τα μέλη  $M_j$  που του απάντησαν.
4. Κάθε ένας από τους  $M_j$  επιλέγει τυχαία δύο πολυώνυμα  $k_j(z)$ ,  $a_j(z)$  βαθμού t - 1 στο  $Z_q$  όπου  $t = \lceil \frac{t+1}{2} \rceil$ . Επιπλέον ισχύει  $k(z) = \sum_{j=1}^t k_j(z)$  και  $a(z) = \sum_{j=1}^t a_j(z)$ . Ο  $M_j$  υπολογίζει το  $k_j$  και  $a_j$  για όλους τους υπογράφοντες  $M_i (i = 1, \dots, t)$  στην  $SL_{new}$  και διανέμει τα  $k_j(i)$  και  $a_j(i)$  σε όλους τους συνυπογράφοντες. Μετά την λήψη του partial share του από τους άλλους συνυπογράφοντες ο  $M_j$  υπολογίζει το  $k_j$ ,  $a_j$  ως εξής:  $k_j = k(j) = \sum_{i=1}^t k_i(z) \pmod{q}$ ,  $a_j = a(j) = \sum_{i=1}^t a_i(z) \pmod{q}$ . Στην συνέχεια ο  $M_j$  υπολογίζει τα  $u_j = k_j a_i(j) \pmod{q}$ ,  $v_j = g^{a_j} \pmod{p}$  και τα αποστέλλει στον  $M_{new}$ .
5. Ο  $M_{new}$  υπολογίζει την ποσότητα  $R$  χωρίς να απαιτείται γνώση των  $k$  και  $k^{-1}$  ως εξής: Αρχικά υπολογίζει τα  $u$  και  $v$ ,  $u = \sum_{j=1}^t u_j l_j(0) \pmod{q} = ka \pmod{q}$ ,  $v = \prod_{j=1}^t (v_j)^{l_j(0)} \pmod{p} = g^a \pmod{p}$ . Στην συνέχεια ο  $M_{new}$  υπολογίζει το αντίστροφο του  $u$  ( $u^{-1}$ ) και την ποσότητα  $R = (v^u)^{-1} \pmod{p} \pmod{q} = (g^{k^{-1}})^{-1} \pmod{p} \pmod{q}$ . Τέλος ο  $M_{new}$  στέλνει το  $R$  στον  $M_j$ .
6. Ο  $M_j$  υπολογίζει την  $vote_j = k_j((m_1 || id_{new}) + x_j R) \pmod{q}$ , όπου  $x_j$  είναι το share του  $M_j$  από το μυστική πληροφορία του group (group secret)  $x$ . Επίσης ο  $M_j$  υπολογίζει το partial share του  $M_{new}$   $pss_j(new) = x_j l_j(new) \pmod{q}$ . Τέλος ο  $M_j$  στέλνει τα  $vote_j$  και  $pss_j(new)$  στον  $M_{new}$ .
7. Ο  $M_{new}$  στέλνει στον GAUTH τις t ψήφους που συνέλεξε, το  $R$  και την λίστα υπογραφόντων και ζητά την έκδοση ενός  $GMC_{new}$ .
8. Ο GAUTH υπολογίζει την υπογραφή  $S$  χρησιμοποιώντας τις t ψήφους και Lagrange. Στην συνέχεια επαληθεύει την ορθότητα του  $R$  και  $S$  με χρήση της διαδικασίας επαλήθευσης του DES και εκδίδει το  $GMC_{new}$ .

## Κατανεμημένο TS-DSA

Τα βήματα του κατανεμημένου TS-DSA πρωτοκόλλου είναι τα ακόλουθα:

1. Ένα πιθανό μέλος  $M_{new}$  επιδεικνύει το πιστοποιητικό δημοσίου κλειδιού που κατέχει  $PKC_{new}$  και υπογράφει με το αντίστοιχο ιδιωτικό κλειδί το  $GMC\_REQ_{new}$  μήνυμα ( $m$ ) για να αποδείξει ότι το γνωρίζει. Στην συνέχεια ο  $M_{new}$  περιμένει για την συγκέντρωση του απαιτούμενου αριθμού (έστω  $t$ ) ψηφιακών υπογραφών από τα τρέχοντα μέλη του group.
2. Τα τρέχοντα μέλη που επιθυμούν να συμμετέχουν στην διαδικασία ελέγχου πρόσβασης αφού επαληθεύσουν το  $PKC_{new}$  απαντούν στον  $M_{new}$  με την αποστολή των  $GMC_i$  τους.
3. Ο  $M_{new}$  διαλέγει τυχαία  $t$  από αυτούς που του απάντησαν ( $M_j$  με  $j \in R$   $i, |j| = t$ ), συλλέγει τα  $id_j$  τους από τα αντίστοιχα  $GMC$ -s και καταρτίζει την λίστα υπογραφόντων  $SL_{new}$  την οποία αποστέλλει σε όλα τα μέλη  $M_j$  που του απάντησαν.
4. Κάθε ένας από τους  $M_j$  επιλέγει τυχαία δύο πολυώνυμα  $k_j(z)$ ,  $a_j(z)$  βαθμού  $t - 1$  στο  $Z_q$  όπου  $t = \lfloor \frac{t+1}{2} \rfloor$ . Επιπλέον ισχύει  $k(z) = \sum_{j=1}^t k_j(z)$  και  $a(z) = \sum_{j=1}^t a_j(z)$ . Ο  $M_j$  υπολογίζει το  $k_j$  και  $a_j$  για όλους τους υπογράφοντες  $M_i$  ( $i = 1, \dots, t$ ) στην  $SL_{new}$  και διανέμει τα  $k_j(i)$  και  $a_j(i)$  σε όλους τους συνυπογράφοντες. Μετά την λήψη του partial share του από τους άλλους συνυπογράφοντες ο  $M_j$  υπολογίζει το  $k_j$ ,  $a_j$  ως εξής:  $k_j = k(j) = \sum_{i=1}^t k_i(z) \pmod{q}$ ,  $a_j = a(j) = \sum_{i=1}^t a_i(z) \pmod{q}$ . Στην συνέχεια ο  $M_j$  υπολογίζει τα  $u_j = k_j a_i(j) \pmod{q}$ ,  $v_j = g^{a_j} \pmod{p}$  και τα αποστέλλει στον  $M_{new}$ .
5. Ο  $M_{new}$  υπολογίζει την ποσότητα  $R$  χωρίς να απαιτείται γνώση των  $k$  και  $k^{-1}$  ως εξής: Αρχικά υπολογίζει τα  $u$  και  $v$ ,  $u = \sum_{j=1}^t u_j l_j(0) \pmod{q} = ka \pmod{q}$ ,  $v = \prod_{j=1}^t (v_j)^{l_j(0)} \pmod{p} = g^a \pmod{p}$ . Στην συνέχεια ο  $M_{new}$  υπολογίζει το αντίστροφο του  $u$  ( $u^{-1}$ ) και την ποσότητα  $R =$

- $(v^{u^{-1}} \bmod p) \bmod q = (g^{k^{-1}} \bmod p) \bmod q$ . Τέλος ο  $M_{new}$  στέλνει το  $R$  στον  $M_j$ .
6. Ο  $M_j$  υπολογίζει την  $vote_j = k_j((m) + x_jR) \bmod q$ , όπου  $x_j$  είναι το share του  $M_j$  από το μυστική πληροφορία του group (group secret)  $x$ . Επίσης ο  $M_j$  υπολογίζει το partial share του  $M_{new}$   $pss_j(new) = x_jl_j(new) \pmod{q}$ . Τέλος ο  $M_j$  στέλνει τα  $vote_j$  και  $pss_j(new)$  στον  $M_{new}$ .
  7. Ο  $M_{new}$  υπολογίζει την υπογραφή  $S$  χρησιμοποιώντας τις τ ψήφους και Lagrange. Στην συνέχεια επαληθεύει την ορθότητα του  $R$  και  $S$  με χρήση της διαδικασίας επαλήθευσης του DES και εκδίδει το  $GMC_{new} = \{GMC\_REC||R||S\}$  όπου  $R = (g^{k^{-1}} \bmod p) \bmod q$ ,  $S = (GMC\_REC_{new}) + xR \bmod q$

To TS-DSA σχήμα δίνει την δυνατότητα επαλήθευσης της ορθότητας των secret shares, προκειμένου όμως να μεταφερθεί το secret share με ασφάλεια απαιτείται η υλοποίηση της τεχνικής του random shuffling. To TS-DSA σχήμα παραμένει ασφαλές όταν υπάρχουν λιγότερα από  $t = \lfloor \frac{t+1}{2} \rfloor$  κακόβουλα μέλη. Τέλος πρέπει να αναφέρουμε ότι για την δημιουργία ενός τυχαίου μυστικού χωρίς την ανάμιξη του dealer απαιτούνται επιπλέον  $O(n^2)$  επικοινωνιακοί γύροι.

### ***Accountable Subgroup Multisignatures – ASM***

To ASM [60] σχήμα υπογραφών επιτρέπει οποιοδήποτε υπο-group ενός group πιθανών υπογραφόντων να υπογράψει κατά τέτοιο τρόπο ώστε η υπογραφή να αποκαλύπτει την ταυτότητα των υπογραφόντων.

### **Kεντρικοποιημένο ASM**

Τα βήματα του κεντρικοποιημένου ASM πρωτοκόλλου είναι τα ακόλουθα:

1. Ένα πιθανό μέλος  $M_{new}$  επιδεικνύει το πιστοποιητικό δημοσίου κλειδιού που κατέχει  $PKC_{new}$  και υπογράφει με το αντίστοιχο ιδιωτικό κλειδί το  $JOIN\_REQ$  μήνυμα  $(m_1)$  για να αποδείξει ότι το γνωρίζει. Στην συνέχεια ο  $M_{new}$  περιμένει για την συγκέντρωση του απαιτούμενου αριθμού (έστω  $t$ ) ψηφιακών υπογραφών από τα τρέχοντα μέλη του group.



2. Για να υπογράψει ένα μήνυμα, κάθε μέλος  $M_i$  στέλνει μία μερική δέσμευση (partial commitment)  $c_i = g^{r_i} \pmod{p}$  στον  $M_{new}$ .
3. Ο  $M_{new}$  πολλαπλασιάζει τα commitments για να δημιουργήσει την διεκδίκηση εισόδου (join challenge)  $E = H(m||C||SL_{new})$  όπου  $C = \prod c_i \pmod{p}$  και αποστέλλει το  $E$  το  $C$  και το  $SL_{new}$  στα τ μέλη.
4. Κάθε μέλος  $M_j$  αρχικά επαληθεύει την σύνοψη του  $E$  για να βεβαιωθεί ότι υπογράφει ένα μήνυμα για την είσοδο νέου μέλους στο group. Και στην συνέχεια υπολογίζει την ψήφο του  $vote_j = r_j + x_j(E||id_{new}) \pmod{q}$  και την αποστέλλει στον  $M_{new}$ .
5. Ο  $M_{new}$  συλλέγει τις ψήφους και τις αποστέλλει μαζί με το  $C$ ,  $SL_{new}$ , και  $GMC_{REQ_{new}}$  στον GAUTH.
6. Ο GAUTH επαληθεύει όλες τις ψήφους ελέγχοντας αν το  $g^S \pmod{p}$  ισούται με το  $CY^{(E \parallel id_{new})} \pmod{p}$  όπου  $S = \sum S_j \pmod{q}$  και  $Y = \prod y_i \pmod{p}$ .

Αν η επαλήθευση ολοκληρωθεί με επιτυχία τότε ο GAUTH δημιουργεί το  $GMC_{new}$ .

### Kατανεμημένο ASM

Τα βήματα του κατανεμημένου ASM πρωτοκόλλου είναι τα ακόλουθα:

1. Ένα πιθανό μέλος  $M_{new}$  επιδεικνύει το πιστοποιητικό δημοσίου κλειδιού που κατέχει  $PKC_{new}$  και υπογράφει με το αντίστοιχο ιδιωτικό κλειδί το  $GMC_{REQ_{new}}$  μήνυμα ( $m$ ) για να αποδείξει ότι το γνωρίζει. Στην συνέχεια ο  $M_{new}$  περιμένει για την συγκέντρωση του απαιτούμενου αριθμού (έστω  $t$ ) ψηφιακών υπογραφών από τα τρέχοντα μέλη του group.
2. Για να υπογράψει ένα μήνυμα, κάθε μέλος  $M_i$  στέλνει μία μερική δέσμευση (partial commitment)  $c_i = g^{r_i} \pmod{p}$  στον  $M_{new}$ .
3. Ο  $M_{new}$  πολλαπλασιάζει τα commitments για να δημιουργήσει την διεκδίκηση εισόδου (join challenge)  $E = H(m||C||SL_{new})$  όπου  $C = \prod c_i \pmod{p}$  και αποστέλλει το  $E$  το  $C$  και το  $SL_{new}$  στα τ μέλη.
4. Κάθε μέλος  $M_j$  αρχικά επαληθεύει την σύνοψη του  $E$  για να βεβαιωθεί ότι υπογράφει ένα μήνυμα για την είσοδο νέου μέλους στο group. Και στην



συνέχεια υπολογίζει την ψήφο του  $vote_j = r_j + x_j(E) \text{ mod } q$  και την αποστέλλει στον  $M_{new}$ .

5. Ο  $M_{new}$  επαληθεύει όλες τις ψήφους ελέγχοντας αν το  $g^S \pmod{p}$  ισούται με

το  $CY^{(E \parallel id_{new})} \pmod{p}$  όπου  $S = \sum S_j \pmod{q}$  και  $Y = \prod y_i \pmod{p}$ . Αν

η επαλήθευση ολοκληρωθεί με επιτυχία τότε ο  $M_{new}$  δημιουργεί το  $GMC_{new}$

$$= \{GMC\_REQ_{new} \parallel SL_{new} \parallel C \parallel E \parallel S\}, \text{ όπου } E = H(GMC\_REQ_{new} \parallel \prod g^{r_j} \parallel SL_{new}),$$

$$S = \sum r_j + x_j E$$

Ένα πλεονέκτημα του ASM είναι ότι η φάση επαλήθευσης απαιτεί μόνο δύο modular εκθετοποιήσεις και k modular πολλαπλασιασμούς, επίσης το ASM σχήμα δεν απαιτεί την ανάμιξη dealer ενώ επιπλέον επιτρέπει την εξακρίβωση της ταυτότητας των μελών που υπέγραψαν.

## Συμπεράσματα

Η αποτελεσματικότητα των μηχανισμών ελέγχου πρόσβασης είναι καθοριστική για την ασφάλεια αλλά και την αποτελεσματικότητα ενός group communication συστήματος. Συγκριτικά με τη σημασία των μηχανισμών έλεγχου πρόσβασης το ενδιαφέρον από την επιστημονική κοινότητα είναι μικρό. Τα περισσότερα συστήματα υιοθετούν κεντρικοποιημένες αρχιτεκτονικές για τον έλεγχο πρόσβασης, που στηρίζονται σε στατικές ACLs. Οι αρχιτεκτονικές αυτές έχουν δύο βασικά μειονεκτήματα. Το πρώτο είναι ότι είναι ότι η ύπαρξη μίας κεντρικής οντότητας επιφορτισμένης με τον έλεγχο πρόσβασης αποτελεί ένα σημείο συμφόρησης και αποτυχίας. Στην περίπτωση αποτυχίας της οντότητας αυτής το group communication σύστημα “καταρρέει”. Το δεύτερο μειονέκτημα είναι ότι η χρήση ACLs είναι αναποτελεσματική στα group communication συστήματα γιατί είναι αδύνατον να γνωρίζουμε από πριν όλα τα πιθανά μέλη, με αποτέλεσμα να περιορίζεται η αποτελεσματικότητα ολόκληρου του group communication συστήματος.

Από τα σχήματα ελέγχου πρόσβασης που περιγράφαμε σ' αυτήν την ενότητα καταλληλότερα φαίνονται τα σχήματα που υλοποιούνται στον Bouncer και το id-based σχήμα. Τα σχήματα των απλών και των ASM υπογραφών, πλεονεκτούν σε σχέση με τα threshold σχήματα, εξαιτίας της δυνατότητας που δίνουν για την

ανάκτηση της ταυτότητας των υπογραφόντων μελών. Ένα επιπλέον μειονέκτημα των threshold σχημάτων είναι το μεγάλο επικοινωνιακό τους κόστος και η ανάγκη για χρήση της τεχνικής του random shuffling για την μετάδοση των secret shares.

Όσον αφορά την επιλογή ανάμεσα στην κατανεμημένη ή κεντρικοποιημένη αρχιτεκτονική των παραπάνω πρωτοκόλλων ο βασικός παράγοντας που θα καθορίσει την επιλογή μας σχετίζεται με το μέγεθος του group. Στα μικρά groups η υιοθέτηση μίας κατανεμημένης αρχιτεκτονικής μπορεί να δώσει μία ικανοποιητική λύση. Στα μεγαλύτερα groups η χρήση μία κατανεμημένης αρχιτεκτονικής θα είχε σαν αποτέλεσμα πιστοποιητικά πολύ μεγάλου μεγέθους μίας και θα έπρεπε να περιέχει τις υπογραφές όλων των μελών (ο αριθμός των μελών ισούται με το threshold) που συμμετείχαν στην διαδικασία. Η υιοθέτηση μίας κεντρικοποιημένης PS ή ASM αρχιτεκτονικής θα εξασφάλιζε καλύτερη απόδοση στο group communication σύστημα. Το μόνο μειονέκτημα των κεντρικοποιημένων PS, ASM ή threshold υπογραφών είναι ότι η οντότητα που θα επιφορτίστει με αυτή την λειτουργία αποτελεί σημείο συμφόρησης και μοναδικό σημείο αποτυχίας μίας και η απόφαση πρόσβασης παίρνεται πάλι συνολικά από όλα τα μέλη του group.

Το id-based σχήμα είναι το πλέον κατάλληλο για τον έλεγχο πρόσβασης, υιοθετεί μία κατανεμημένη αρχιτεκτονική και δεν απαιτεί την ύπαρξη μίας έμπιστης τρίτης οντότητας. Επιπλέον η membership revocation πολιτική που υλοποιεί του δίνει συγκριτικό πλεονέκτημα σε σχέση με όλα τα άλλα σχήματα ελέγχου πρόσβασης.

## Διαχείριση μυστικού κλειδιού συνόδου ομάδας

Μέχρι στιγμής έχουμε εξετάσει μηχανισμούς αυθεντικοποίησης και έλεγχού πρόσβασης ενός υποψηφίου μέλουν σε ένα group. Σκοπός των μηχανισμών έλεγχου πρόσβασης είναι να εξετάσουν αν μία οντότητα δικαιούται την είσοδό της στο group ή όχι. Στο σημείο αυτό γίνεται υπαρκτή η ανάγκη ενός μηχανισμού που θα εξασφαλίζει ότι μόνο οι οντότητες που έχουν δικαίωμα πρόσβασης στο group θα μπορούν να λαμβάνουν και να στέλνουν μηνύματα σ' αυτό.

Μία λύση στο παραπάνω πρόβλημα είναι η κρυπτογράφηση των μηνυμάτων που ανταλλάσσουν τα μέλη του group. Τα μέλη που έχουν δικαίωμα εισόδου στο group θα γνωρίζουν ένα κοινό μυστικό κλειδί συνόδου με το οποίο θα κρυπτογραφούν και θα αποκρυπτογραφούν τα μηνύματα που θα ανταλλάσσουν με τα υπόλοιπα μέλη που

ανήκουν στο ίδιο group. Το πρόβλημα που τίθεται είναι ή ύπαρξη κατάλληλου μηχανισμού για την εγκαθίδρυση και η διαχείριση του μυστικού κλειδιού.

Έχουν προταθεί αρκετά πρωτόκολλα που επιτελούν την παραπάνω λειτουργία. Τα πρωτόκολλα αυτά μπορούν να κατανεμηθούν σε τρις ομάδες τα:

- Κεντρικοποιημένα. Όπου η διαχείριση του μυστικού κλειδιού συνόδου γίνεται από μια έμπιστη τρίτη οντότητα
- Μη κεντρικοποιημένα. Όπου η διαχείριση του μυστικού κλειδιού γίνεται από πολλές έμπιστες τρίτες οντότητες με σκοπό να διαμοιραστεί το φόρτο εργασίας που αναλάμβανε η μία και μόνη έμπιστη τρίτη οντότητα στις κεντρικοποιημένες μεθόδους.
- Κατανεμημένα. Όπου δεν γίνεται χρήση έμπιστης τρίτης οντότητας αλλά τα μέλη του group από κοινού αποφασίζουν εγκαθιδρύουν και διαχειρίζονται το μυστικό κλειδί συνόδου.

Ένας ακόμα διαχωρισμός που μπορεί να γίνει στα κατανεμημένα πρωτόκολλα διαχείρισης του μυστικού κλειδιού ομάδας είναι μεταξύ αυτών που παρέχουν αυθεντικοποίηση του μυστικού κλειδιού συνόδου (key authentication) και αυτόν που δεν έχουν αυτή την δυνατότητα.

Ένα πρωτόκολλο παρέχει αυθεντικοποίηση κλειδιού αν εξασφαλίζει ότι αν ένα group M μελών αποφασίσουν την δημιουργία ενός μυστικού κλειδιού, καμία οντότητα που δεν ανήκει στο group των M, δεν θα μπορεί να συμμετάσχει στην δημιουργία του κλειδιού και να έχει γνώση αυτού.

Η αυθεντικοποίηση του κλειδιού μπορεί να επιτευχθεί αν κάθε μέλος του group μοιράζεται ένα μυστικό κλειδί μακράς διάρκειας (long term key) με κάθε άλλο μέλος. Με ενσωμάτωση του long term key μέσα στο πρωτόκολλο διαχείρισης του μυστικού κλειδιού συνόδου του group τα μέλη μπορούν να αυθεντικοποιηθούν αμοιβαία (strong key authentication). Η ιδέα κάθε μέλος να μοιράζεται ένα long term key με κάθε άλλο μέλος με τα οποία θα αυθεντικοποιούνται αμοιβαία έχει σαν αποτέλεσμα την δημιουργία πρωτοκόλλων με πολύ μεγάλο κόστος. Μία παραλλαγή της παραπάνω ιδέας είναι όλα τα μέλη να διαμοιράζονται ένα μυστικό κλειδί μόνο με ένα μέλος, το οποίο θα κάνει τον έλεγχο αυθεντικοποίησης εκ' μέρους όλων (implicit key authentication).

Ένα πρωτόκολλο που παρέχει αυθεντικοποίηση κλειδιού μπορεί να θεωρηθεί και ένα είδος μηχανισμού ελέγχου πρόσβασης.

Τα πρωτόκολλα που δεν παρέχουν αυθεντικοποίηση κλειδιού απαιτούν την χρήση αυθεντικοποιημένων καναλιών επικοινωνίας.

## Ιδιότητες πρωτοκόλλων

Ορισμένες ιδιότητες των πρωτοκόλλων διαχείρισης του μυστικού κλειδιού συνόδου είναι οι ακόλουθες.

- Group Key Secrecy. Η αρχή αυτή εξασφαλίζει ότι είναι υπολογιστικά αδύνατο για έναν παθητικό επίβουλο να ανακαλύψει το μυστικό κλειδί συνόδου.
- Key Independence. Η αρχή αυτή εξασφαλίζει ότι αν ένας παθητικός γνωρίζει κάποια προηγούμενα μυστικά κλειδιά του group δεν μπορεί να υπολογίσει μελλοντικά κλειδιά.
- Forward secrecy – FS. Η αρχή αυτή εξασφαλίζει ότι αν ένας παθητικός επίβουλος έχει γνώση διαδοχικών παρελθόντων κλειδιών ενός group, δεν μπορεί να υπολογίσει μελλοντικά κλειδιά.
- Backward Secrecy – BS. Η αρχή αυτή εξασφαλίζει ότι αν ένας παθητικός επίβουλος έχει γνώση διαδοχικών παρελθόντων κλειδιών ενός group, δεν μπορεί να υπολογίσει άλλα παρελθόντα κλειδιά.
- Perfect Forward Secrecy – PFS. Η αρχή αυτή εξασφαλίζει ότι η αποκάλυψη ενός long term key δεν βάζει σε κίνδυνο την ασφάλεια παρελθόντων μυστικών κλειδιών του group.
- Αντίσταση στις επιθέσεις τύπου know – key. Ένα πρωτόκολλο θεωρείται ευάλωτο στις know key επιθέσεις, αν η αποκάλυψη ενός κλειδιού συνόδου επιτρέπει σε ένα παθητικό εισβολέα (που δεν τροποποιεί τα μηνύματα) να αποκαλύψει και άλλα κλειδιά συνόδου ή επιτρέπει σε ένα ενεργό εισβολέα (που τροποποιεί τα μηνύματα) να υποδυθεί ένα από τα εμπλεκόμενα μέλη του πρωτοκόλλου.

Στην συνέχεια θα δώσουμε μία σύντομη περιγραφή κάποιων κεντρικοποιημένων, μη κεντρικοποιημένων και κατανεμημένων πρωτοκόλλων. Η εστιάσουμε όμως την προσοχή μας στα τελευταία γιατί θεωρούμε ότι είναι τα πλέον κατάλληλα για ένα group communication σύστημα μίας και δεν παραβιάζουν την κατανεμημένη φύση του.

## Κεντρικοποιημένα Πρωτόκολλα

Στα κεντρικοποιημένα πρωτόκολλα όπως ήδη αναφέραμε υπάρχει μόνο μία οντότητα που ελέγχει την διαχείριση του μυστικού κλειδιού συνόδου. Όπως συμβαίνει σε κάθε κεντρικοποιημένο πρωτόκολλο έτσι και στα πρωτόκολλα διαχείρισης του μυστικού κλειδιού συνόδου η ύπαρξη μίας μόνο έμπιστης τρίτης οντότητας επιφορτισμένης με την ευθύνη διαχείρισης του μυστικού κλειδιού δημιουργεί ένα σημείο αποτυχίας. Στην περίπτωση αποτυχίας της οντότητας αυτής η ασφάλεια ολόκληρου του group κινδυνεύει. Επίσης σε πολύ μεγάλα groups η έμπιστη τρίτη οντότητα είναι πιθανό να μην μπορεί να αντιμετωπίσει το φόρτο της διαχείρισης των κλειδιών για ένα πολύ μεγάλο αριθμό μελών. Η επεκτασιμότητα αποτελεί κατά συνέπεια ένα σημαντικό πρόβλημα των κεντρικοποιημένων συστημάτων. Η αποτελεσματικότητα ενός κεντρικοποιημένου πρωτοκόλλου μπορεί να μετρηθεί ανάλογα:

- Με τις απαιτήσεις αποθήκευσης. Δηλαδή των αριθμών κλειδιών που θα χρησιμοποιούνται για κρυπτογράφηση άλλων κλειδιών (Key Encryption Key – KEK)
- Το μέγεθος των μηνυμάτων
- Την ικανοποίηση των αρχών του Forward Secrecy και Backward Secrecy.
- Την αντιμετώπιση επιθέσεων συνομωσίας. Όπου η συνεργασία κάποιου αριθμού μελών που αποχώρησαν από το group δεν θα επιτρέψει την αποκάλυψη του τρέχοντος μυστικού κλειδιού συνόδου του group.

Θα περιγράψουμε στην συνέχεια ορισμένα κεντρικοποιημένα πρωτόκολλα διαχείρισης του μυστικού κλειδιού συνόδου ενός group

### *Group Key Management Protocol*

Το Group Key Management Protocol – GKMP [38] είναι ένα πρωτόκολλο δημιουργίας και διαχείρισης του μυστικού κλειδιού συνόδου ενός group. Σε αυτό το πρωτόκολλο το KDC με την συνεργασία του πρώτου μέλουν που εισέρχεται στο group δημιουργεί ένα πακέτο (Group Key Packet – GKP) που περιέχει ένα κλειδί για την κρυπτογράφηση των μηνυμάτων που θα αποστέλλονται μεταξύ των μελών του group (Group Traffic Encryption Key – GTEK) και ένα κλειδί για την κρυπτογράφηση μελλοντικών κλειδιών (Group Key Encryption Key – GKEK).

Στο GKMP όταν ένα νέο μέλος ζητά πρόσβαση στο group τότε το KDC στέλνει στο υποψήφιο μέλος ένα αντίγραφο του GKP. Στην περίπτωση της αναθεώρησης του

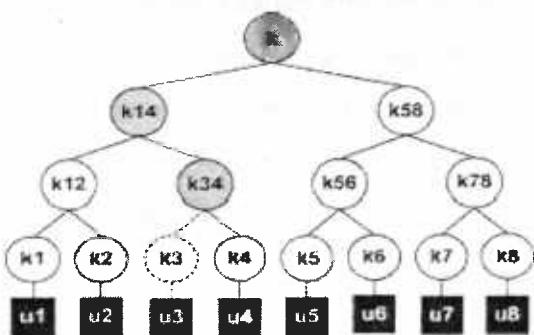
μυστικού κλειδιού συνόδου ο αρχηγός του group (Group Controller – GC) δημιουργεί ένα GKP και το κρυπτογραφεί με το τρέχον GKEK. Όμως επειδή όλα τα μέλη γνωρίζουν το GKEK, δεν μπορεί να εξασφαλιστεί η αρχή του FS όταν ένα μέλος εγκαταλείπει το group. Μια λύση στο πρόβλημα αυτό θα μπορούσε να αποτελέσει η επαναδημιουργία του group χωρίς να συμπεριληφθεί το μέλος που αποχώρησε.

### **Logical Key Hierarchy**

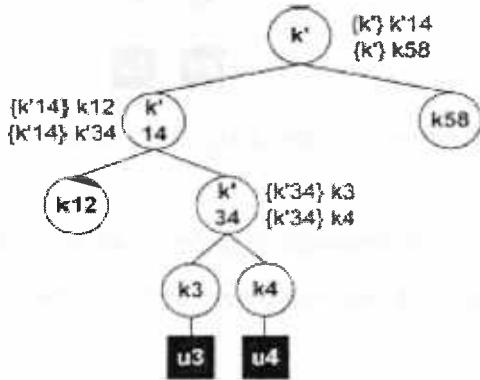
Μια άλλη προσέγγιση [39,40] στην κεντρικοποιημένη διαχείριση του μυστικού κλειδιού συνόδου ενός group, κάνει χρήση ενός KDC και ενός δένδρου κλειδιών, την διαχείριση του οποίου αναλαμβάνει το KDC. Σε κάθε κόμβο του δένδρου αντιστοιχίζεται ένα κλειδί για την κρυπτογράφηση άλλων κλειδιών (Key Encryption Key – KEK). Τα φύλλα του δένδρου αντιστοιχούν στα μέλη του group και σε κάθε φύλλο αντιστοιχίζεται ένα KEK που συσχετίζεται με το μέλος που αντιστοιχεί στο συγκεκριμένο φύλλο. Κάθε μέλος λαμβάνει και διαχειρίζεται ένα αντίγραφο του KEK που αντιστοιχίζεται στο συγκεκριμένο φύλλο και τα KEKs των κόμβων από την ρίζα του δέντρου μέχρι και τον κόμβο πατέρα του φύλλου. Το κλειδί που αντιστοιχεί στην ρίζα του δένδρου είναι το μυστικό κλειδί συνόδου του group.

Στην περίπτωση ενός ζυγισμένου δένδρου ο αριθμός των κλειδιών που το κάθε μέλος αποθηκεύει είναι  $\log_2 n + 1$  κλειδιά. Όπου  $\log_2 n$  είναι το ύψος του δένδρου.

Κατά την είσοδο ενός νέου μέλους αυτό αντιστοιχίζεται σε ένα φύλλο. Όλα τα KEKs από την ρίζα μέχρι τον κόμβο “πατέρα” του νέου μέλους πρέπει να αλλαχθούν προκειμένου να εξασφαλιστεί η αρχή του BS. Δημιουργείται ένα μήνυμα αναθεώρησης κλειδιού που περιέχει κάθε νέο KEK κρυπτογραφημένο με τα κλειδιά των κόμβων παιδιών του. Το μέγιστο μέγεθος του μηνύματος αυτού θα περιέχει  $2\log_2 n$  κλειδιά. Στο σχήμα που ακολουθεί με γκρι χρώμα είναι σημειωμένοι οι κόμβοι των οποίων τα KEKs αλλάζουν με την είσοδο ενός νέου μέλους του u3.



Το νέο μέλος  $u_3$  λαμβάνει ένα μυστικό κλειδί και το φύλλο που του αντιστοιχεί έχει πατέρα τον κόμβο  $k_{34}$ . Τα KEKs των κόμβων  $k_{34}$ ,  $k_{14}$  και  $k$  που είναι οι κόμβοι από τον κόμβο πατέρα στην ρίζα δεσμεύονται για να αντικατασταθούν με νέα KEKs ( $k_{34}'$ ,  $k_{14}'$  και  $k'$ ) όπως φαίνεται στό παρακάτω σχήμα.

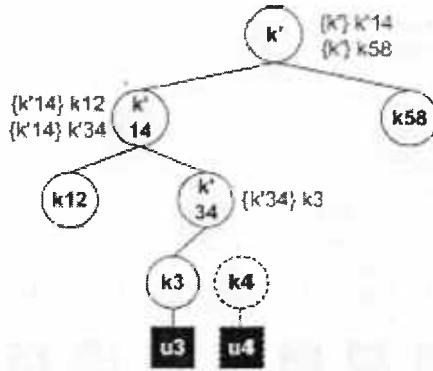


$\{x\}k$  σημαίνει ότι το  $x$  έχει κρυπτογραφηθεί με το  $k$

Τελικά τα KEKs κρυπτογραφούνται με κάθε κλειδί των κόμβων παιδιών του κόμβου που αντιστοιχεί το νέο KEK π.χ.  $\{k_{34}'\}k_3, k_4 ; \{k_{14}'\}k_{12}, k_{34}$  και  $\{k\}k_{14}, k_{58}$ . Το μέγιστο μέγεθος του μηνύματος ανανέωσης του κλειδιού συνόδου είναι  $2(\log_2 n)$ .

Κατά την αποχώρηση ενός μέλους από το group ακολουθείται μία παρόμοια διαδικασία. Όταν ένα μέλος αποχωρεί ή απορρίπτεται από το group τότε το KEK του κόμβου πατέρας του και όλων των κόμβων που ανήκουν στο μονοπάτι από τον πατέρα του στην ρίζα πρέπει να αντικατασταθούν με νέα προκειμένου να εξασφαλιστεί η αρχή του FS. Ένα μήνυμα αλλαγής κλειδιού κατά την αποχώρηση ή αποβολή ενός μέλους περιλαμβάνει τα νέα KEKs κρυπτογραφημένα με τα κλειδιά των κόμβων παιδιών του εκάστοτε κόμβου που αντιστοιχεί το KEK. Εξαίρεση αποτελεί ο πατέρας του κόμβου που αποχωρεί. Το KEK αυτού του κόμβου κρυπτογραφείται μόνο με το κλειδί του παιδιού που απομένει. Το νέο κλειδί συνόδου με τον τρόπο αυτό δεν μπορεί να αποκαλυφθεί μίας και το κλειδί του αποχωρούντος μέλους δεν χρησιμοποιήθηκε στην κρυπτογράφηση κάποιου νέου KEK.

Στο σχήμα που ακολουθεί απεικονίζεται η διαδικασία ανανέωσης κλειδιού κατά την αποχώρηση ή αποβολή κάποιου μέλους.



$\{x\}k$  σημαίνει ότι το  $x$  έχει κρυπτογραφηθεί με το  $k$

Όπως παρατηρούμε στο παραπάνω σχήμα η αποχώρηση του  $u_4$  που είναι γνώστης των KEKs των  $k_{34}$ ,  $k_{14}$  και  $k$  απαιτεί την ανανέωση των τελευταίων ( $k_{34}$ ,  $k_{14}$  και  $k$  αντίστοιχα).

### One-way Function Tree

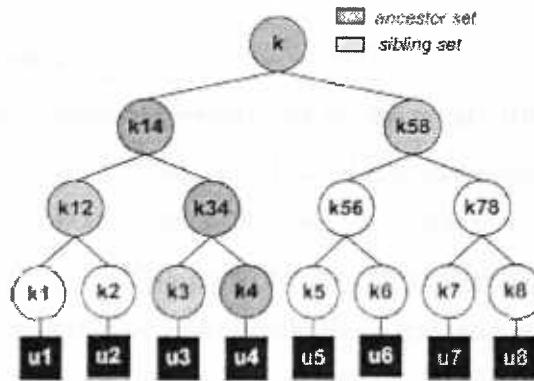
Θα περιγράψουμε ένα πρωτόκολλο που προτάθηκε από τους McGrew και Sherman [41]. Το πρωτόκολλο αυτό κάνει χρήση one-way function δένδρων και μειώνει το μέγεθος του μηνύματος ανανέωσης κλειδιού από  $2\log_2 n$  σε  $\log_2 n$ .

Στο πρωτόκολλο που περιγράφουμε τα KEK των κόμβων δεν αποδίδονται στους κόμβους άλλα οι ίδιοι οι κόμβοι μπορούν να τα υπολογίσουν. Τα KEK των κόμβων παιδιών ενός κόμβου “κρύβονται” (blinding) με την χρήση μίας μονόδρομης συνάρτησης και στην συνέχεια αναμειγνύονται με χρήση μίας κατάλληλης συνάρτησης. Το αποτέλεσμα των παραπάνω υπολογισμών είναι το KEK του κόμβου πατέρα. Οι υπολογισμοί που περιγράψαμε παραπάνω συνοψίζονται στην παρακάτω σχέση.

$$k_i = f(g(k_{\text{left}(i)}), g(k_{\text{right}(i)}))$$

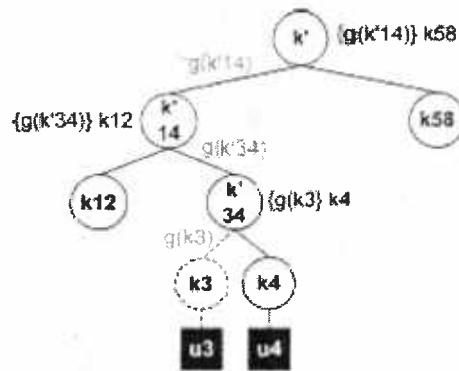
Όπου  $\text{left}(i)$  και  $\text{right}(i)$  είναι το αριστερό και το δεξί παιδί του κόμβου  $i$  ενώ η  $g$  είναι η μονόδρομη συνάρτηση και  $f$  η συνάρτηση μίξης.

Σαν πρόγονοι ενός κόμβου θεωρούνται οι κόμβοι εκείνοι που ανήκουν στο μονοπάτι από τον κόμβο πατέρα μέχρι την ρίζα του δένδρου. Το σύνολο των κόμβων προγόνων ενός κόμβου ονομάζεται ancestor set ενώ το σύνολο των κόμβων που έχουν κοινό πατέρα με αυτούς του ancestor set ονομάζεται sibling set. Για τον κόμβο  $u_4$  το σύνολο τα ancestor και sibling set απεικονίζονται στο παρακάτω σχήμα.



Κάθε μέλος λαμβάνει ένα κλειδί και τα blinding κλειδιά των κόμβων που ανήκουν στο sibling set του. Σε ένα ισορροπημένο δένδρο ο αριθμός των κλειδιών που αποθηκεύει ένα μέλος είναι  $\log_2 n + 1$ . Για παράδειγμα το μέλον  $u_4$  στο παραπάνω σχήμα γνωρίζει το κλειδί  $k_4$  και τα blinding κλειδιά των κόμβων  $k_3$ ,  $k_{12}$  και  $k_{58}$ . Με την γνώση αυτών των κλειδιών και της σχέσης  $k_i = f(g(k_{left(i)}), g(k_{right(i)}))$  το μέλος  $u_4$  μπορεί να παράγει τα κλειδιά των κόμβων που ανήκουν στο ancestor set του.

Κατά την ανανέωση του μυστικού κλειδιού το νέο κλειδί πρέπει να κρυπτογραφηθεί μόνο με το κλειδί του sibling κόμβου. Ένα παράδειγμα ανανέωσης κλειδιού φαίνεται στο παρακάτω σχήμα.



$\{x\}_k$  σημαίνει ότι το  $x$  έχει κρυπτογραφηθεί με το  $k$   
 $g(x)$  σημαίνει ότι πάνω στο  $x$  έγινε σύνοψη με χρήση της συνάρτησης  $g$

Όταν το μέλος  $u_3$  εισέρχεται στο group, απαιτείται η αναθεώρηση των κλειδιών  $k_{34}$ ,  $k_{14}$  και  $k$ . Η μόνες τιμές που πρέπει να διανεμηθούν είναι τα blinding κλειδιά των  $k_3$ ,  $k_{34}$  και  $k_{14}$  που κρυπτογραφούνται αντίστοιχα με τα  $k_4$ ,  $k_{12}$  και  $k_{58}$ . Τα νέα KEKs μπορούν να υπολογιστούν από κάθε μέλος του group με τον ακόλουθο τρόπο:

$$k_{34} = f(g(k_3), g(k_4)) \quad k_{14} = f(g(k_{12}), g(k_{34})) \quad k = f(g(k_{14}), g(k_{58}))$$

## One-way Function Chain Tree

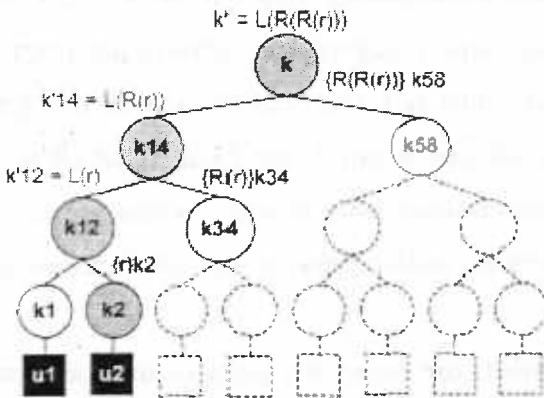
Ένα ελαφρός τροποποιημένο πρωτόκολλο σε σχέση με αυτό που περιγράψαμε παραπάνω προτάθηκε από τον Chanetti κ.α. [42]. Το πρωτόκολλο αυτό επιτυγχάνει το ίδιο επικοινωνιακό φόρτο αλλά κάνει χρήση μίας γεννήτριας ψευδοτυχαίων αριθμών για την δημιουργία των νέων KEKs στην περίπτωση αποχώρησης ενός μέλους αντί της μονόδρομης συνάρτησης που χρησιμοποιήθηκε στο παραπάνω πρωτόκολλο. Η γεννήτρια ψευδοτυχαίων αριθμών  $G(x)$  διπλασιάζει το μήκος της συμβολοσειράς εισόδου  $x$  ενώ υπάρχουν και δύο συναρτήσεις  $L(x)$  και  $R(x)$  που αντιπροσωπεύουν το αριστερό και δεξί μισό της  $G(x)$  αντίστοιχα.

Όταν ένα μέλος έστω  $u$  αποχωρεί από το group τα βήματα του αλγόριθμου ανανέωσης του μυστικού κλειδιού του group είναι τα ακόλουθα.

1. Μία νέα τιμή  $r_v$  αντιστοιχίζεται σε κάθε κόμβο  $v$  που ανήκει στο μονοπάτι από τον κόμβο πατέρα του μέλους που αποχώρισε μέχρι την ρίζα του δένδρου.
2. Τα νέα κλειδιά παράγονται από την σχέση  $k_v = L(r_v)$
3. Κάθε  $r_{p(v)}$  κρυπτογραφείται με το κλειδί  $k_{s(v)}$  όπου  $s(v)$  είναι το sibling κλειδί του  $v$  και αποστέλλεται.

Από το  $r_v$  μπορεί κάποιος να υπολογίσει όλα τα κλειδιά  $k_v$ ,  $k_{p(v)}$ ,  $k_{p(p(v))}$

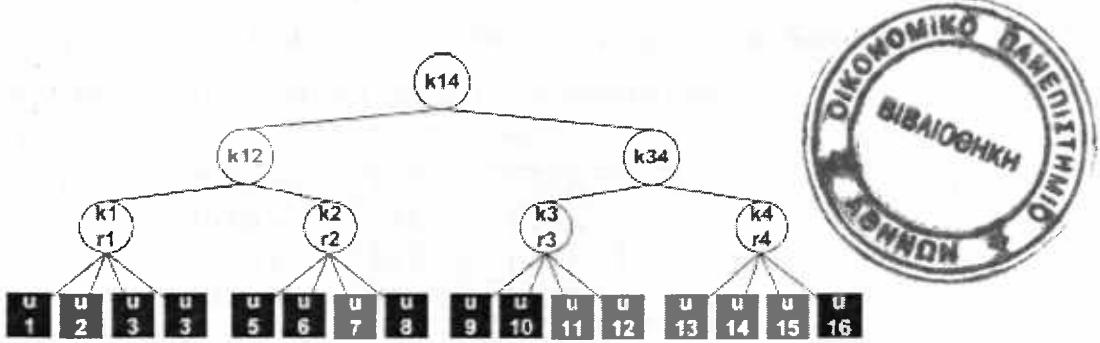
Στο σχήμα που ακολουθεί παρουσιάζεται το σενάριο αποχώρησης του μέλους  $u_1$ .



Σύμφωνα με τον αλγόριθμο που περιγράψαμε παραπάνω με την αποχώρηση του  $u_1$  στους κόμβους  $n_{12}$ ,  $n_{14}$  και  $n_0$  θα οι τιμές  $r$ ,  $R(r)$  και  $R(R(r))$  οι οποίες θα κρυπτογραφηθούν με τα αντίστοιχα KEKs  $k_2$ ,  $k_{34}$  και  $k_{58}$ . Τελικά τα νέα KEKs  $k_{12}$ ,  $k_{14}$  και  $k$  θα είναι τα  $L(r)$ ,  $L(R(r))$  και  $L(R(R(r)))$ .

## Hierarchical a-ary Tree with Clustering

Ένα άλλο κεντρικοποιημένο πρωτόκολλο διαχείρισης του μυστικού κλειδιού ενός group προτάθηκε από τον Canetti [43]. Το πρωτόκολλο αυτό κάνει χρήση a-ary δένδρων [44]. Η λογική του στηρίζεται στο σπάσιμο ενός group  $n$  μελών σε υπο-groups (clusters) των  $m$  των  $m$  μελών. Κάθε cluster θα αντιστοιχίζεται σε ένα φύλλο του δένδρου. Ο αριθμός των clusters που θα δημιουργούνται με αυτό τον τρόπο θα είναι  $n/m$  και το δένδρο θα έχει ύψος  $\log_a(n/m)$ , όπως φαίνεται και στο παρακάτω σχήμα.



Παράδειγμα δένδρου με τέσσερα clusters

Όλα τα μέλη που ανήκουν στο ίδιο cluster θα διαμοιράζονται το ίδιο KEK ενώ σε κάθε μέλος του cluster θα αντιστοιχίζεται ένα μοναδικό κλειδί  $k_i$  το οποίο θα γνωρίζει μόνο το εκάστοτε μέλος και το KDC. Το KDC χρησιμοποιεί ένα τυχαίο αριθμό  $r$  σαν δείκτη για μία συνάρτηση παραγωγής ψευδοτυχαίων αριθμών  $f_r$  προκειμένου να παράγει το κλειδί  $k_i$  π.χ. για το μέλος  $i$   $k_i = f_r(i)$ . Για κάθε cluster αποθηκεύεται ο τυχαίος αριθμός  $r$  και το KEK. Τα μέλη που ανήκουν στο ίδιο cluster γνωρίζουν το σύνολο των κλειδιών από τον κόμβο – φύλλο στον οποίο συνδέονται μέχρι την ρίζα του δένδρου κάτι που συνεπάγεται ότι το κάθε μέλος αποθηκεύει  $\log_a(n/m) + 1$  KEKs.

Όταν ένα μέλος αποχωρεί από το group στέλνεται στο cluster στο οποίο το μέλος άνηκε ένα νέο KEK κρυπτογραφημένο με τα KEKs των μελών που παρέμειναν στο cluster. Κατά συνέπεια η αποχώρηση ενός μέλους από το group απαιτεί  $m-1$  κρυπτογραφήσεις. Επιπλέον το KDC ανανεώνει όλα τα κλειδιά που περιέχονται στο μονοπάτι από το φύλλο του cluster μέχρι την ρίζα του δένδρου και κάθε νέο KEK κρυπτογραφείται με τα KEK των κόμβων παιδιών του εκάστοτε κόμβου που το KEK ανανεώνεται.

## Centralized Flat Table

Το πρωτόκολλο αυτό προτάθηκε από τον Waldvogel κ.α. [45] και κάνει χρήση ενός πίνακα κλειδιών σε αντίθεση με τα προηγούμενα πρωτόκολλα που χρησιμοποιούσαν μία δενδρική μορφή για την αντιστοίχιση κλειδιών στα μέλη κάθε group. Η χρήση του πίνακα είχε σκοπό να μειωθεί ο αριθμός των κλειδιών που το KDC πρέπει να διαχειρίζεται. Ο πίνακας περιέχει μία έγγραφή για κάθε TEK και 2w έγγραφές για τα KEKs όπου w ο αριθμός των bits στο ID κάθε μέλους. Ως γνωστόν κάθε bit μπορεί να πάρει τιμή 0 ή 1. Σε κάθε τιμή του bit αντιστοιχίζεται ένα KEK. Ένα μέλος γνωρίζει μόνο τα KEK που στο πίνακα αντιστοιχούν στην δική του τιμή bit. Το παράδειγμα ενός τέτοιου πίνακα δίνεται στο παρακάτω σχήμα.

TEK			
ID Bit #0	KEK 0.0	KEK 0.1	
ID Bit #1	KEK 1.0	KEK 1.1	
ID Bit #2	KEK 2.0	KEK 2.1	
ID Bit #3	KEK 3.0	KEK 3.1	

Bit 0                      Bit 1

Αν ένας χρήστης έχει ID 0101 τότε σύμφωνα με τον παραπάνω πίνακα θα γνωρίζει τα KEKs: KEK0.0, KEK1.1, KEK2.0, KEK3.1

Όταν ένα μέλος εγκαταλείψει το group όλα τα KEKs που αυτό γνωρίζει αντικαθίστανται με νέα και το KDC στέλνει ένα μήνυμα. Το περιεχόμενο του μηνύματος απαρτίζεται από δύο μέλη.

- Το πρώτο μέλος περιέχει το νέο TEK κρυπτογραφημένο με κάθε KEK που δεν έχει επηρεαστεί από την αποχώρηση του μέλους
- Στο δεύτερο μέλος κάθε νέο KEK κρυπτογραφείται με το παλιό KEK και με το νέο TEK. Με τον τρόπο αυτό τα εναπομείναντα μέλη μπορούν να ανανεώσουν τα παλιά KEK χωρίς επιπρόσθετες πληροφορίες για τα KEK των άλλων μελών. Το μήνυμα για τον αποκλεισμό του μέλους με ID 0101 απεικονίζεται στον παρακάτω πίνακα.

TEK			
(KEK 0.0 <sub>new</sub> ) TEK <sub>new</sub>	(TEK <sub>new</sub> ) KEK 0.1		ID Bit #0
(TEK <sub>new</sub> ) KEK 1.0	(KEK 1.1 <sub>new</sub> ) TEK <sub>new</sub>		ID Bit #1
(KEK 2.0 <sub>new</sub> ) TEK <sub>new</sub>	(TEK <sub>new</sub> ) KEK 2.1		ID Bit #2
(TEK <sub>new</sub> ) KEK 3.0	(KEK 3.1 <sub>new</sub> ) TEK <sub>new</sub>		ID Bit #3

Bit 0                      Bit 1

## *Efficient Large-Group Key*

Ο Petrig κ.α. πρότεινε ένα πρωτόκολλο (ELK) [46] για αποτελεσματική διαχείριση του μυστικού κλειδιού ενός group. Το πρωτόκολλο αυτό έχει πολλά κοινά με το One-way Function Tree (OFT) πρωτόκολλο αφού για τον υπολογισμό του κλειδιού του κόμβου πατέρα συμμετέχουν τα κλειδιά των παιδιών του. Το ELK κάνει χρήση ψευδοτυχαίων συναρτήσεων (PRFs) προκειμένου να δημιουργήσει και να διαχειριστεί τα κλειδιά που θα περιέχονται σε ένα ιεραρχικό δένδρο. Μία PRF χρησιμοποιεί σαν είσοδο μία συμβολοσειρά  $M$  μήκους  $m$  για να παράγει μία έξοδο μήκους  $n$ . Η παραπάνω διαδικασία απεικονίζεται συμβολικά με τον ακόλουθο τρόπο  $\text{PRF}_k^{m \rightarrow n}(M)$ . Χρησιμοποιώντας την PRF σε ένα κλειδί μπορεί να προκύψουν έως τέσσερα διαφορετικά κλειδιά που μπορούν να χρησιμοποιηθούν για διαφορετικές λειτουργίες.

Στο ELK η ανανέωση του μυστικού κλειδιού γίνεται περιοδικά σε τακτά χρονικά διαστήματα. Το κλειδί του group ανανεώνεται κάνοντας χρήση της σχέσης  $k'_G = \text{PRF}_{k'_i}^{n \rightarrow n}(0)$  και όλα τα άλλα κλειδιά  $k'_i$  προκύπτουν από την σχέση  $k'_i = \text{PRF}_{k'_i}^{n \rightarrow n}(k'_G)$ . Με την παραγωγή όλων των κλειδιών το ELK δεν απαιτεί την αποστολή multicast μηνυμάτων κατά την εισαγωγή ενός νέου μέλους αλλά χρειάζεται η αποστολή unicast μηνυμάτων όταν υπάρχουν μετακινήσεις μελών στο δένδρο του group εξαιτίας της εισαγωγής κάποιου νέου μέλους.

Κατά την αποβολή μελών από το group απαιτείται η ανανέωση των κλειδιών στο μονοπάτι από το κόμβο του μέλους που αποχώρισε προς την ρίζα. Αυτό επιτυγχάνεται με την παραγωγή των νέων κλειδιών  $k'_i$  από την σχέση  $k'_i = \text{PRF}_{C_{LR}}^{n \rightarrow n}(k_i)$  όπου  $C_{LR} = \text{PRF}_{k'_{il}}^{n \rightarrow n_1}(k_i) | \text{PRF}_{k'_{ir}}^{n \rightarrow n_2}(k_i)$  όπου  $k_{il}$  και  $k_{ir}$  είναι το αριστερό και δεξί παιδί αντίστοιχα του κόμβου  $i$ . Η διαδικασία ανανέωσης του κλειδιού μετά την αποχώρηση κάποιου μέλους κλείνει με την πολλαπλή αποστολή (multicast) των  $\{\text{PRF}_{k'_{il}}^{n \rightarrow n_1}(k_i)\}_{k'_{il} \in \beta}$  και  $\{\text{PRF}_{k'_{ir}}^{n \rightarrow n_2}(k_i)\}_{k'_{ir} \in \beta}$  από τον server.

Τέλος το ELK εισάγει την ιδέα χρήσης μικρών μηνυμάτων (hints) που θα χρησιμοποιούνται στην περίπτωση απώλειας κάποιων μηνυμάτων ανανέωσης κλειδιού. Τα hints βελτιώνουν την αξιοπιστία και την λειτουργικότητα του

μηχανισμού αναθεώρησης κλειδίου. Κάθε κλειδί  $k_i$  δημιουργείται από τα  $n_1$  bits του αριστερού παιδιού του κόμβου  $i$  και τα  $n_2$  του δεξιού παιδιού. Η χρήση μίας επιβεβαίωσης της τιμής του νέου κλειδιού, που παράγεται από το νέο κλειδί με χρήση της σχέσης  $V_{k_i} = PRF_{k_i}(0)$  επιτρέπει στα παιδιά του κόμβου  $i$  να ανακτήσουν το κλειδί  $k_i$  ακόμα και στην περίπτωση που δεν γνωρίζουν την συνεισφορά των bits του αδελφού τους δοκιμάζοντας όλους τους πιθανούς συνδυασμούς.

## Μη κεντρικοποιημένα πρωτόκολλα

Στα μη κεντρικοποιημένα πρωτόκολλα τα μεγάλα group σπάνε σε upo – groups καθένα από τα οποία έχει την δική του κεντρική διαχείριση. Με αυτό τον τρόπο ελαχιστοποιείται το πρόβλημα συσσώρευσης όλων των λειτουργιών σε μία μόνο οντότητα ενώ η αποτυχία ορισμένων οντοτήτων που έχουν αναλάβει τον ρόλο της διαχείρισης του μυστικού κλειδιού συνόδου του group δεν συνεπάγεται αποτυχία σε ολόκληρο το group.

Πριν μπούμε στην περιγραφή πρωτοκόλλων που υλοποιούν μη κεντρικοποιημένες αρχιτεκτονικές θα αναφέρουμε τα γενικά χαρακτηριστικά αυτών των πρωτοκόλλων που είναι τα ακόλουθα.

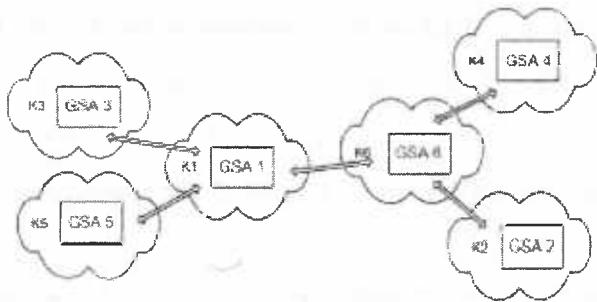
- Ανεξαρτησία κλειδιού (Key Independence). Η αποκάλυψη ενός κλειδιού δεν πρέπει να θέτει σε κίνδυνο προηγούμενα κλειδιά.
- Μη κεντρικοποιημένος ρυθμιστής (Decentralized controller). Ο διαχειριστής του group, σκοπός του οποίου είναι ο συντονισμός των διαχειριστών των upo – groups δεν πρέπει να είναι κάποια κεντρικοποιημένη οντότητα. Η χρήση μίας κεντρικοποιημένης οντότητας σαν συντονιστή του group συνεπάγεται τα μειονεκτήματα των κεντρικοποιημένων πρωτοκόλλων, μίας και αποτυχία της οντότητας αυτής θα σήμαινε και αποτυχία ολόκληρου του group.
- Τοπική ανανέωση κλειδιού. Η ανανέωση του μυστικού κλειδιού σε κάθε upo – group εξαιτίας της αποχώρησης ή της εισόδου κάποιου μέλους σε αυτό θα πρέπει να αντιμετωπίζεται τοπικά .
- Το μονοπάτι αποστολής δεδομένων πρέπει να είναι ανεξάρτητο από αυτό για την διαχείριση της διαδικασίας ανανέωσης κλειδιού έτσι ώστε στην περίπτωση ανανέωσης του μυστικού κλειδιού σε κάποιο upo – group να μην παρατηρείται καθυστέρηση στην αποστολή δεδομένων.

➤ Ανανέωση του κλειδιού του υπο – group με την αποχώρηση ή την είσοδο μελών, ώστε να ικανοποιούνται οι αρχές του FS και BS.

Στην συνέχεια ακολουθεί μία σύντομη περιγραφή κάποιων πρωτοκόλλων που στηρίζονται σε μη κεντρικοποιημένες αρχιτεκτονικές.

### Iolus

Ένα μη κεντρικοποιημένο πρωτόκολλο προτάθηκε από τον Mittra [47]. Στο πρωτόκολλο αυτό γίνεται χρήση μίας ιεραρχίας πρακτόρων (agents) που σπάνε το group σε υπο – groups. Ένας πράκτορας ασφάλειας του group (Group Security Agent – GSA) διαχειρίζεται κάθε υπο – group. Το σύνολο των GSAs θεωρείται σαν ένα group το οποίο έχει τον δικό του GSA. Στο παρακάτω σχήμα απεικονίζεται ο τρόπος με τον οποίο ιεραρχούνται οι GSAs στο πρωτόκολλο που αναλύουμε.



Στο Iolus υπάρχει ανεξαρτησία μεταξύ των κλειδιών σε κάθε υπο – group ενώ δεν υπάρχει ένα κοινό κλειδί για ολόκληρο το group. Οι ιδιότητες αυτές επιτρέπουν μία αλλαγή στο membership (είσοδος ή αποχώρηση κάποιου μέλους) να χειρίζεται τοπικά σε κάθε υπο – group. Επιπλέον η έλλειψη κεντρικής διαχείρισης εξασφαλίζει στο πρωτόκολλο μεγαλύτερη ανοχή σε λάθη μίας και αποτυχία κάποιου GSA επηρεάζει μόνο το υπο – group στο οποίο ο GSA ανήκει.

Ένα βασικό μειονέκτημα του Iolus είναι ότι δεν εξασφαλίζει ανεξαρτησία των μονοπατιών δεδομένων. Όταν τα δεδομένα μεταφέρονται από ένα υπο – group σε ένα άλλο απαιτείται η μετάφραση τους μίας και τα δύο υπο – groups χρησιμοποιούν διαφορετικά κλειδιά. Αυτό σε συνδυασμό με το ότι ο GSA πρέπει να διαχειρίζεται ταυτόχρονα και το κλειδί του υπο – group στο οποίο ανήκει δημιουργεί προβλήματα καθυστέρησης στην μετάδοση των μηνυμάτων.

### Dual-Encryption Protocol

Το Dual-Encryption Protocol (DEP) [48] προτάθηκε από τους Donteti κ.α. με σκοπό να δώσει λύση στο πρόβλημα εμπιστοσύνης σε μία μόνο έμπιστη τρίτη

οντότητα. Στο DEP το group χωρίζεται σε υπο – groups που οργανώνονται iεραρχικά, κάθε υπο – group έχει τον δικό του διαχειριστή (Subgroup Manager – SGM). Σε αυτό το πρωτόκολλο γίνεται χρήση τριών διαφορετικών τύπων κλειδιών για κρυπτογράφηση άλλων κλειδιών (KEKs) και ένα κλειδί για κρυπτογράφηση των δεδομένων (DEK). Ο πρώτος τύπος KEK, KEK<sub>i1</sub> διαμοιράζεται μεταξύ του SGM<sub>i</sub> και των μελών του υπο – group. Το KEK<sub>i2</sub> διαμοιράζεται μεταξύ του διαχειριστή ολόκληρου του group (Group Controller – GC) και των μελών του υπο – group i εκτός του SGM<sub>i</sub>. Τέλος το KEK<sub>i3</sub> διαμοιράζεται μεταξύ του GC και του SGM<sub>i</sub>.

Για την διανομή του DEK στα μέλη του group, ο GC δημιουργεί και διανέμει ένα πακέτο το οποίο περιέχει το DEK κρυπτογραφημένο με το KEK<sub>i2</sub> και κρυπτογραφημένο επίσης με το KEK<sub>i3</sub>. Με την λήψη του πακέτου ο SGM<sub>i</sub> αποκρυπτογραφεί το μέρος του μηνύματος που τον αντιστοιχεί με το KEK<sub>i3</sub> και ανακτά το DEK που είναι κρυπτογραφημένο με το KEK<sub>i2</sub>. Στην συνέχεια ο SGM<sub>i</sub> κρυπτογραφεί με το KEK<sub>i1</sub> το κρυπτογραφημένο με το KEK<sub>i2</sub> DEK και το αποστέλει στα μέλη του υπο – group του. Τέλος κάθε μέλος αποκρυπτογραφεί το πακέτο που έλαβε πρώτα με το KEK<sub>i1</sub> και στην συνέχεια με το KEK<sub>i3</sub> και ανακτά το μυστικό κλειδί DEK.

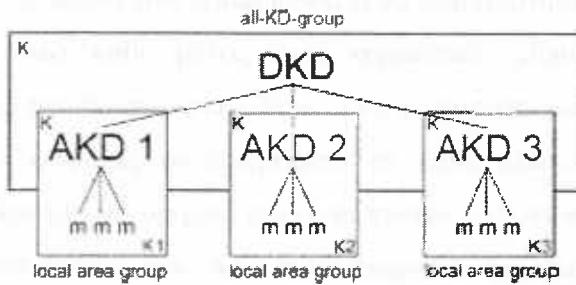
Με την αρχιτεκτονική που περιγράψαμε παραπάνω το DEK δεν μπορεί να ανακτηθεί από κάποια οντότητα που δεν γνωρίζει και τα δύο κλειδιά. Όταν η αποχώρηση ή η είσοδος κάποιου μέλους προκαλέσει αλλαγή στο membership του υπο – group i τότε ο SGM<sub>i</sub> αλλάζει το KEK<sub>i1</sub> και το στέλνει στα μέλη του group όπως έχουν διαμορφωθεί μετά τις αλλαγές.

Ένα μειονέκτημα του DEP είναι ότι τα μέλη που αποχώρησαν από το group μπορούν να λαμβάνουν πληροφορίες από το group ακόμα και μετά την αποχώρηση τους αφού το DEK δεν αλλάζει με την αλλαγή του KEK<sub>i1</sub> κάτι που παραβιάζει την αρχή του PFS.

### ***Intra-Domain Group Key Management***

To Intra-Domain Group Key Management (IGKMP) [49] προτάθηκε από τους DeCleene κ.α.. Στο πρωτόκολλο αυτό υπάρχει ένας κεντρικός διανομέας κλειδιών (Domain Key Distributor – DKD) και τοπικοί διανομείς κλειδιών (Area Key Distributors – AKD) που είναι υπεύθυνοι για μία περιοχή. Ο DKD δημιουργεί το κλειδί του group, οι AKDs διαδίνουν στην συνέχεια το κλειδί στα μέλη του group. Οι

διαχειριστές των κλειδιών (DKD και AKDs) τοποθετούνται σε ένα group που ονομάζεται All – KD – group όπως φαίνεται και στο παρακάτω σχήμα.



Το All-KD-group χρησιμοποιείται από τον DKD για την μετάδοση του κλειδιού στους AKDs. Όλες οι περιοχές στο group χρησιμοποιούν το ίδιο μυστικό κλειδί με αποτέλεσμα να μην απαιτείται μετάφραση των πακέτων όταν περνούν από το ένα upo – group στο άλλο. Επιπλέον ο DKD δεν απαιτείται να γνωρίζει όλα τα μέλη που είναι συνδεδεμένα στο group άλλα μόνο τους AKDs κάθε upo – group.

Ένα βασικό μειονέκτημα του IGKMP είναι ότι η χρήση μίας κεντρικής οντότητας (DKD) δημιουργεί στο πρωτόκολλο αδυναμίες όμοιες με αυτές των κεντρικοποιημένων πρωτοκόλλων, μίας και αποτυχία του DKD οδηγεί σε αποτυχία ολόκληρου του group.

### *Hydra*

Ο Rafaeli και Hutchison πρότειναν το πρωτόκολλο Hydra [50]. Το Hydra σπάει μεγάλα group σε μικρότερα upo – groups, ένας server που στο πρωτόκολλο καλείται σαν Hydra Server – HS διαχειρίζεται κάθε upo – group.

Το Hydra είναι ένα μη κεντρικοποιημένο πρωτόκολλο διαχείρισης του μυστικού κλειδιού συνόδου ενός group που δεν απαιτεί την ύπαρξη ενός κεντρικού διαχειριστή. Όταν συμβεί μία αλλαγή στο membership ενός upo – group έστω i τότε ο HS; πρέπει να δημιουργήσει ένα νέο κλειδί και να το διανείμει και στους υπόλοιπους HSs του group. Στην περίπτωση που ένας ή περισσότεροι HSs είναι μη διαθέσιμοι δεν δημιουργείται κάποιο πρόβλημα στο group μίας και το πρωτόκολλο υλοποιεί ένα μηχανισμό διανομής κλειδιού (Synchronized Group Key Distribution Protocol – SGKDP). Το SGKDP εγγυάται ότι μόνο ένας έγκυρος HS δημιουργεί κάθε φορά το νέο κλειδί.

## Κατανεμημένα πρωτόκολλα

Τα κατανεμημένα πρωτόκολλα ακολουθούν αρχιτεκτονικές που δεν απαιτούν κεντρική διαχείριση. Το κλειδί του group δύναται να δημιουργηθεί είτε συνεργατικά με την συνεισφορά από κάθε μέλος ενός κομματιού (share) και κατάλληλο συνδυασμό των shares που θα συγκεντρωθούν ή να το παράγει ένα μέλος του group. Γενικά δεν θεωρείται ασφαλές να επιτρέπεται σε κάθε μέλος να δημιουργεί το μυστικό κλειδί συνόδου του group μίας και είναι πιθανό κάποια από τα μέλη να μην ικανοποιούν τις προϋποθέσεις για μία τέτοια ενέργεια, για παράδειγμα τα μέλη μπορεί να μην διαθέτουν κάποια κατάλληλη γεννήτρια παραγωγής ψευδοτυχαίων αριθμών. Επιπλέον στα περισσότερα συνεργατικά πρωτόκολλα, με εξαίρεση αυτά που στηρίζονται σε δένδρα (tree based approaches), το υπολογιστικό και επικοινωνιακό κόστος είναι γραμμική συνάρτηση του αριθμού των μελών κάτι που τα κάνει ακατάλληλα για μεγάλα groups. Ένα άλλο εμπόδιο στην χρήση συνεργατικών πρωτοκόλλων είναι ότι απαιτείται από τα μέλη γνώση του membership προκειμένου να είναι αποτελεσματικά.

Μερικά χαρακτηριστικά των κατανεμημένων πρωτοκόλλων που μπορούν να χρησιμοποιηθούν σαν μέτρο της αποτελεσματικότητάς τους είναι τα ακόλουθα:

- Αριθμός των γύρων (Number of rounds). Το πρωτόκολλο πρέπει να προσπαθεί να ελαχιστοποιήσει των αριθμό των επαναλήψεων μεταξύ των μελών προκειμένου να ελαχιστοποιήσει το υπολογιστικό και επικοινωνιακό του κόστος
- Αριθμός των μηνυμάτων (Number of messages). Η καθυστέρηση που δημιουργεί η ανταλλαγή ενός μηνύματος μεταξύ των μελών του group γίνεται ανυπόφορη όσο το group μεγαλώνει
- Υπολογισμοί που απαιτούνται κατά την εγκαθίδρυση (Processing during setup). Οι υπολογισμοί που απαιτούνται κατά την φάση εγκαθίδρυσης ενός νέου group αποτελούν το μεγαλύτερο μέρος του συνολικού κόστους μίας και απαιτείται όλα τα μέλη να επικοινωνήσουν μεταξύ τους.
- Χρήση του Diffie – Helman πρωτοκόλλου. Η χρήση του D-H πρωτοκόλλου σηματοδοτεί ότι έχει επιλεγεί μία συνεργατική αρχιτεκτονική δημιουργίας του κλειδιού συνόδου.

Στην συνέχεια θα δώσουμε την περιγραφή κάποιων κατανεμημένων πρωτοκόλλων διαχείρισης του μυστικού κλειδιού συνόδου ενός group.

## *Secure key agreement for group communications*

Ένα κατανεμημένο πρωτόκολλο διαχείρισης του κλειδιού συνόδου ενός group προτάθηκε από τους Wen-Her Yang και Shiuh-Pyng Shieh [51]. Το πρωτόκολλο αυτό υιοθετεί μία ID – based αρχιτεκτονική για την αμοιβαία αυθεντικοποίηση και εγκατάσταση του κλειδιού συνόδου μεταξύ των μελών του group. Μία αναλυτική περιγραφή της αρχιτεκτονικής αμοιβαίας αυθεντικοποίησης και συμφωνίας ενός μυστικού κλειδιού μεταξύ δύο οντοτήτων δώσαμε παραπάνω στην ενότητα της αυθεντικοποίησης (Πρωτόκολλα αυθεντικοποίησης που στηρίζονται στην ταυτότητα του χρήστη – Σχήμα 2<sup>o</sup>). Στην ενότητα αυτή θα περιγράψουμε συνοπτικά την διαδικασία αυθεντικοποίησης για να κατανοήσουμε πώς αυτή ενοποιείται με το πρωτόκολλο εγκατάστασης και διαχείρισης του μυστικού κλειδιού συνόδου σε ένα group communication σύστημα.

Στο πρωτόκολλο που περιγράφουμε δεν υπάρχει απαίτηση για την ύπαρξη μίας on – line έμπιστης τρίτης οντότητας και επιπλέον δεν απαιτείται κάποιο μέλος του group να επιτελεί τον ρόλο του διαχειριστή του group (group controller). Τέλος πρέπει να επισημάνουμε ότι το πρωτόκολλο ικανοποιεί τις αρχές του PFS και BS. Το πρωτόκολλο χωρίζεται σε τέσσερις φάσεις κάθε μία από τις οποίες θα περιγραφεί αναλυτικά στην συνέχεια:

1. Αρχικοποίηση του κλειδιού (Key Initiation)
2. Δημιουργία του group (Group Creation)
3. Είσοδος μέλουνς (Member Join)
4. Αποχώρηση μέλουνς (Member Departure)

### **Key Initiation**

Στην φάση αρχικοποίησης απαιτείται η ύπαρξη μίας έμπιστης τρίτης οντότητας (Key Information Center – KIC). Σκοπός του KIC είναι η δημιουργία της δημόσιας και μυστικής πληροφορίας που πρέπει να έχει ένα νέο μέλος προκειμένου να δικαιούται την είσοδό του στο group. Από την στιγμή που όλα τα υποψήφια μέλη του group εγγραφούν η ύπαρξη του KIC δεν είναι απαραίτητη. Το KIC θα ενεργοποιηθεί ξανά όταν θα υπάρξει κάποιο αίτημα εγγραφής νέου υποψήφιου μέλουνς. Η διαδικασία εγγραφής ξεκινά με την αποστολή του ID του υποψήφιου μέλουνς στο KIC. Η διαδικασία περιλαμβάνει τα παρακάτω βήματα.

1. Το KIC διαλέγει δύο μεγάλους πρώτους αριθμούς  $p$ ,  $q$  και υπολογίζει τον  $n = p^*q$ .

2. Το KIC ανακτά την μυστική πληροφορία  $d$  την οποίο γνωρίζει το KIC και μόνον αυτό μέσο του υπολογισμού

$$3*d \pmod{(p-1)(q-1)} = 1$$

3. Το KIC υπολογίζει έναν ακέραιο  $g$  που είναι ένα πρωταρχικό στοιχείο και στις δύο  $GF(p)$  και  $GF(q)$ . Όπου  $g$  είναι η δημόσια πληροφορία του KIC.

4. Έστω ότι  $ID_i$  είναι η ταυτότητα του χρήστη  $i$

5. Το KIC επιλέγει μία συνάρτηση σύνοψης και υπολογίζει την εκτεταμένη ταυτότητα (Extended identity –  $EID_i$ ) με τον ακόλουθο τρόπο.

$$EID_i \equiv f(ID_i) \pmod{2^N} \equiv (EID_{i1}, EID_{i2}, \dots, EID_{iN})$$

όπου  $N$  το μήκος σε bit του EID.

6. Μετά τον υπολογισμό του  $EID_i$  υπολογίζεται από τον KIC η μυστική πληροφορία  $S_i$  που αντιστοιχεί στο χρήστη  $i$  ως εξής:

$$S_i \equiv EID_i^d \pmod{n}$$

7. Η initial φάση ολοκληρώνεται με την αποστολή των  $(n, g, f(x), S_i)$  πίσω στον χρήστη  $i$  μέσω ενός ασφαλούς καναλιού. Με την λήψη της παραπάνω πληροφορίας ο χρήστης  $i$  πρέπει να κρατήσει μυστικό το  $S_i$  και να αποθηκεύσει την δημόσια πληροφορία  $(n, g, f(x))$ .

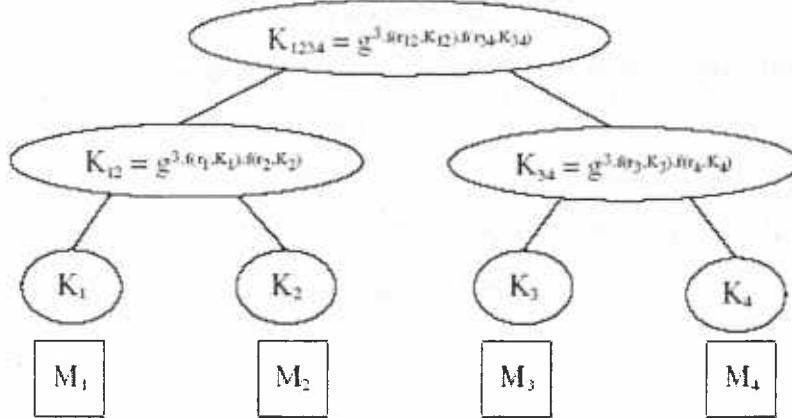
Όταν ένας νέος χρήστης ζητά διαπιστευτήριο για την είσοδο του στο σύστημα το KIC ενεργοποιείται ξανά και τα βήματα 5 – 7 της παραπάνω διαδικασίας επαναλαμβάνονται.

## Group Creation

Σε αυτή την φάση ο δημιουργός του group (Group Creator – GC) καταρτίζει αρχικά την λίστα με τα μέλη που επιθυμούν την είσοδο τους στο group από την φάση δημιουργίας του. Στην συνέχεια ο GC στέλνει σε όλα τα υποψήφια μέλη που περιέχει η λίστα ένα μήνυμα – πρόσκληση και περιμένει για αιτήσεις σύνδεσης από τα μέλη αυτά.

Με την λήψη των μηνυμάτων απάντησης από τα μέλη ο GC ανακοινώνει την έναρξη λειτουργίας του group στέλνοντας ένα μήνυμα σε όλα τα μέλη από τα οποία έλαβε απάντηση, στην συνέχεια τα μέλη αυτά ακολουθούν το πρωτόκολλο συμφωνίας μυστικού κλειδιού προκειμένου να εγκαθιδρύσουν ένα κοινό κλειδί συνόδου.

Το πρωτόκολλο συμφωνίας του μυστικού κλειδιού κάνει χρήση ιεραρχικών δυαδικών δένδρων. Η δομή ενός τέτοιου δένδρου φαίνεται στο παρακάτω σχήμα.



Στο παραπάνω δένδρο κάθε υπο-δένδρο αντιπροσωπεύει ένα υπο-group και κάθε κόμβος περιέχει το μυστικό κλειδί του υπο-group. Κατά συνέπεια το κλειδί που περιέχεται στην ρίζα του δένδρου είναι το μυστικό κλειδί συνόδου ολόκληρου του group. Η κατασκευή του δένδρου γίνεται από τα φύλα προς την ρίζα. Τα κλειδιά που περιέχονται στους κόμβους φύλλα π.χ.  $K_1, K_2$  κ.τ.λ. δημιουργούνται από τα μέλη π.χ.  $M_1, M_2$  αντίστοιχα με τυχαίο τρόπο και κρατούνται από τα μέλη που τα δημιούργησαν μυστικά. Τα κλειδιά που περιέχονται στους κόμβους που δεν είναι φύλλα δημιουργούνται με χρήση του πρωτοκόλλου συμφωνίας κλειδιού και την μεσολάβηση δύο μελών. Το κάθε μέλος αντιπροσωπεύει το υπο-group στο οποίο ανήκει. Όταν τα δύο μέλη συμφωνήσουν το κλειδί του νέου κοινού τους group ανακοινώνουν το κλειδί που συμφώνησαν και στα υπόλοιπα μέλη των υπο-groups τους.

Τα βήματα του πρωτοκόλλου συμφωνίας του μυστικού κλειδιού είναι τα ακόλουθα:

1. Αν ένα μέλος  $i$  επιθυμεί να συμφωνήσει ένα μυστικό κλειδί με ένα άλλο μέλος  $j$  τότε πρέπει να υπολογίσει τους παρακάτω δύο ακέραιους

$$X_i \equiv g^{3f(r_i, c_i)} \pmod{n}$$

$$Y_i \equiv S_i g^{2f(r_i, c_i)} \pmod{n}$$

Όπου  $f(x,y)$  είναι μία μονόδρομη συνάρτηση σύνοψης ενώ  $r_i$  είναι ένας τυχαία επιλεγμένος αριθμός από το μέλος  $i$ . Η τιμή του  $C_i$  εξαρτάται από τον ρόλο του μέλουνς  $i$ . Αν το μέλος  $i$  είναι αντιπρόσωπος ενός υπο-group το  $C_i$  είναι το μυστικό κλειδί του υπο-group. Διαφορετικά το  $C_i$  είναι η μυστική πληροφορία που έχει επιλεγεί τυχαία από το μέλος  $i$ .

- Το μέλος  $i$  στέλνει τους ακέραιους  $X_i$  και  $Y_i$  μαζί με το  $ID_i$  στο μέλος  $j$
- Το μέλος  $j$  υπολογίζει το  $EID_j = h(ID_i)$  και ελέγχει αν ισχύει η ισότητα

$$EID_j = Y_i^3/X_i^2$$

- Αν η ισότητα ισχύει το μέλος  $j$  δημιουργεί έναν τυχαίο αριθμό  $r_j$  και υπολογίζει το κοινό κλειδί ως εξής

$$K_{ji} = X_i^{f(r_j, Ci)} = g^{3f(r_i, Ci)f(r_j, Cj)}(\text{mod}n)$$

- Το μέλος  $j$  υπολογίζει στην συνέχεια τους ακόλουθους τρις ακέραιους

$$X_j \equiv g^{3f(j, Ci)}(\text{mod}n)$$

$$Y_j \equiv S_i g^{2f(r_j, Cj)}(\text{mod}n)$$

$$Z_j = \{X_j\} K_{ji}$$

Όπου  $Z_j$  είναι το αποτέλεσμα της κρυπτογράφησης του  $X_j$  με το μυστικό κλειδί  $K_{ji}$  με χρήση κάποιου συμμετρικού αλγορίθμου κρυπτογράφησης.

Σκοπός του είναι να παρέχει στον χρήστη  $i$  την δυνατότητα να επαληθεύσει την ορθότητα του κοινού κλειδιού.

- Το μέλος  $j$  στέλνει τους τρις ακέραιους  $X_j$ ,  $Y_j$  και  $Z_j$  μαζί με το  $ID_i$  στο μέλος  $i$ .

- Το μέλος  $i$  υπολογίζει το  $EID_j = h(ID_j)$  και ελέγχει αν ισχύει η ισότητα

$$EID_j = Y_j^3/X_j^2$$

- Αν η ισότητα ισχύει τότε το μέλος  $i$  υπολογίζει το κοινό κλειδί ως εξής

$$K_{ij} = X_j^{f(r_i, Ci)} = g^{3f(r_i, Ci)f(r_j, Cj)}(\text{mod}n)$$

- Το μέλος  $i$  ελέγχει την ορθότητα του  $K_{ij}$  αποκρυπτογραφώντας το  $Z_j$ .

- Αν τα μέλη  $i$  και  $j$  αντιπροσωπεύουν υπο-groups φέροντας την ευθύνη για την εγκατάσταση ενός κλειδιού μεταξύ των υπο-groups τους πρέπει να ενημερώσουν τα μέλη των υπο-groups που αντιπροσωπεύουν για το συμφωνηθέν μυστικό κλειδί.

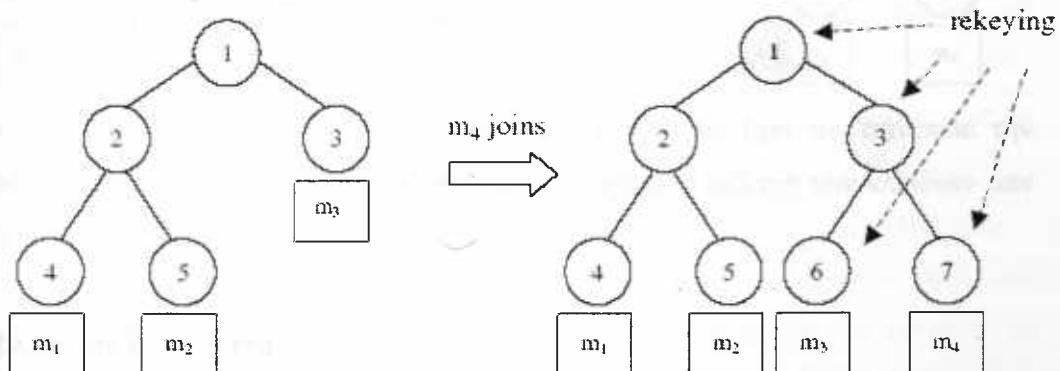
Το πρωτόκολλο που περιγράψαμε παραπάνω απαιτεί την ανταλλαγή μόνο δύο μηνυμάτων για την ολοκλήρωση των διαδικασιών αμοιβαίας αυθεντικοποίησης και συμφωνίας ενός κοινού κλειδιού συνόδου μεταξύ δύο υπο-groups. Με την εφαρμογή του παραπάνω πρωτοκόλλου στα υπο-group που παράγονται μετά την κάθε επανάληψη μετά από κάποιο αριθμό επαναλήψεων (εξαρτάται από τον αριθμό των μελών) όλα τα μέλη θα έχουν συμφωνήσει σε ένα κοινό κλειδί.

## Member Join

Κατά την διάρκεια μίας συνόδου είναι πιθανόν κάποια νέα μέλη να ζητούν περιστασιακά την είσοδό τους στο group. Κάθε νέο μέλος πρέπει να επιλέξει το μέλος του group που θα συνδεθεί (κόμβο του δένδρου). Ο κόμβος αυτός εξ' ορισμού πρέπει να έχει ένα παιδί (φύλλο) και να είναι όσο το δυνατόν ποιο κοντά στην ρίζα του δένδρου εκτός. Το πρωτόκολλο ορίζει τον κόμβο αυτό υπεύθυνο για την εισαγωγή και αποχώρηση μελών στο group.

Το νέο μέλος στέλνει στο μέλος με το οποίο θα συνδεθεί (φύλλο του κόμβου που περιγράψαμε παραπάνω) ένα μήνυμα που περιέχει μία αίτηση πρόσβασης στο group. Στο σημείο αυτό πρέπει να αναφέρουμε ότι δύο μέλη που συνδέονται σε ένα κόμβο θα τα αναφέρουμε στο έξης ως συντρόφους. Αν το νέο μέλος έχει δικαίωμα πρόσβασης στο group τότε εκτελεί με τον σύντροφό του το πρωτόκολλο συμφωνίας κλειδιού. Για να εξασφαλιστεί η αρχή του PFS πρέπει το μυστικό κλειδί του group πριν την είσοδο του νέου μέλους να αλλαχθεί. Αυτό επιτυγχάνεται με την αλλαγή όλων των κλειδιών από τον κόμβο εισόδου του νέου μέλους έως την ρίζα.

Στο παρακάτω σχήμα απεικονίζεται ένα παράδειγμα εισόδου ενός νέου μέλους



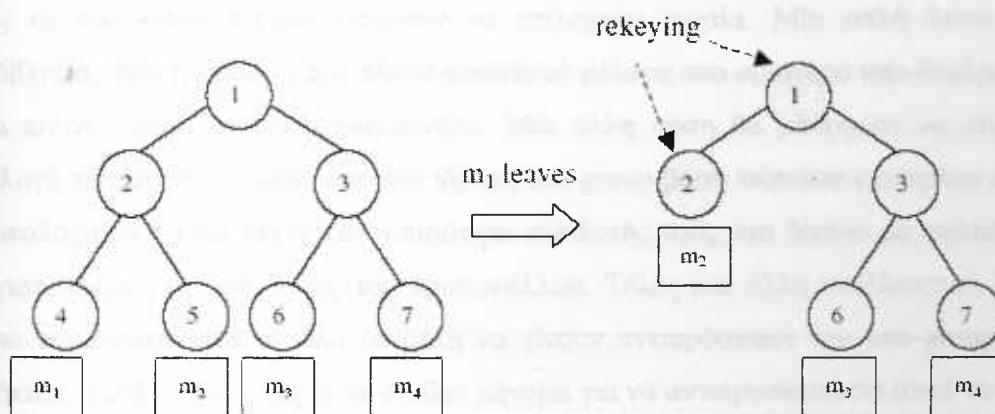
Στο παραπάνω σχήμα το νέο μέλος  $m_4$  επιθυμεί την σύνδεσή του στο group. Ο συνέταιρός του σύμφωνα με όσα αναφέραμε παραπάνω θα είναι το μέλος  $m_3$  που συνδέεται στον κόμβο 3. Το νέο μέλος θα συνδεθεί στον κόμβο 7 ενώ το  $m_3$  κατεβαίνει ένα επίπεδο και συνδέεται στον κόμβο 6. Τα κλειδιά των κόμβων 1,3,6 και 7 πρέπει να ανανεωθούν προκειμένου να εξασφαλιστεί η αρχή του PFS.

Στο σημείο αυτό πρέπει να διευκρινιστεί ότι δύο ζητήματα που περιγράψαμε παραπάνω μένουν ακόμα ανοικτά. Το πρώτο είναι πως ο νέος κόμβος θα βρει το συνέταιρό του και το δεύτερο πως γίνεται η ανανέωση των κλειδιών. Τα ζητήματα αυτά θα συζητηθούν στην συνέχεια.

## Member Departure

Κατά την διάρκεια μίας συνόδου μπορεί να παραστεί η ανάγκη για την αποχώρηση κάποιου μέλους. Η αποχώρηση μπορεί να είναι εθελούσια, να επιβληθεί από τα άλλα μέλη του group ή να οφείλεται σε σφάλμα του επικοινωνιακού μέσου. Ανεξάρτητα από τον λόγο αποχώρησης το πρωτόκολλο πρέπει να εγγυάται ότι μετά την έξοδο ενός μέλους από το group, αυτό δεν πρέπει να έχει την δυνατότητα απόκτησης οποιασδήποτε πληροφορίας που ανταλλάσσουν τα εναπομείναντα μέλη. Για να εξασφαλιστεί κάτι τέτοιο απαιτείται η αλλαγή όλων των κλειδιών που γνώριζε το μέλος που αποχώρησε.

Στο παρακάτω σχήμα έχουμε το παράδειγμα της αποχώρησης ενός μέλους



Όπως φαίνεται και στο σχήμα η αποχώρηση του μέλους  $m_1$  έχει σαν συνέπεια την μετακίνηση του  $m_2$  ένα επίπεδο ποιο πάνω και απαιτείται η αλλαγή των κλειδιών των κόμβων 1 και 2.

## Προβλήματα Υλοποίησης

Μέχρι στιγμής δώσαμε μία γενική περιγραφή του πρωτοκόλλου, πολλές πτυχές όμως που έχουν σχέση με την υλοποίηση του πρωτοκόλλου ώστε να είναι πρακτικά εφαρμόσιμο έχουν μείνει ακάλυπτες.

Όπως έχουμε ήδη αναφέρει όλα τα μέλη του group συμμετέχουν στην κατασκευή του δένδρου κλειδιών προκειμένου να εγκαθιδρύσουν το κοινό μυστικό κλειδί συνόδου. Ένα βασικό ερώτημα που γεννάται είναι πως τα μέλη γνωρίζουν την δομή των υπο-δένδρων στα οποία ανήκουν.

Αρχικά ο δημιουργός του group με βάση την λίστα των μελών που θα συμμετέχουν στο group με την έναρξη λειτουργίας του καταρτίζει ένα δένδρο και το διανέμει σε όλα τα μέλη που θα απαρτίζουν αρχικά το group. Η διανομή στα μέλη

γίνεται με την ενσωμάτωση του δένδρου στο μήνυμα αρχικοποίησης. Με την λήψη του μηνύματος τα μέλη του group αποθηκεύουν τοπικά την δομή που έλαβαν και αρχίζουν την εκτέλεση του πρωτοκόλλου για την συμφωνία και εγκαθίδρυση του κοινού κλειδιού συνόδου.

Κατά την είσοδο ενός νέου μέλους ή την αποχώρηση κάποιου μέλους από το group, τα μέλη του group μπορούν να προσδιορίσουν την νέα δομή του δένδρου από την ροή των μηνυμάτων.

Όπως έχει ήδη αναφερθεί για τον υπολογισμό της τιμής του κλειδιού ενός κόμβου που δεν είναι φύλλο επιλέγονται τυχαία δύο μέλη σαν αντιπρόσωποι των υπο-groups (υπο-δένδρων) που συνδέονται στον εν λόγῳ κόμβο. Το πρόβλημα που τίθεται είναι πώς οι δύο αυτοί κόμβοι μπορούν να επιλεγούν τυχαία. Μία απλή λύση στο πρόβλημα είναι η επιλογή του πλέον αριστερού μέλους στο αριστερό υπο-δένδρο και του πλέον δεξιού στο δεξιό υπό-δένδρο. Μία άλλη λύση θα μπορούσε να είναι η επιλογή των αντιπροσώπων από τον ιδρυτή του group βάση κάποιων κριτηρίων όπως η υπολογιστική τους ισχύς και η ποιότητα σύνδεσής τους στο δίκτυο με σκοπό την μεγιστοποίηση της απόδοσης του πρωτοκόλλου. Τέλος μία άλλη εναλλακτική λύση δίνει την δυνατότητα σε όλα τα μέλη να γίνουν αντιπρόσωποι του υπο-group που ανήκουν. Κάθε μέλος μπορεί να στείλει μήνυμα για να αντιροσωπεύσει αυτό το υπο-group το οποίο ανήκει. Στην περίπτωση όμως που λάβει ένα τέτοιου τύπου μήνυμα από κάποιο άλλο μέλος στο ίδιο υπο-group σταματάει την αποστολή του δικού του μηνύματος. Με τον τρόπο αυτό το πλέον ταχύ μέλος επιλέγεται σαν αντιπρόσωπος.

Ένα άλλο πρόβλημα που έχει ήδη αναφερθεί και αναζητεί λύση είναι ο τρόπος με τον οποίο ένα νέο μέλος που επιθυμεί την είσοδό του στο group, θα γνωρίζει το κόμβο του δένδρου που πρέπει να συνδεθεί. Στην περιγραφή της φάσης εισόδου νέου μέλους αναφέραμε ότι προκειμένου το δένδρο να παραμείνει ζυγισμένο πρέπει το νέο μέλος να συνδεθεί στον κόμβο - φύλλο που βρίσκεται ποιο κοντά στην ρίζα του δένδρου. Για να γίνει κάτι τέτοιο εφικτό το νέο μέλος πρέπει να γνωρίζει την δομή του δένδρου. Για να αποκτήσει το νέο μέλος το δένδρο μπορεί να κάνει μία αίτηση σύνδεσης σε οποιοδήποτε μέλος του group, εφ' όσων το νέο μέλος έχει δικαίωμα πρόσβασης στο group το ήδη συνδεδεμένο μέλος θα του απαντήσει με ένα μήνυμα που θα περιέχει το δένδρο του group. Έχοντας πλέον την δομή του δένδρου το νέο μέλος μπορεί να εντοπίσει τον κατάλληλο κόμβο και να συνδεθεί σ' αυτόν. Στην περίπτωση που το νέο μέλος δεν συνδεθεί στον πλέον κοντινό στην ρίζα του δένδρου κόμβο, το πρωτόκολλο συνεχίζει να δουλεύει αλλά μειώνεται η απόδοσή του.

Ένα τελευταίο ζήτημα που παραμένει ανοικτό είναι η αναθεώρηση του κλειδιού συνόδου κατά την είσοδο νέου ή αποχώρηση υπάρχοντος μέλους. Όταν ένα μέλος εισέρχεται ή κάποιο αποχωρεί από ένα group αλλάζει η θέση των κόμβων του δένδρου στο οποίο τα μέλη αυτά συνδέονται. Κατά συνέπεια απαιτείται η αναθεώρηση των κλειδιών αυτών των κόμβων και των κλειδιών των προγονικών τους κόμβων. Στην περίπτωση των κόμβων φύλλων η αλλαγή του κλειδιού είναι απλή, μίας και οι ίδιοι οι κόμβοι επιλέγουν τυχαία τα κλειδιά τους. Για την αλλαγή των κλειδιών των προγονικών κόμβων μπορούν να χρησιμοποιηθούν δύο τρόποι. Ο πρώτος είναι να εκτελεστεί το πρωτόκολλο συμφωνίας του κοινού κλειδιού για όλους τους κόμβους που δεν είναι φύλλα. Ο δεύτερος τρόπος είναι να επιλεγεί τυχαία ένα νέο κλειδί να κρυπτογραφηθεί με το παλιό κλειδί και να αποσταλεί σε όλα τα μέλη του group. Την επιλογή του νέου κλειδιού μπορεί να την κάνει ο συνέταιρος του μέλους που εισήλθε ή αποχώρησε. Αν ο συνέταιρος έχει αποχωρίσει τότε επιλέγεται τυχαία ένα μέλος από το ίδιο υπο-group. Ο δεύτερος τρόπος είναι ποιο γρήγορος όμως καταστρατηγεί την αρχή της από κοινού δημιουργίας του κλειδιού συνόδου και επιπλέον δεν εξασφαλίζει την αρχή του PFS αφού κάνει χρήση του παλιού κλειδιού συνόδου που είναι γνωστό στα μέλη που αποχώρησαν.

### ***Pairing – based Group Key Agreement (PAGKA)***

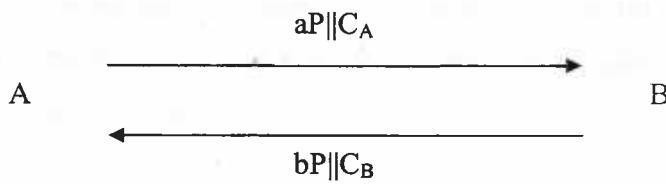
Το πρωτόκολλο PAGKA [52] συνδυάζει τα πλεονεκτήματα της χρήσης ενός δημόσιου κλειδιού για το group και δενδρικών δομών για τα κλειδιά προκειμένου να επιτύχει την δημιουργία ενός αποτελεσματικού πρωτοκόλλου συμφωνίας μυστικού κλειδιού που εξασφαλίζει τις αρχές του PFS, BS και Key Authentication.

Θα περιγράψουμε αρχικά το πρωτόκολλο συμφωνίας ενός μυστικού κλειδιού μεταξύ δύο μελών έστω A και B και στην συνέχεια θα το προεκτείνουμε για n μέλη. Μία αρχή πιστοποίησης CA δημιουργεί αρχικά τα πιστοποιητικά των A και B. Η μορφή του πιστοποιητικού είναι η ακόλουθη

$$C_A = (I_A || \mu_A // P || G || S_{CA}(I_A || \mu_A // P || Q))$$

Όπου  $I_A$  είναι η ταυτότητα του A ενώ ο συμβολισμός  $\parallel$  ερμηνεύεται σαν αλληλουχία των δεδομένων που διαχωρίζει. Το  $S_{CA}$  είναι η υπογραφή του CA. Το δημόσιο κλειδί του A είναι το  $\mu_A=xP$ , το x είναι το ιδιωτικό κλειδί του A με  $x \in Z_q^*$ . Τα στοιχεία P και Q είναι δημόσια και χρησιμοποιούνται για την δημιουργία του  $\mu_A$  και προσωρινών δημόσιων κλειδιών.

Η ροή των μηνυμάτων μεταξύ του A και B είναι η ακόλουθη.



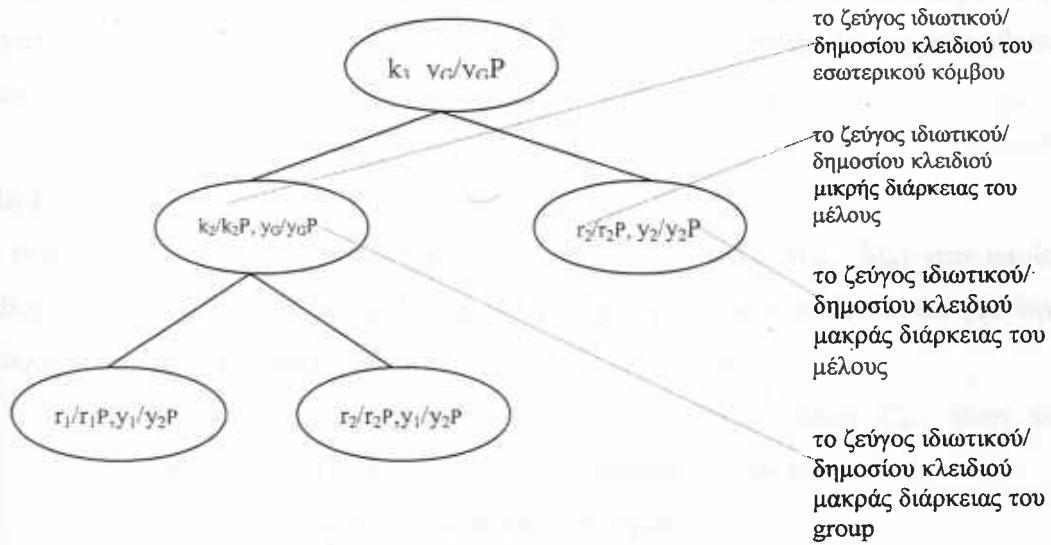
$$k_A = \hat{e} ( bP + H_1(bP||yP)yP, Q )^{a + H_1(aP||xP)x}$$

$$k_B = \hat{e} ( aP + H_1(aP||xP)xP, Q )^{b + H_1(aP||yP)y}$$

$$k_{AB} = \hat{e} ( P, Q )^{(a + H_1(aP||xP)x)(b + H_1(bP||yP)y)}$$

Αν  $G_1$  και  $G_2$  είναι δύο κυκλικά groups κάποιου μεγάλου πρώτου έστω ρ τότε το  $\hat{e}$  είναι μία διγραμμική απεικόνιση του  $G_1 \times G_1$  στο  $G_2$ .

Ως δούμε στο σημείο αυτό πώς το παραπάνω πρωτόκολλο μπορεί να γενικευτεί για να δώσει λύση στο πρόβλημα της συμφωνίας ενός μυστικού κλειδιού συνόδου μεταξύ των μελών ενός group. Στο σχήμα που ακολουθεί έχουμε το παράδειγμα ενός δυαδικού δένδρου κλειδιών πάνω στο οποίο στηρίζεται η λειτουργία του πρωτοκόλλου.



κάθε κόμβος φύλλο του δένδρου  $L_i$  συσχετίζεται με ένα μέλος του group και κάθε εσωτερικός κόμβος  $I_i$  περιέχει ένα εσωτερικό κλειδί. Προκειμένου οι εσωτερικοί κόμβοι να εξασφαλίζουν την αρχή του key authentication στους κόμβους φύλλα συσχετίζεται του μακράς διάρκειας κλειδί του group  $y_G P$  με αυτούς. Το ιδιωτικό κλειδί του group  $y_G$  που αντιστοιχεί στο δημόσιο είναι γνωστό σε όλα τα μέλη του group. Το μικρής διάρκειας ιδιωτικό κλειδί  $r_i$  του κόμβου φύλλου  $L_i$  επιλέγεται

τυχαία από το μέλος  $M_i$  ενώ το  $k_i$  του εσωτερικού κόμβου  $I_i$  είναι το αποτέλεσμα της εφαρμογής του πρωτοκόλλου συμφωνίας μυστικού κλειδιού δύο μελών που περιγράψαμε παραπάνω μεταξύ των κόμβων παιδιών του κόμβου  $I_i$ . Το  $k_i$  μπορεί να υπολογιστεί περιοδικά ως έξης

$$k_i = r_i$$

$$k_i = H_1(\hat{e}(P, Q)^{(ki-1 + H_2(ki-1P|yGP)yG)*(ri + H_2(riP|yiP)y_i)}$$

όπου  $H_1$  και  $H_2$  είναι δύο συναρτήσεις σύνοψης που ορίζονται ως έξης

$$H_1 : G_2 \rightarrow Z_p^*$$

$$H_2 : S \rightarrow Z_p^* \quad S = \{uP||vP | u, v \in Z_p^* \text{ & } P \in G_1\}$$

Για να είναι ένα πρωτόκολλο συμφωνίας μυστικού κλειδιού συνόδου ενός group πλήρες πρέπει να διαχειρίζεται αποτελεσματικά τις αλλαγές στο membership ενός group όπως είναι η είσοδος νέου μέλους, η αποχώρηση μέλους, η αποκοπή μέρους του group εξαιτίας κάποιου επικοινωνιακού λάθους και τέλος η επανένωση των αποκομμένων τμημάτων ενός group. Στην συνέχεια θα εξετάσουμε τους μηχανισμούς εισόδου νέου μέλους (Join Protocol) και αποχώρησης μέλους (Leave Protocol) ενώ οι περιπτώσεις της αποκοπής και επανένωσης κάποιου υπο-group θα παραληφθούν χάριν συντομίας.

### Join Protocol

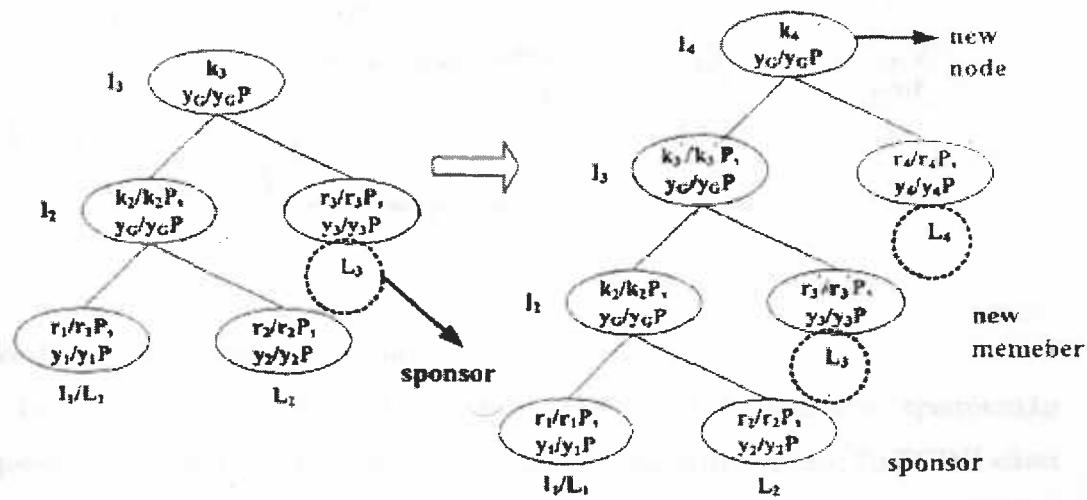
Θεωρούμε ένα group έστω  $G$  που περιέχει η μέλη  $\{M_1, M_2, \dots, M_n\}$  στο οποίο επιθυμεί να συνδεθεί ένα νέο μέλος το  $M_{n+1}$ . Τα βήματα που απαιτούνται για την ολοκλήρωση μίας επιτυχημένης σύνδεσης είναι τα ακόλουθα.

1. Ο  $M_{n+1}$  στέλνει στο group το μήνυμα  $r_{n+1}P||C_{n+1}$  όπου  $C_{n+1}$  είναι το πιστοποιητικό του  $n+1$  που περιέχει το δημόσιο κλειδί του.
2. Το βήμα αυτό περιέχει τις ακόλουθες ενέργειες
  - Δημιουργείται ένας κόμβος ρίζα που περιέχει σαν παιδία του τον παλιό κόμβο ρίζα (αριστερό παιδί) και το νεοεισελθέν μέλος (δεξί παιδί)
  - Ο δεξιός πλησιέστερος στην ρίζα κόμβος  $M_n$  (πριν την είσοδο του νέου μέλους) επιλέγεται σαν υπεύθυνος για την εισαγωγή του νέου μέλους. Το μέλος που συνδέεται με τον κόμβο αυτό ελέγχει το πιστοποιητικό του νέου μέλους. Στην περίπτωση αποτυχημένου ελέγχου  $M_n$  τερματίζει διαδικασία.

- Ο  $M_n$  παράγει μία τυχαία τιμή  $r_n$  που αποτελεί το νέο του μυστικό κλειδί μικρής διάρκειας και υπολογίζει όλα τα κλειδιά που επηρεάζονται από την αλλαγή του μυστικού του κλειδιού.
- Ο  $M_n$  στέλνει στο group (περιέχει και το νέο μέλος) το παρακάτω μήνυμα  $B_{n+1}||C_n||E_s(y_G)||y_GQ$  όπου  $B_{n+1}$  είναι το δένδρο του  $M_n$  που περιέχει όλα τα αλλαγμένα κλειδιά. Το  $s$  είναι το νέο κλειδί του group όπως υπολογίστηκε απ' τον  $M_n$ .

3. Κάθε μέλος υπολογίζει το νέο κλειδί του group και ελέγχει αν είναι ίδιο με το  $s$  που υπολόγισε ο  $M_n$ .

Στο σχήμα που ακολουθεί απεικονίζονται οι αλλαγές στο δένδρο κλειδιών μετά την είσοδο ένός νέου μέλους του  $M_4$ .



## Leave Protocol

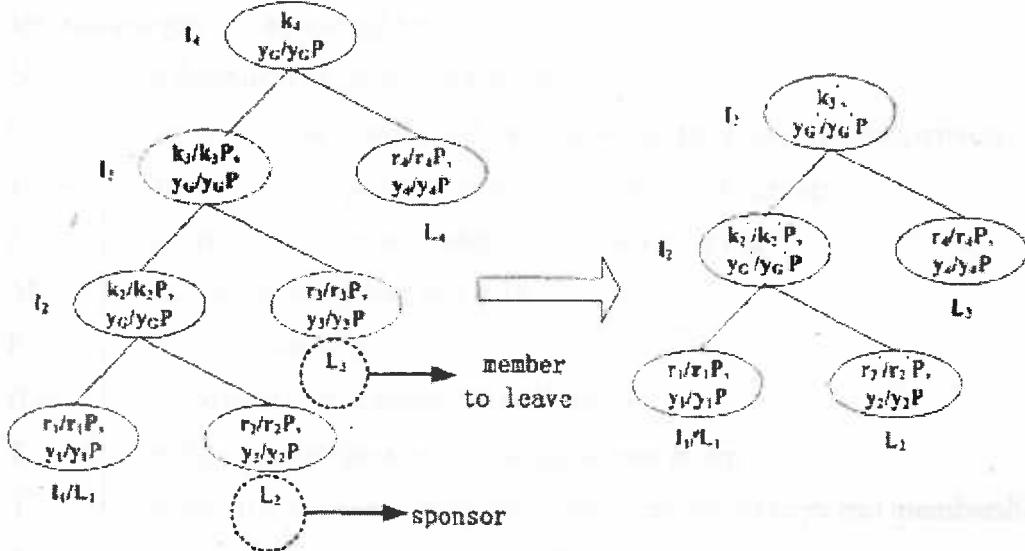
Έστω ότι το μέλος  $M_i$  ( $i \leq n$ ) αποχωρεί από το group  $\{M_1, M_2, \dots, M_n\}$ . Το μέλος  $M_s$  που επιλέγεται σαν υπεύθυνο για την αποχώρηση του μέλους  $i$  είναι το  $M_{i-1}$  όταν  $i > 1$  και το  $M_2$  για  $i = 1$ . Τα βήματα της διαδικασίας αποχώρησης είναι τα ακόλουθα.

1. Όλα τα μέλη προάγουν το κόμβο φύλλο  $L_{i+1}$  στην θέση που κατείχε ο εσωτερικός κόμβος  $I_{i+1}$  πριν την αποχώρηση του μέλους  $M_i$
2. Ο  $M_s$  δημιουργεί μία τυχαία τιμή  $r_n$  που την χρησιμοποιεί σαν το νέο του μυστικό κλειδί και επιπλέον δημιουργεί ένα νέο μακράς διάρκειας ιδιωτικό κλειδί του group ( $y_G$ ). Στην συνέχεια υπολογίζει όλα τα κλειδιά που επηρεάζονται από τις αλλαγές, κρυπτογραφεί το  $y_G$  με το νέο κλειδί του group

και αποστέλλει μέσω ενός μηνύματος ( $B_{n-1} \parallel E_s(y_G)$ ) αυτές τις πληροφορίες στα εναπομείναντα μέλη του group.

- Σε αυτό το βήμα κάθε μέλος υπολογίζει το νέο κλειδί του group και ανανεώνει το μακράς διάρκειας μυστικό κλειδί του group.

Στο παρακάτω σχήμα απεικονίζεται η δομή του δένδρου κλειδιών όπως διαμορφώνεται μετά την αποχώρηση του μέλους  $M_3$



### Tree-Based Group Diffie-Helman

To Tree-Based Group Diffie-Helman (TGDH) [53] είναι ένα πρωτόκολλο συμφωνίας του μυστικού κλειδιού ενός group με συνεργατικό τρόπο. To TGDH κάνει χρήση δένδρων που περιέχουν κλειδιά (key trees) και του πρωτοκόλλου Diffie-Helman. Προκειμένου το TGDH πρωτόκολλο να επιτυγχάνει ανοχή στα λάθη που μπορεί να προέρχονται από αποτυχία του δικτύου ή λάθος της εφαρμογής χρειάζεται ένα group communication σύστημα που υποστηρίζει το Virtual Synchrony (VS) μοντέλο. Στην περίπτωση που το υποκείμενο σύστημα επικοινωνίας δεν υποστηρίζει το VS μοντέλο η ασφάλεια του TGDH δεν επηρεάζεται απλά μειώνεται η ανοχή του πρωτοκόλλου σε λάθη.

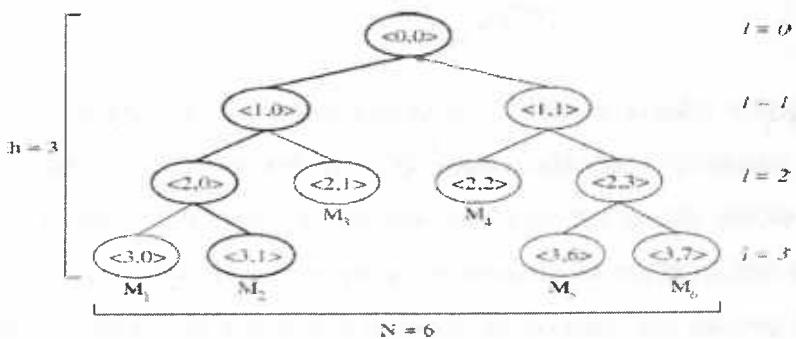
Το πρωτόκολλο TGDH υποστηρίζει τις αρχές του Forward και Backward Secrecy δεν υποστηρίζει όμως τις αρχές του Perfect Forward και Backward Secrecy μίας και δεν κάνει χρήση κάποιου είδους μυστικής πληροφορίας μακράς διάρκειας όπως στην περίπτωση των δύο ID-based σχημάτων που περιγράφηκαν παραπάνω. Η αρχή του key authentication δεν υποστηρίζεται σαν μέρος του πρωτοκόλλου διαχείρισης κλειδιού άλλα στηρίζεται στην χρήση αυθεντικοποιημένων καναλιών επικοινωνίας.

Αυτό σημαίνει ότι όλα τα μηνύματα θα υπογράφονται ψηφιακά από τον αποστολέα τους με κάποιο αξιόπιστο σχήμα ψηφιακών υπογραφών που στηρίζονται την κρυπτογραφία δημόσιου κλειδιού όπως το DSA ή το RSA. Οι λήπτες των μηνυμάτων θα πρέπει με την σειρά τους να επαληθεύουν την υπογραφή κάθε μηνύματος που έλαβαν.

Στην συνέχεια θα δώσουμε την σημειολογία και κάποιους ορισμούς απαραίτητους για την κατανόηση του πρωτοκόλλου.

N	ο αριθμός των μελών του group
C	το σύνολο των συνδεδεμένων μελών την τρέχουσα χρονική στιγμή
L	το σύνολο των μελών που αποχώρησαν από το group
J	το σύνολο των νεοεισελθέντων μελών στο group
M <sub>i</sub>	το i μέλος του group με $i \in \{1, \dots, N\}$
h	το ύψος του δένδρου
(l,u)	ο u κόμβος στο επίπεδο l του δένδρου
T <sub>i</sub>	η όψη του δένδρου από το M <sub>i</sub> μέλος του group
F <sub>i</sub>	η όψη του δένδρου από το M <sub>i</sub> μέλος μετά την αλλαγή στο membership
T <sub>&lt;i,j&gt;</sub>	ένα υποδένδρο με ρίζα τον κόμβο <i,j>
bkey	το blinded key ενός κλειδιού. Αν K το κλειδί τότε το bkey του K είναι η ποσότητα $a^K \bmod p$
BK <sub>i</sub> *	το σύνολο των blinded keys του M <sub>i</sub>
p,q	πρώτοι αριθμοί
a	βάση εκθετοποίησης

Η χρήση δένδρων που περιέχουν κλειδιά όπως ήδη έχουμε δει χρησιμοποιείται σε πολλά κεντρικοποιημένα πρωτόκολλα στο TGDH οι δενδρικές δομές χρησιμοποιούνται με ένα πλήρως συνεργατικό και κατανεμημένο τρόπο. Παράδειγμα ενός τέτοιου δένδρου απεικονίζεται στο παρακάτω σχήμα



Όπως παρατηρούμε πρόκειται για ένα δυαδικό δένδρο όπου η ρίζα του ανήκει στο επίπεδο 0 ενώ τα φύλλα στο επίπεδο h. Στα δυαδικά δένδρα οι κόμβοι είναι είτε φύλλα είτε γονής δύο άλλων κόμβων. Οι κόμβοι συμβολίζονται με  $\langle l, u \rangle$  όπου  $0 \leq u \leq 2^l - 1$  μίας και κάθε επίπεδο 1 περιέχει το πολύ  $2^l$  κόμβους. Σε κάθε κόμβο  $\langle l, u \rangle$  αντιστοιχίζεται ένα κλειδί  $K_{\langle l, u \rangle}$  και ένα blinded key (bkey)  $BK_{\langle l, u \rangle} = f(K_{\langle l, u \rangle})$  όπου  $f$  είναι μία συνάρτηση της μορφής  $f(k) = a^k \bmod p$ . Αν θεωρήσουμε ότι το μέλος  $M_i$  είναι συσχετισμένο με τον κόμβο  $\langle l, u \rangle$  τότε στον κόμβο αυτό έχει αντιστοιχηθεί το κλειδί  $K_{\langle l, u \rangle}$  που έχει επιλεχθεί από τον  $M_i$  με τυχαίο τρόπο. Επιπλέον το μέλος  $M_i$  γνωρίζει όλα τα κλειδιά που περιέχουν οι κόμβοι από τον κόμβο  $\langle l, u \rangle$  στον οποίο έχει συσχετιστεί μέχρι την ρίζα του δένδρου  $\langle 0, 0 \rangle$ , αυτό το μονοπάτι κλειδιών για το μέλος  $M_i$  θα συμβολίζεται ως  $KEY_i^*$ . Αν στο παράδειγμα του παραπάνω δένδρου θεωρήσουμε το μέλος  $M_2$  τότε ο  $M_2$  γνωρίζει κάθε κλειδί  $\{K_{\langle 3, 1 \rangle}, K_{\langle 2, 0 \rangle}, K_{\langle 1, 0 \rangle}, K_{\langle 0, 0 \rangle}\}$  στο  $KEY_2^* = \{(3, 1), (2, 0), (1, 0), (0, 0)\}$  και κάθε bkey  $BK_2^* = \{BK_{(0,0)}, BK_{(1,0)}, \dots, BK_{(3,7)}\}$ . Κάθε κλειδί  $K_{\langle l, u \rangle}$  υπολογίζεται περιοδικά με χρήση του παρακάτω τύπου.

$$\begin{aligned} K_{\langle l, u \rangle} &= (BK_{\langle l+1, 2v+1 \rangle})^{K_{\langle l+1, 2v \rangle}} \bmod p = (BK_{\langle l+1, 2v \rangle})^{K_{\langle l+1, 2v+1 \rangle}} \bmod p \\ &= a^{K_{\langle l+1, 2v \rangle} K_{\langle l+1, 2v+1 \rangle}} \bmod p = f(K_{\langle l+1, 2v \rangle} K_{\langle l+1, 2v+1 \rangle}) \end{aligned}$$

Ο υπολογισμός του κλειδιού που περιέχεται στον κόμβο  $\langle l, u \rangle$  προϋποθέτει την γνώση του κλειδιού ενός από τους κόμβους παιδιά του  $\langle l, u \rangle$  και του bkey του άλλου κόμβου παιδιού. Το κλειδί  $K_{\langle 0, 0 \rangle}$  που περιέχεται στην ρίζα του δένδρου είναι το μυστικό κλειδί που γνωρίζουν όλα τα μέλη του group. Το  $K_{\langle 0, 0 \rangle}$  δεν χρησιμοποιείται απευθείας για την κρυπτογράφηση ή την αυθεντικοποίηση δεδομένων αλλά αποτελεί την είσοδο μίας συνάρτησης σύνοψης η οποία παράγει τα κλειδιά που θα χρησιμοποιηθούν στην συνέχεια. Αν θεωρήσουμε το παράδειγμα του δένδρου του παραπάνω σχήματος διαπιστώνουμε ότι το μέλος  $M_2$  μπορεί να υπολογίσει τα κλειδιά  $K_{\langle 2, 0 \rangle}, K_{\langle 1, 0 \rangle}, K_{\langle 0, 0 \rangle}$  χρησιμοποιώντας τα bkeys  $BK_{\langle 3, 0 \rangle}, BK_{\langle 2, 1 \rangle}, BK_{\langle 1, 1 \rangle}$  και το κλειδί του  $K_{\langle 3, 1 \rangle}$ . Το τελικό κλειδί του group  $K_{\langle 0, 0 \rangle}$  υπολογίζεται από την σχέση:

$$K_{\langle 0, 0 \rangle} = a^{(a^{r_3(a^{r_1 r_2})} \times a^{r_4(a^{r_5 r_6})})}$$

Για τις ανάγκες περιγραφής του πρωτοκόλλου TGDH οι κόμβοι που έχουν κοινό πατέρα θα ονομάζονται από δω και στο εξής κόμβοι αδέρφια. Το σύνολο των κόμβων αδερφών των κόμβων που ανήκουν στο μονοπάτι από ένα κόμβο φύλλο στον οποίο συνδέεται ένα μέλος έστω  $M_i$  έως την ρίζα του δένδρου θα συμβολίζεται ως  $CO_i^*$  (co-path). Στο παράδειγμα του παραπάνω δένδρου το co-path του μέλους  $M_2$  είναι το σύνολο των κόμβων  $\{\langle 3, 0 \rangle, \langle 2, 1 \rangle, \langle 1, 1 \rangle\}$ . Κάθε μέλος που συνδέεται με τον κόμβο

φύλλο  $\langle l, v \rangle$  μπορεί να αποκομίσει το κλειδί του group  $K_{\langle 0,0 \rangle}$  ένα γνωρίζει όλα τα bkeys του  $CO_i^*$  και το μυστικό του, τυχαία επιλεγμένο κλειδί  $K_{\langle l, v \rangle}$ .

## Περιγραφή του TGDH

Το TGDH περιέχει τέσσερα επιμέρους πρωτόκολλα κάθε ένα από τα οποία αναλαμβάνει την διαχείριση μίας αλλαγής στο membership. Τα πρωτόκολλα αυτά είναι τα :

- Join Protocol
- Leave Protocol
- Merge Protocol
- Partition Protocol

Τα παραπάνω τέσσερα πρωτόκολλα διαμοιράζονται ένα κοινό πλαίσιο εργασίας με τα έξης χαρακτηριστικά:

- Όλα τα μέλη του group συνεισφέρουν εξίσου ένα κομμάτι πληροφορίας (share) για την δημιουργία του κοινού μυστικού κλειδιού. Ο υπολογισμός του μυστικού κλειδιού γίνεται συναρτήσει των shares των τρεχόντων μελών του group.
- Κάθε share είναι μυστικό και δεν αποκαλύπτεται ποτέ.
- Με την είσοδο νέου μέλους στο group, το share του νέου μέλους συνυπολογίζεται στο μυστικό κλειδί, ενώ ένα από τα παλιά μέλη αλλάζει το share του.
- Με την αποχώρηση μέλους από το group το share του παύει να συμμετέχει στον υπολογισμό του κλειδιού, ενώ ένα τουλάχιστον από τα εναπομείναντα μέλη αλλάζει το share του.
- Όλα τα μηνύματα των πρωτοκόλλων υπογράφονται ψηφιακά, περιέχουν χρονοσήμανση, έχουν αριθμό σειράς και περιέχουν τον προσδιοριστή του αποστολέα.

Με κάθε αλλαγή στο membership όλα τα μέλη του group ανανεώνουν την δομή του δέντρου κλειδιών. Μίας και όπως έχουμε ήδη αναφέρει το υποκείμενο σύστημα επικοινωνίας υποστηρίζει την αρχή του VS όλα τα μέλη του group θα υπολογίσουν το ίδιο δένδρο κλειδιών.

Έχουμε ήδη αναφέρει ότι αν ένα μέλος γνωρίζει τα bkeys των κόμβων που ανήκουν στο co-path του και με χρήση του μυστικού του κλειδιού μπορεί να υπολογίσει το κοινό μυστικό κλειδί του group. Αν και δεν είναι άκρος απαραίτητο για

τον υπολογισμό του μυστικού κλειδιού το TGDH προϋποθέτει την γνώση των bkeys όχι μόνο κόμβων που ανήκουν στο co-path άλλα όλων των εσωτερικών κόμβων του δένδρου.

Κάποιο μέλος του group μπορεί να επιφορτιστεί με ένα ειδικό ρόλο (sponsor). Στις αρμοδιότητες του sponsor περιέχεται ο υπολογισμός των ενδιάμεσων κλειδιών και η αποστολή μηνυμάτων σε όλα τα μέλη του group που περιέχουν την όψη του δένδρου του αποστολέα, στην όψη του δένδρου περιέχονται μόνο τα bkeys που ο αποστολέας γνωρίζει.

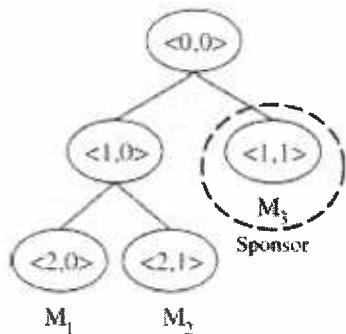
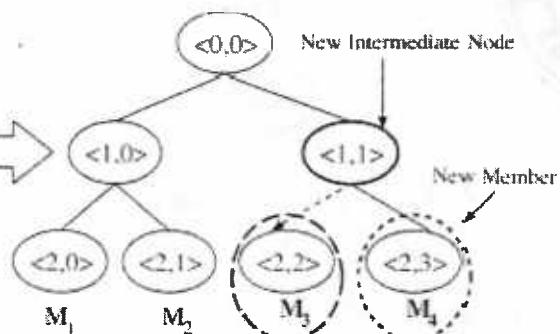
### Join Protocol

Έστω ένα group που περιέχει η μέλη  $\{M_1, M_2, \dots, M_n\}$ . Το πρωτόκολλο ξεκινά με την αποστολή σε όλα τα μέλη του group ενός μηνύματος από το νέο μέλος ( $M_{n+1}$ ). Το μήνυμα περιέχει το bkey ( $BK_{<0,0>}$ ) του νέου μέλους .

Με την λήψη του μηνύματος κάθε τρέχον μέλος του group υπολογίζει την θέση στο δένδρο στην οποία το νέο μέλος πρέπει να συνδεθεί. Το σημείο αυτό στην περίπτωση που το δένδρο δεν είναι πλήρως ζυγισμένο είναι ο δεξιότερος κόμβος που είναι όσο το δυνατόν ποιο κοντά στην ρίζα του δένδρου, αν έχουμε πλήρως ζυγισμένο δένδρο το νέο μέλος συνδέεται στον κόμβο ρίζα. Σαν sponsor επιλέγεται το πλέον δεξιό φύλλο στο υποδένδρο με ρίζα τον κόμβο εισόδου. Στην συνέχεια κάθε μέλος δημιουργεί ένα ενδιάμεσο κόμβο και ένα νέο κόμβο φύλλο στον οποίο θα συνδεθεί το νέο μέλος, επίσης θέτει τον νέο ενδιάμεσο κόμβο σαν πατέρα του κόμβου του νέου μέλους και του κόμβου που επιλέχθηκε σαν σημείο εισόδου. Μετά την ανανέωση του δένδρου όλα τα μέλη με εξαίρεση τον sponsor διακόπτουν την λειτουργία τους (block). Στην συνέχεια ο sponsor στέλνει όλα σε όλα τα άλλα μέλη (εμπεριέχεται και το νέο) το ανανεωμένο δένδρο που περιέχει όλα τα bkeys . Με την λήψη του μηνύματος τα μέλη μπορούν να υπολογίσουν το νέο μυστικό κλειδί του group.

Η αποστολή ολόκληρου του δένδρου που περιέχει bkeys σε όλα τα μέλη φαίνεται με την πρώτη ματιά σαν σπατάλη μίας και τα περισσότερα bkeys είναι ήδη γνωστά στα μέλη του group. Όμως με αυτό τον τρόπο το πρωτόκολλο επιτυγχάνει την αποστολή ενός επιπλέον unicast μηνύματος στο νεοεισελθέν μέλος που δεν γνωρίζει κανένα bkey.

Ένα παράδειγμα εισαγωγής νέου μέλους απεικονίζεται στο παρακάτω σχήμα.

Tree T<sub>3</sub>Tree T<sub>3</sub>

Στο παράδειγμα της παραπάνω εικόνας το μέλος M<sub>4</sub> εισέρχεται στο group. Οι ενέργειες του sponsor (M<sub>3</sub>) είναι οι ακόλουθες:

- Μετονομάζει τον κόμβο <1,1> σε <2,2>
- Δημιουργεί ένα ενδιάμεσο κόμβο <1,1> και τον κόμβο του νέου μέλους <2,3>
- Τοποθετεί τον <1,1> σαν πατέρα του <2,2> και <2,3>

Τα δύο πρώτα βήματα όπως έχουμε πει εκτελούνται και από όλα τα άλλα μέλη του group.

Για την ολοκλήρωση της διαδικασίας εισαγωγής νέου μέλους απαιτούνται δύο γύροι και η ανταλλαγή τριών μηνυμάτων.

### Leave Protocol

Έστω ένα group με η μέλη από το οποίο το μέλος M<sub>d</sub> αποχωρεί. Στο πρωτόκολλο αποχώρησης θέτουμε σαν sponsor το πλέον αριστερό κόμβο του υποδένδρου με ρίζα τον αδερφό κόμβο του μέλους που αποχώρησε.

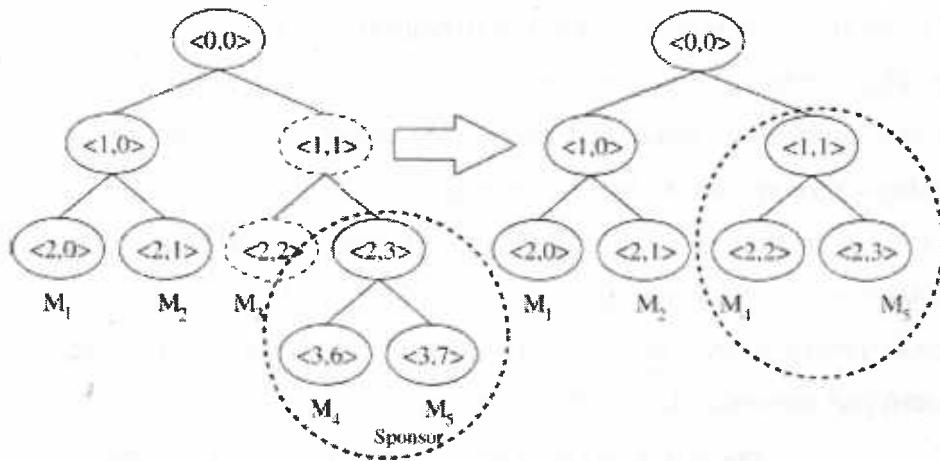
Στο πρωτόκολλο αποχώρησης με την έξοδο του M<sub>d</sub> κάθε μέλος που παραμένει στο group ανανεώνει το δένδρο κλειδιών του διαγράφοντας τον κόμβο φύλλο στον οποίο συνδέονταν ο M<sub>d</sub>. Ο κόμβος αδερφός του M<sub>d</sub> προωθείται στην θέση του κόμβου πατέρα του. Στην συνέχεια ο sponsor δημιουργεί ένα καινούργιο share και υπολογίζει όλα τα ζεύγη κλειδιών – bkeys των κόμβων στο μονοπάτι από τον κόμβο που τον περιέχει μέχρι την ρίζα του δένδρου. Με την λήξη του προηγούμενου υπολογισμού ο sponsor αποστέλλει σε όλα τα μέλη του group την δική του δένδρου με όλα τα bkeys που γνωρίζει.

Στο παρακάτω σχήμα απεικονίζεται η περίπτωση αποχώρησης του μέλους M<sub>3</sub> από το group.



Tree T<sub>3</sub>

Tree T<sub>3</sub>



Μετά την αποχώρηση του  $M_3$  οι εναπομείναντες κόμβοι διαγράφουν τους κόμβους  $<1,1>$  και  $<2,2>$ . Μετά την ανανέωση του δένδρου ο sponsor ( $M_5$ ) δημιουργεί ένα καινούργιο share στην συνέχεια υπολογίζει τα  $K_{<1,1>}$ ,  $K_{<0,0>}$ ,  $BK_{<2,3>}$  και  $BK_{<1,1>}$  και αποστέλλει σε όλα τα μέλη του group το ανανεωμένο δένδρο. Με την λήψη του νέου δένδρου τα μέλη είναι πλέον σε θέση να υπολογίσουν το κλειδί του group.

Για την ολοκλήρωση του πρωτοκόλλου απαιτείται ένας γύρος και ένα μήνυμα.

## Partition Protocol

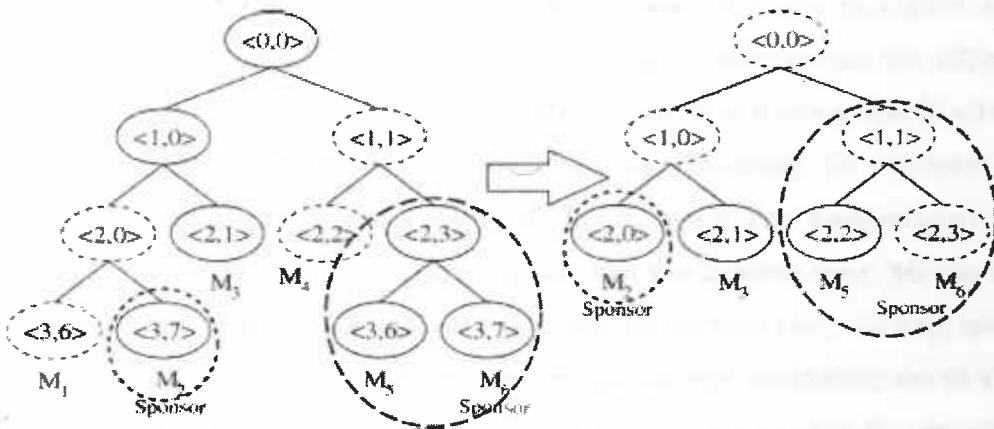
Ο διαμελισμός (partition) ενός group μπορεί να είναι αποτέλεσμα αποτυχίας του δικτύου. Στην περίπτωση του partition τα μέλη το αντιλαμβάνονται σαν πολλαπλή αποχώρηση μελών. Για την αποκατάσταση του δένδρου κλειδιών μετά από ένα partition απαιτούνται πολύ γύροι. Το πρωτόκολλο τρέχει μέχρι όλα τα μέλη να υπολογίσουν ξανά το νέο κλειδί του group.

Κατά τον πρώτο γύρο κάθε μέλος που παραμένει στο group ανανεώνει το δένδρο κλειδιών διαγράφοντας όλα τα μέλη και τους κόμβους πατέρες των μελών που αποκόπηκαν από το group εξαιτίας του partition. Η αναλυτική διαδικασία ανανέωσης του δένδρου κλειδιών μετά από ένα partition έχει ως εξής:

Όλοι οι κόμβοι που αποχώρησαν ταξινομούνται ανάλογα με το βάθος της θέσης που κατείχαν στο δένδρο. Η ταξινόμηση ξεκινά από το πλέον βαθύ επίπεδο, κάθε ζεύγος κόμβων αδελφών που αποχώρησαν ενσωματώνονται στον κόμβο πατέρα ο οποίος σημειώνεται σαν κόμβος που έχει αποχωρίσει από το group. Ο σημειωμένος πλέον κόμβος εισάγεται ξανά στην λίστα των κόμβων που έχουν αποχωρίσει από το

group. Η παραπάνω διαδικασία συνεχίζεται έως ότου δεν υπάρχουν άλλοι κόμβοι που να μπορούν να ενσωματωθούν στον κόμβο πατέρα τους. Το δένδρο που προκύπτει με την λήξη της παραπάνω διαδικασίας περιέχει έναν αριθμό κόμβων φύλλων που έχουν αποχωρίσει από το group, χαρακτηριστικό αυτών των κόμβων είναι ότι ο κόμβος αδερφός τους παραμένει ακόμα στο group. Στην συνέχεια για κάθε κόμβο που έχει αποχωρίσει από το group ορίζουμε ένα sponsor με βάση τα κριτήρια που ορίζαμε τους sponsors στο πρωτόκολλο αποχώρησης μέλους. Κάθε sponsor υπολογίζει τα κλειδιά και τα bkeys των κόμβων που η τρέχουσα δομή του δένδρου του επιτρέπει και στην συνέχεια αποστέλλει σε όλα τα μέλη του group το σύνολο των νέων bkeys. Με την λήψη του μηνύματος που απέστειλε ο sponsor τα μέλη του group ελέγχουν αν στο έλαβαν κάποια νέα bkeys. Η διαδικασία αυτή επαναλαμβάνεται έως ότου όλα τα μέλη του group να μπορούν να υπολογίσουν το κλειδί του group.

Για επιτευχθεί ανεξαρτησία κλειδιού, απαιτείται ένα τουλάχιστον μέλος από τα εναπομείναντα να αλλάξει το share του. Για το λόγο αυτό κατά την φάση του πρώτου γύρου του πρωτοκόλλου, ο χαμηλότερος και πλέον δεξιός sponsor παράγει ένα νέο share.



Στο παραπάνω σχήμα απεικονίζεται το παράδειγμα ανανέωσης του μυστικού κλειδιού ενός group μετά από ένα partition. Όπως παρατηρούμε τα εναπομείναντα μέλη στο group διαγράφουν τους κόμβους στους οποίους συνδέονταν τα μέλη που αποχώρησαν και υπολογίζουν τα κλειδιά και τα bkeys κατά την διάρκεια του πρώτου γύρου.

Ο αριθμός των γύρων που απαιτούνται για την ολοκλήρωση της διαδικασίας υπολογισμού του νέου κλειδιού είναι συνάρτηση δύο παραγόντων.

- Του αριθμού των μελών που αποχώρησαν

- ii. Του ύψους του δένδρου που προκύπτει μετά την διαγραφή των μελών που αποχώρησαν.

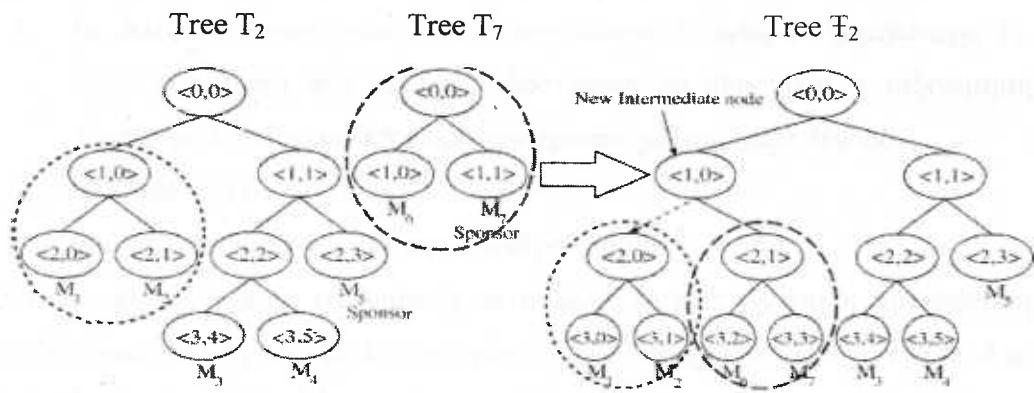
## Merge Protocol

Σκοπός του merge πρωτοκόλλου είναι η συνένωση groups που δημιουργήθηκαν εξαιτίας κάποιου partition σε κάποιο αρχικό group.

Έστω ότι ένα group εξαιτίας αποτυχίας του δικτύου διαμελίστηκε σε k μέρη. Για την συνένωση των k groups που προέκυψαν ως αποτέλεσμα του partition στο αρχικό group εκτελείται το merge πρωτόκολλο. Κατά τον πρώτο γύρο της εκτέλεσης ο sponsor κάθε group (το πλέον δεξιό μέλος κάθε group) αποστέλλει σε όλα τα άλλα groups το δένδρο του συμπληρωμένο με τα bkeys. Στην συνέχεια ανανεώνει το share του και υπολογίζει όλα τα κλειδιά και τα bkeys των κόμβων που ανήκουν στο μονοπάτι από τον κόμβο πατέρα του έως την ρίζα του δένδρου. Με την λήψη των μηνυμάτων που έστειλαν οι sponsors όλα τα μέλη μπορούν να διαμορφώσουν ένα νέο δένδρο κλειδιών που θα συνενώνει τα k groups σύμφωνα με την πολιτική διαχείρισης δένδρου που θα περιγράψουμε στην συνέχεια.

Μετά την ανανέωση του δένδρου κάθε sponsor υπολογίζει όλα τα κλειδιά και τα bkeys των κόμβων που ανήκουν στο μονοπάτι από τον κόμβο πατέρα του μέχρι την ρίζα του δένδρου, αν κάποιος sponsor δεν καταφέρει να υπολογίσει το κλειδί κάποιου ενδιάμεσου κόμβου διακόπτει την λειτουργία του (blocked). Οι sponsors που κατάφεραν να υπολογίσουν όλα τα κλειδιά και τα bkeys στο προηγούμενο βήμα αποστέλλουν σε όλα τα μέλη του group την νέα όψη του δένδρου τους. Με την λήψη των μηνυμάτων όλα τα μέλη ανανεώνουν την όψη του δένδρου τους. Αν ένας sponsor που απέστειλε την νέα όψη του δένδρου του στο group, έχει υπολογίσει και το κλειδί του κόμβου ρίζας, τότε με την λήψη του μηνύματός του όλα τα μέλη θα μπορούν να υπολογίσουν το κλειδί του κόμβου ρίζας. Στην γενικότερη περίπτωση που κάτι τέτοιο δεν συμβεί η αποστολή των μηνυμάτων των sponsors θα “ξεμπλοκάρουν” κάθε φορά κάποιον από τους blocked sponsors μέχρι κάποιος sponsor να υπολογίσει το κλειδί του κόμβου ρίζας.

Ένα παράδειγμα συνένωσης δύο groups απεικονίζεται στο παρακάτω σχήμα.



### Διαχείριση δένδρου

Η modular εκθετοποιήση είναι υπολογιστικά η πλέον “ακριβή” διαδικασία στο TGDH πρωτόκολλο. Ο αριθμός των εκθετοποιήσεων που απαιτούνται για την ανανέωση του κλειδιού του group μετά από κάποια αλλαγή στο membership εξαρτάται από την τρέχουσα δομή του δένδρου και την απόσταση του κόμβου στον οποίο συνδέεται ο sponsor από την ρίζα του δένδρου. Αν η απόσταση είναι 1 το join και το leave πρωτόκολλο απαιτούν 31 modular εκθετοποιήσεις. Προκειμένου να μεγιστοποιηθεί η απόδοση του πρωτοκόλλου ο sponsor πρέπει να βρίσκεται όσο το δυνατόν πλησιέστερα στον κόμβο ρίζα του δένδρου. Ένας άλλος στόχος της διαχείρισης του δένδρου κλειδιών είναι να διατηρείται το δένδρο όσο το δυνατόν ποιο ζυγισμένο γίνεται.

### Πολιτικές διαχείρισης του δένδρου στις περιπτώσεις εισόδου και αποχώρησης

Η πολιτική που ακολουθεί το TGDH προκειμένου το δένδρο να παραμένει ζυγισμένο, είναι να επιλέγεται σαν κόμβος εισόδου στην περίπτωση εισόδου νέου μέλους ή την περίπτωση συνένωσης groups ο ποίο απομακρυσμένος από την ρίζα πλέον δεξιός κόμβος. Η υιοθέτηση αυτής της πολιτικής δεν αυξάνει συνήθως το ύψος του δένδρου. Στην περίπτωση της αποχώρησης ή του partition το TGDH δεν υλοποιεί κάποιο σχήμα εξισορρόπησης του δένδρου.

Στο σημείο αυτό θα κάνουμε μία αναλυτική περιγραφή της πολιτικής διαχείρισης δένδρου, στην περίπτωση συνένωσης groups. Θεωρούμε την είσοδο νέου μέλους υποπερίπτωση της συνένωσης groups. Η πολιτική εξασφαλίζει την ανανέωση του δένδρου κλειδιών ανεξάρτητα, ταυτόχρονα και με συνέπεια από κάθε μέλος του group.

Αν θεωρήσουμε το παράδειγμα της συνένωσης ενός group που έχει διαμελιστεί σε k μέρη, τότε τα βήματα για την ανακατασκευή του δένδρου είναι τα ακόλουθα:

1. Τα δένδρα ταξινομούνται από το υψηλότερο  $T_1$  προς το χαμηλότερο  $T_k$ .  
Στην περίπτωση που πολλά δένδρα έχουν το ίδιο ύψος η ταξινόμηση γίνεται με λεξικογραφική σειρά του πρώτου μέλους κάθε δένδρου.
2. Εάντομε  $T=T_1$
3. Από  $i=2$  έως k συγχωνεύεται το δένδρο  $T$  με το  $T_i$

Ένα σημείο που πρέπει να διασαφηνιστεί είναι το πώς γίνεται η συγχώνευση μεταξύ δύο δένδρων. Αν τα δύο δένδρα έχουν το ίδιο ύψος εισάγουμε το ένα δένδρο στον κόμβο ρίζα του άλλου. Προκειμένου το τελικό δένδρο να είναι κοινό για όλα τα μέλη η συγχώνευση γίνεται βάση της ταξινόμησης που έχει προηγηθεί (βήμα 1) συγκεκριμένα το δέντρο που έπειται στην ταξινόμηση ενσωματώνεται σε αυτό που προηγείται. Το σημείο εισόδου του δένδρου είναι όπως έχουμε ήδη πει το χαμηλότερο πλέον δεξιό στο δένδρο. Στην περίπτωση που το δένδρο είναι πλήρως ζυγισμένο σημείο εισόδου είναι ο ρίζα του δένδρου.

## Επιλογή Sponsor

Η επιλογή του sponsor στο TGDH γίνεται σε κάθε γύρο εκτέλεσης του πρωτοκόλλου. Η επιλογή του sponsor για τον αρχικό γύρο γίνεται ως έξης:

- Στην περίπτωση προσθήκης μέλους στο group, το μέλος που συσχετίζεται με τον χαμηλότερο πλέον δεξιό κόμβο φύλλο του δένδρου γίνεται sponsor
- Στην περίπτωση αποχώρησης μέλους από το group, το μέλος που συσχετίζεται με τον χαμηλότερο πλέον δεξιό κόμβο φύλλο του κόμβου αδερφού του μέλους που αποχώρησε γίνεται sponsor. Στην περίπτωση του partition μπορεί να υπάρχουν περισσότεροι από έναν sponsors.

Στους επόμενους γύρους εκτέλεσης του πρωτοκόλλου σαν sponsor επιλέγεται πάντα το χαμηλότερο πλέον δεξιό φύλλο του κόμβου που τυγχάνει της έλλειψης ενός bkey.

Ο ρόλος του sponsor στο TGDH είναι τριπλός

1. Ανανεώνει το share του
2. Υπολογίζει όλα τα κλειδιά και τα bkeys στο μονοπάτι από τον κόμβο πατέρα του έως την ρίζα του δένδρου εφ' όσον κάτι τέτοιο είναι εφικτό
3. Αποστέλλει την όψη του δένδρου που κατέχει σε όλα τα τρέχοντα μέλη του group.



## Cliques toolkit – Μία σουίτα πρωτοκόλλων διαχείρισης του κλειδιού ενός group

To clique [61] είναι ένα project πάνω στην ασφάλεια των group communication συστημάτων. Το clique toolkit υλοποιήθηκε στα πλαίσια του παραπάνω project και παρέχει ένα σύνολο πρωτοκόλλων για την εγκαθίδρυση και διαχείριση του μυστικού κλειδιού συνόδου ενός group. Όλα τα πρωτόκολλα που υλοποιούνται στο clique στηρίζονται στο πρωτόκολλο σύναψης μυστικού κλειδιού Diffie-Hellman και εξασφαλίζουν αυθεντικοποίηση του κλειδιού. Όλα τα πρωτόκολλα του clique, έκτος από το Centralized key distribution – CDK ακολουθούν κατανεμημένες συνεργατικές τεχνικές για την σύναψη και διαχείριση του μυστικού κλειδιού, επιπλέον ικανοποιούν τις αρχές του key independence και perfect forward secrecy. Τα πρωτόκολλα που υλοποιεί το clique είναι τα:

- STR [62]
- Burmester- Desmedt – BD [63]
- Group Diffie – Hellman – GDH [64]
- Centralized group key distribution – CDK (κεντρικοποιημένο)
- Tree-Based Group Diffie-Hellman – TGDH στηρίζεται στο TGDH πρωτόκολλο που περιγράψαμε παραπάνω.

To clique toolkit είναι open source, για την λειτουργία του απαιτείται η εγκατάσταση του OpenSSL και ενός group communication συστήματος που θα εξασφαλίζει:

- Δυνατότητα αποστολής μηνυμάτων που θα προς όλα τα μέλη (group multicast)
- Δυνατότητα αποστολής μηνυμάτων από μέλος του group σε άλλο μέλος του group (group member to group member unicast)
- Ορθή σειρά μετάδοσης των μηνυμάτων (FIFO ordering on messages)
- Δυνατότητα γνώσης των μελών του group (current membership)

Στον παρακάτω πίνακα μπορούμε να δούμε τα υπολογιστικά και επικοινωνιακά κόστη καθενός από τα κατανεμημένα πρωτόκολλα που υλοποιεί το clique. Όπου n ο αριθμός των μελών του group, m ο αριθμός των μελών που θέλουν να συνενωθούν με το group, k ο αριθμός των υπό συνένωση groups και p ο αριθμός των μελών που αποχωρούν από το group.

		Communication		Computation		
		Rounds	Messages	Mod exps	Signatures	Verifications
GDH	Join	4	$n + 3$	$n + 3$	4	$n + 3$
	Leave	1	1	$n - 1$	1	1
	Merge	$m + 3$	$n + 2m + 1$	$n + 2m + 1$	$m + 3$	$n + 2m + 1$
	Partition	1	1	$n - p$	1	1
TGDH	Join	2	3	$3h - 3$	2	3
	Leave	1	1	$3h - 3$	1	1
	merge	$\lceil \log_2 k \rceil + 1$	$2k$	$3h - 3$	$\lceil \log_2 k \rceil + 1$	$\lceil \log_2 k \rceil$
	Partition	$p$	$\min(2p, \lceil \frac{n}{2} \rceil)$	$3h - 3$	$p$	$\min(2p, \lceil \frac{n}{2} \rceil)$
STR	Join	2	3	4	2	3
	Leave	1	1	$\frac{3n}{2} + 2$	1	1
	Merge	2	$k + 1$	$3m + 1$	2	3
	Partition	1	1	$\frac{3n}{2} + 2$	1	1
BD	Join	2	$2n + 2$	3	2	$n + 3$
	Leave	2	$2n - 2$	3	2	$n + 1$
	Merge	2	$2n + 2m$	3	2	$n + m + 2$
	Partition	2	$2n - 2p$	3	2	$n - p + 2$

## Συμπεράσματα

Στις προηγούμενες ενότητες παρουσιάσαμε κεντρικοποιημένα, μη κεντρικοποιημένα και κατανεμημένα σχήματα για την εγκαθίδρυση και διαχείριση του μυστικού συνόδου, σε ένα group communication σύστημα.

Τα κεντρικοποιημένα πρωτόκολλα κάνουν χρήση μίας έμπιστης τρίτης οντότητας για την κατασκευή και διανομή του μυστικού κλειδιού. Η χρήση κεντρικοποιημένων σχημάτων εκτός του ότι παραβιάζει την κατανεμημένη φύση των group communication συστημάτων δημιουργεί επιπλέον ένα σημείο συμφόρησης και αποτυχίας. Τα κεντρικοποιημένα σχήματα έχουν περιορισμένη επεκτασιμότητα μίας και ο αριθμός των μελών που μπορεί η οντότητα που διαχειρίζεται το μυστικό κλειδί να εξυπηρετήσει είναι περιορισμένος.

Τα μη κεντρικοποιημένα σχήματα κάνουν χρήση περισσότερων από μία έμπιστων τρίτων οντοτήτων. Ο αριθμός τους εξαρτάται από τον αριθμό των μελών του group. Ένα βασικό πλεονέκτημα των σχημάτων που ακολουθούν μη κεντρικοποιημένες αρχιτεκτονικές είναι η επεκτασιμότητα τους. Στα σχήματα αυτά, αποτυχία ενός αριθμού έμπιστων τρίτων οντοτήτων δεν συνεπάγεται και κατάρρευση ολόκληρου του συστήματος. Παρ' όλα αυτά το γεγονός ότι η εγκαθίδρυση και διαχείριση του κλειδιού του group γίνεται κεντρικά παραβιάζει την κατανεμημένη φύση των group communication συστημάτων.

Κάποια κοινά μειονεκτήματα τόσο των κεντρικοποιημένων όσο και των μη κεντρικοποιημένων σχημάτων που τα κάνουν ακατάλληλα για τα group communication συστήματα είναι τα ακόλουθα.

- Το γεγονός ότι τα μυστικά του group παράγονται σε ένα ή μικρό αριθμό σημείων καθιστά τα σημεία αυτά δελεαστικό στόχο παραβίασης πιθανών επίβουλων
- Τα συστήματα που δεν ακολουθούν μία ιεραρχική δομή της εμπιστοσύνης δεν συμβαδίζουν με την λογική της κεντρικής διαχείρισης του κλειδιού συνόδου.
- Στα συστήματα που ακολουθούν την λογική διανομής του κλειδιού συνόδου και όχι της αμοιβαίας δημιουργίας αυτού, τα μέλη δεν μπορεί να είναι σίγουρα κατά πόσο το κλειδί αυτό είναι πρόσφατο και έχει επιλεχθεί με τυχαίο τρόπο.
- Η επίτευξη της αρχής perfect forward secrecy και η ικανότητα του πρωτοκόλλου να αντιστέκεται σε επιθέσεις του τύπου know – key με αποτελεσματικό τρόπο είναι ιδιαίτερα δύσκολη στα πρωτόκολλα που δεν ακολουθούν μία κατανεμημένη αρχιτεκτονική.

Γίνεται φανερό ότι η χρήση πρωτοκόλλων που κάνουν χρήση μίας ή περισσοτέρων έμπιστων τρίτων οντοτήτων δεν ενδείκνυται για την διαχείριση του μυστικού κλειδιού συνόδου ενός group. Σαν πλέον κατάλληλα θεωρούνται τα κατανεμημένα πρωτόκολλα που δημιουργούν το κλειδί συνόδου με ένα συνεργατικό τρόπο. Ένα μειονέκτημα των κατανεμημένων πρωτοκόλλων είναι το μεγάλο επικοινωνιακό και υπολογιστικό τους κόστος. Το κόστος των κατανεμημένων πρωτοκόλλων είναι ανάλογο του μεγέθους του group, το γεγονός αυτό δεν αποτελεί τροχοπέδη, στην χρήση των κατανεμημένων πρωτοκόλλων διαχείρισης κλειδιού στα group communication συστήματα, μία και τις περισσότερες φορές τα group είναι μικρού ή μεσαίου μεγέθους (της τάξης κάποιων εκατοντάδων χρηστών). Εξάλλου η χρήση κατάλληλων αρχιτεκτονικών από το υποκείμενο επικοινωνιακό σύστημα μπορεί να δώσει μερική λύση σ' αυτό το πρόβλημα. Η επιλογή ενός κατανεμημένου πρωτοκόλλου είναι μία ιδιαίτερη δύσκολη διαδικασία. Τα κριτήρια βάση των οποίων θα επιλέξουμε ένα πρωτόκολλο είναι τα ακόλουθα:

- Αν το πρωτόκολλο εξασφαλίζει τις έξι βασικές αρχές που περιγράψαμε στην αρχή της ενότητας

- Αν εξασφαλίζει αυθεντικοποίηση του μυστικού κλειδιού συνόδου
- Το επικοινωνιακό (αριθμός multicast και unicast μηνυμάτων) και υπολογιστικό (αριθμός modular εκθετοποιήσεων) του κόστος
- Η απόδοση του, που μπορεί να μετρηθεί σε χρόνο εγκαθίδρυσης και διαχείρισης (χρόνοι για αλλαγή κλειδιού στα join, leave, partition, merge events) του κλειδιού συνόδου.
- Η αποδοχή του από την επιστημονική κοινότητα. Στο σημείο αυτό πρέπει να δοθεί ιδιαίτερη προσοχή κατά την επιλογή ενός πρωτοκόλλου. Τα περισσότερα πρωτόκολλα συνοδεύονται από μία ανάλυση ασφάλειας που έχουν κάνει οι ίδιοι οι δημιουργοί τους. Σε πολλά πρωτόκολλα η ανάλυση ασφάλειας αδυνατεί να αποκαλύψει όλες τις αδυναμίες τους. Ένα πρωτόκολλο που έχει κεντρίσει το ενδιαφέρον της επιστημονικής κοινότητας και η ασφάλεια του έχει εξεταστεί διεξοδικά αποτελεί μία ποιο αξιόπιστη επιλογή.

### Ενότητα 3<sup>η</sup>: Συστήματα ασφαλούς επικοινωνίας ομάδας

Μέχρι τώρα έχουμε περιγράψει πρωτόκολλα αυθεντικοποίησης, διανομής μυστικού κλειδιού συνόδου και πρωτόκολλα ελέγχου πρόσβασης και δικαιωμάτων σε ένα group. Για να κατανοήσουμε όμως πως τα πρωτόκολλα αυτά μπορούν να συσχετιστούν και να αποτελέσουν ένα ολοκληρωμένο σύστημα ασφαλούς επικοινωνίας ενός group θα παρουσιάσουμε τα δημοφιλέστερα συστήματα που έχουν υλοποιηθεί τα τελευταία χρόνια και μπορούν να χρησιμοποιηθούν τόσο σε τοπικά δίκτυα (LAN) όσο και σε δίκτυα ευρύτερης περιοχής (WAN).

Τα συστήματα αυτά είναι τα:

- Antigone
- Secure Spread
- Ensemble
- Secure Group Layer (SGL)

Τα παραπάνω συστήματα δεν πραγματεύονται τις εκ' τον έσω απελές σε ένα group communication σύστημα. Αν και στην παρούσα εργασία θεωρούμε ότι όλα τα μέλη που συμμετέχουν σε ένα group είναι μη διεφθαρμένα και κατά συνέπεια δεν θα υπονομεύσουν την ασφάλεια του συστήματος, για λόγους πληρότητας θα

αφιερώσουμε μία ενότητα για την περιγραφή τέτοιου είδους απειλών και θα παρουσιάσουμε το σύστημα Secure Ring που υλοποιεί μηχανισμούς για την αντιμετώπιση τους.

## ***Antigone***

To Antigone είναι ένα σύστημα για ασφαλή επικοινωνία ενός group. Δεν προϋποθέτει IP multicast σύστημα μετάδοσης αλλά μπορεί να εκμεταλλευτεί αποτελεσματικά τις δυνατότητες ενός τέτοιου συστήματος.

Σκοπός του Antigone είναι:

- Η υποστήριξη ευέλικτων πολιτικών ασφάλειας
- Η υποστήριξη ευέλικτων μοντέλων απειλών (threat model)
- Ανεξαρτησία από τον μηχανισμό μετάδοσης
- Ανεξαρτησία από την υποδομή ασφάλειας
- Απόδοση

To Antigone θεωρεί ότι όλα τα μέλη ενός group είναι αξιόπιστα και δεν έχουν σκοπό να υπονομεύσουν την ασφάλεια του συστήματος.

Επειδή κάθε εφαρμογή μπορεί να έχει διαφορετικό μοντέλο εμπιστοσύνης και διαφορετικές απαιτήσεις απόδοσης, το σύστημα Antigone παρέχει ένα βασικό σύνολο μηχανισμών που μπορούν να χρησιμοποιηθούν για την υλοποίηση μιας μεγάλης ποικιλίας πολιτικών ασφάλειας.

### ***Πολιτικές και Αρχιτεκτονική του Antigone***

Οι μηχανισμοί του Antigone μπορούν να διακριθούν σε τέσσερις ομάδες. Η διάκριση γίνεται βάση του μέρους της συνολικότερης πολιτικής την οποία υλοποιούν.

Οι πολιτικές είναι:

- Πολιτική ανανέωσης κλειδιού συνόδου (session rekeying policy)
- Πολιτική μηνυμάτων επιπέδου εφαρμογής (application message policy)
- Πολιτική ενημέρωσης σχετικά με τα τρέχοντα μέλη του group (membership awareness policy)
- Πολιτική διαχείρισης αποτυχιών (failure policy)

Η πολιτική σχετικά με την αναθεώρηση του μυστικού κλειδιού του group είναι ιδιαίτερα σημαντική στα group communication συστήματα. Γενικότερα μπορεί να λεχθεί ότι ορισμένα γεγονότα που συμβαίνουν σε ένα GCS έχουν σαν αποτέλεσμα την ενεργοποίηση των μηχανισμών αναθεώρησης κλειδιού.

Τα γεγονότα αυτά είναι:

- Είσοδος νέου χρήστη
- Αποχώρηση χρήστη
- Αποτυχία εφαρμογής
- Αποβολή μέλους από το group

Όπως έχουμε ήδη αναφέρει το σύστημα Antigone δίνει την δυνατότητα υλοποίησης membership policies. Μία membership policy σε ένα GCS εξασφαλίζει την δυνατότητα όλων των μελών να έχουν πληροφορίες σχετικά με το membership. Επειδή η υποστήριξη μιας τέτοιας πολιτικής συνεπάγεται μεγάλο κόστος, είναι στην ευκαίρια της εφαρμογής αν θα την υλοποιήσει ή όχι ανάλογα με το αν η εφαρμογή έχει μεγαλύτερη ανάγκη από ασφάλεια ή απόδοση.

Όσον αφορά την πολιτική σχετικά με την ανοχή σε λάθη, το Antigone δεν υποστηρίζει μηχανισμούς για επιθέσεις του τύπου denial of service και Byzantine failures.

Ανάλογα με το thread model της κάθε εφαρμογής ένα από τα παραπάνω γεγονότα (είσοδος, αποχώρηση μέλους κ.τ.λ.) μπορεί να οδηγήσουν στην αναθεώρηση του κλειδιού. Η αλλαγή του κλειδιού συνόδου ενός GCS μπορεί να γίνεται και περιοδικά προκειμένου να αντιμετωπιστούν απειλές κρυπτανάλυσης.

Το σύστημα Antigone υποστηρίζει μηχανισμό αναθεώρησης του κλειδιού συνόδου χωρίς την χρήση του τρέχοντος κλειδιού συνόδου. Με τον τρόπο αυτό χρήστες που έφυγαν από το group ή απομακρύνθηκαν από τα άλλα μέλη του group, δεν έχουν πρόσβαση στο κλειδί συνόδου του νέου view.

Το Antigone με την χρήση του κλειδιού συνόδου υποστηρίζει:

- Ακεραιότητα (integrity)
- Εμπιστευτικότητα (confidentiality)
- Αυθεντικότητα του group (group authenticity)
- Αυθεντικότητα του αποστολέα (sender authenticity)

## Αρχιτεκτονική του Antigone

Το σύστημα Antigone έχει μία αρχιτεκτονική τριών επιπέδων:

- Broadcast transport layer
- Mechanism layer
- Predefined policy layer

Το mechanism layer ακολουθεί μία modular αρχιτεκτονική. Η επιλογή αυτής της αρχιτεκτονικής δίνει την δυνατότητα εύκολων μελλοντικών τροποποιήσεων. Το mechanism layer περιέχει τους μηχανισμούς που ευθύνονται για:

- Αυθεντικοποίηση (authentication)
- Εισαγωγή νέου μέλους (member join)
- Διανομή κλειδιού συνόδου και διανομή πληροφοριών για τα μέλη του group (session key and group membership distribution)
- Διαδικασία χειρισμού μηνυμάτων που αποστέλλονται από τις εφαρμογές (application messaging)
- Ανίχνευση αποτυχιών (failure detection)
- Αποχώρηση μέλους (member leave)

Στο σημείο αυτό θα δώσουμε μία σύντομη περιγραφή των ορολογιών που χρησιμοποιούνται στο Antigone προκειμένου να γίνει ποιο κατανοητός ο τρόπος με τον οποίο υλοποιούνται οι δυνατότητες του συστήματος.

### Session leader

Κάθε group έχει ένα μοναδικό αρχιγό του οποίου η ταυτότητα είναι γνωστή σε όλους και ονομάζεται session leader (SL)

### Εμπιστη τρίτη οντότητα

Στο Antigone γίνεται χρήση μιας έμπιστης τρίτης οντότητας (trusted third party - TTP) που παρέχει τους κατάλληλους μηχανισμούς στον SL για την αυθεντικοποίηση τα υποψήφια μέλη του group.

### Περιγραφή λειτουργίας

Κατά την έναρξη της διαδικασίας αυθεντικοποίησης ο SL στέλνει σε όλα τα μέλη την πολιτική του group. Ο SL χρησιμοποιεί ένα ζεύγος ιδιωτικού δημόσιου κλειδιού,

με σκοπό την αποστολή μηνυμάτων του τύπου “I’m alive” με οικονομικό τρόπο. Η χρήση κρυπτογραφίας δημοσίου κλειδιού δεν απαιτεί την ύπαρξη υποδομής δημοσίου κλειδιού μίας και τα πιστοποιητικά που χρησιμοποιούνται δημιουργούνται από τον SL, την ταυτότητα του οποίου μπορεί να ελέγξει κάθε μέλος με την βοήθεια του TTP που περιγράψαμε παραπάνω.

Το σύστημα Antigone για την αυθεντικοποίηση των μελών του και την διανομή του κλειδιού συνόδου χρησιμοποιεί τον Leighton – Micali αλγόριθμο [9]. Η επιλογή του έγινε με γνώμονα την απόδοση, ο αλγόριθμος κάνει χρήση συμμετρικής κρυπτογραφίας.

Κάθε πιθανό μέλος και ο SL μοιράζονται με την TTP ένα συμμετρικό κλειδί μεγάλης διάρκειας. Ένα υποψήφιο μέλος αρχικοποιεί την διαδικασία αυθεντικοποίησης στέλνοντας ένα μήνυμα στον SL που περιέχει την ταυτότητα του. Ο SL λαμβάνει από την TTP ένα κλειδί που θα διαμοιράζονται από κοινού ο SL και το υποψήφιο μέλος. Προκειμένου να αποφευχθεί η απειλή επανάληψης χρησιμοποιούνται nonces και όχι χρονοσήμανση στα μηνύματα. Η δημιουργία του μυστικού κλειδιού SL – υποψηφίου μέλους γίνεται με τέτοιο τρόπο ώστε ο TTP να χρειάζεται να το αποστείλει μόνο στον SL ενώ το υποψήφιο μέλος να μπορεί να το υπολογίσει μόνο του.

Μετά την απόκτηση του κοινού μυστικού κλειδιού ο SL στέλνει στο πιθανό μέλος ένα μήνυμα απάντησης. Το μήνυμα περιέχει τις ταυτότητες του SL και του μέλους μη κρυπτογραφημένα, το τρέχον view του group, τον προσδιοριστή (identifier) του group, τα απαραίτητα nonces, την πολιτική ασφάλειας και το δημόσιο κλειδί του group κρυπτογραφημένα με το κοινό τους μυστικό κλειδί.

Ο μηχανισμός σύνδεσης (*join mechanism*) δίνει την δυνατότητα σε ένα αυθεντικοποιημένο μέλος να αποκτήσει πρόσβαση σε ένα group. Η διαδικασία αρχίζει με την αποστολή ενός μηνύματος του υποψήφιου μέλους που περιέχει την ταυτότητά του και ένα nonce που είχε λάβει από τον SL κατά την αυθεντικοποίηση, κρυπτογραφημένα με το μυστικό κλειδί SL – υποψηφίου μέλους, στον SL. Αν το nonce είναι έγκυρο τότε το μέλος γίνεται αποδεκτό στο group.

Όλα τα μηνύματα του Antigone περιέχουν τον προσδιοριστή του group (group identifier), τον αριθμό σειράς (sequence number) του τελευταίου SL και τον κωδικό αυθεντικότητας του μηνύματος (message authentication code – MAC) ο οποίος

υπολογίζεται με την συνάρτηση σύνοψης MD5 πάνω σε όλο το μήνυμα. Ο MAC εξασφαλίζει την ακεραιότητα του μηνύματος.

Τα μηνύματα που αποστέλλονται από τον SL για την αλλαγή κλειδιού είναι κρυπτογραφημένα με το μυστικό κλειδί του SL – μέλους και όχι με το παλιό μυστικό κλειδί συνόδου προκειμένου μέλη που δεν συμμετέχουν στο νέο view αλλά γνώριζαν το παλιό κλειδί συνόδου να μην έχουν πρόσβαση στο νέο κλειδί.

Όπως έχουμε ήδη αναφέρει ο *application messaging mechanism* αναλαμβάνει την μετάδοση μηνυμάτων επιπέδου εφαρμογής. Τα μηνύματα επιπέδου εφαρμογής κρυπτογραφούνται με το μυστικό κλειδί συνόδου. Η ακεραιότητα των μηνυμάτων επιτυγχάνεται με την χρήση MACs με χρήση του MD5 και του μυστικού κλειδιού συνόδου. Η αυθεντικοποίηση του μέλους που στέλνει ένα μήνυμα δεν είναι δυνατόν να υποστηριχθεί με την τρέχουσα έκδοση του Antigone κάτι τέτοιο όμως θα μπορούσε να επιτευχθεί με την χρήση ψηφιακών υπογραφών που στηρίζονται στη χρήση πιστοποιητικών και κρυπτογραφίας δημοσίου κλειδιού.

Το σύστημα Antigone έχει υλοποιημένο ένα μηχανισμό για την αντιμετώπιση αποτυχιών που μπορεί να οφείλονται είτε σε προβλήματα του μέσου μετάδοσης είτε σε εισβολείς. Ο μηχανισμός αυτός στηρίζεται στην αποστολή μηνυμάτων που διαβεβαιώνουν ότι ο SL ή το κάθε μέλος είναι “ζωντανό” (heartbeat message). Κάθε μέλος στέλνει στον SL ένα heartbeat μήνυμα κρυπτογραφημένο με το μυστικό κλειδί μέλους – SL, όταν κάποιος αριθμός μηνυμάτων (ο αριθμός εξαρτάται από την πολιτική της εφαρμογής) δεν φτάσει στον SL το μέλος θεωρείται ότι έχει αποτύχει και απομακρύνεται από το group. Τα μέλη μπορούν να διαπιστώσουν αν ο SL είναι “ζωντανός” από τα heartbeat μηνύματα που αποστέλλει, αν κάποιος αριθμός heartbeat δεν φτάσει στους παραλήπτες ο SL θεωρείται ότι έχει αποτύχει. Στα heartbeat μηνύματα του SL η ταυτότητα του SL αποδεικνύεται με χρήση της ψηφιακής του υπογραφής μέσo του ιδιωτικού του κλειδιού και χρήση κρυπτογραφίας δημοσίου κλειδιού.

Τα heartbeat μηνύματα εξυπηρετούν διπλό σκοπό. Έκτος από την ανίχνευση αποτυχιών του συστήματος τα μέλη μπορούν επιπρόσθετα να γνωρίζουν αν έχουν την ποιο πρόσφατη κατάσταση του group. Και κατά συνέπεια αν χρησιμοποιούν το ποιο πρόσφατο μυστικό κλειδί συνόδου.

Στο πλαίσιο του *failure detection mechanism* ένα μέλος που απέτυχε να λάβει πληροφορία για το membership ή για το τρέχον κλειδί συνόδου μπορεί να στείλει στον SL ένα μήνυμα για την επαναποστολή της ζητούμενης πληροφορίας.

Πρέπει να σημειώσουμε ότι η συμφόρηση που δημιουργείται από το χάσιμο μηνυμάτων αναθεώρησης του κλειδιού συνόδου ή heartbeat μηνυμάτων γίνεται εκρηκτική σε μεγάλα groups ή σε δίκτυα με μεγάλο ποσοστό απώλειας μηνυμάτων. Για την αντιμετώπιση αυτού του προβλήματος το σύστημα Antigone σκοπεύει να κάνει μελλοντικά χρήση μηχανισμού επαναποστολής όμοιο του SRM [10]. Με την υιοθέτηση μίας τέτοιας προσέγγισης το κόστος επαναποστολής διαμοιράζεται μεταξύ των μελών.

Το Antigone υλοποιεί ένα μηχανισμό αποχώρησης. Ο μηχανισμός αυτός μπορεί να χρησιμοποιηθεί από ένα μέλος για να δηλώσει την αποχώρηση του από το group αλλά και για να ζητήσει την αποβολή ενός μέλους του group. Όταν ο SL λάβει ένα μήνυμα αποβολής, αποφασίζει με βάση την πολιτική του group αν θα αποβάλει το μέλος.

## Υλοποίηση πολιτικών στο Antigone

Ένα από τα βασικότερα πλεονεκτήματα του Antigone είναι η δυνατότητα που δίνει στην εφαρμογή να επιλέξει την δική της πολιτική ασφάλειας.

Η πολιτική αναθεώρησης του κλειδιού συνόδου σχετίζεται με το ποία γεγονότα σε ένα group θα οδηγήσουν σε αναθεώρηση του κλειδιού. Οι πολιτικές αναθεώρησης του κλειδιού είναι:

- Time – sensitive πολιτική. Το κλειδί συνόδου αλλάζει μετά την πάροδο συγκεκριμένου χρονικού διαστήματος.
- Join – sensitive πολιτική. Αλλαγή κλειδιού γίνεται με την είσοδο νέου μέλους.
- Leave – sensitive πολιτική. Αλλαγή κλειδιού με την αποχώρηση, αποβολή ή αποτυχία ενός μέλους.
- Membership – sensitive πολιτική. Αλλαγή του κλειδιού μετά την είσοδο, αποχώρηση, αποβολή ή αποτυχία μέλους.

Κάθε εφαρμογή μπορεί να δημιουργήσει πολιτική με συνδυασμό των παραπάνω πολιτικών ανάλογα με τις ανάγκες της.

Το Antigone παρέχει διεπαφή στην οποία μπορεί κάποιος να επιλέξει μεταξύ προκαθορισμένων πολιτικών. Καθώς και διεπαφές μέσω των οποίων μπορεί να

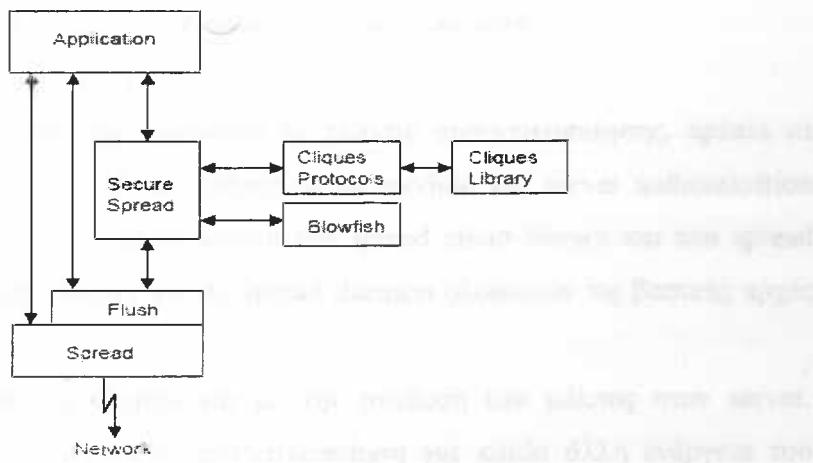
παραμετροποιήσει τις μεταβλητές των πολιτικών ασφάλειας, για παράδειγμα μπορεί να επιλέξει το χρόνο αναμονής heartbeats ή το χρόνο αναθεώρησης κλειδιού σε μία time – sensitive υλοποίηση του συστήματος.

## Secure Spread

To secure spread είναι ένα σύστημα που υποστηρίζει αξιόπιστη multicast και group επικοινωνία για LAN και WAN. Παρέχει αξιόπιστη μετάδοση μηνυμάτων και σωστή σειρά παράδοσης και υποστηρίζει τις αρχές του extended virtual synchrony και view synchrony. To secure spread ακολουθεί μια client – server αρχιτεκτονική.

### Αρχιτεκτονική του Secure Spread

To secure spread ακολουθεί μία modular αρχιτεκτονική, το σύστημα αποτελείται από components κάθε ένα από τα οποία επιτελεί και μία διαφορετική λειτουργία. Η βάση του συστήματος είναι το σύστημα spread που εξασφαλίζει την επικοινωνία μεταξύ των μελών του group, παρέχοντας τις δυνατότητες που αναφέραμε παραπάνω. Η αρχιτεκτονική του secure spread απεικονίζεται στο παρακάτω σχήμα.

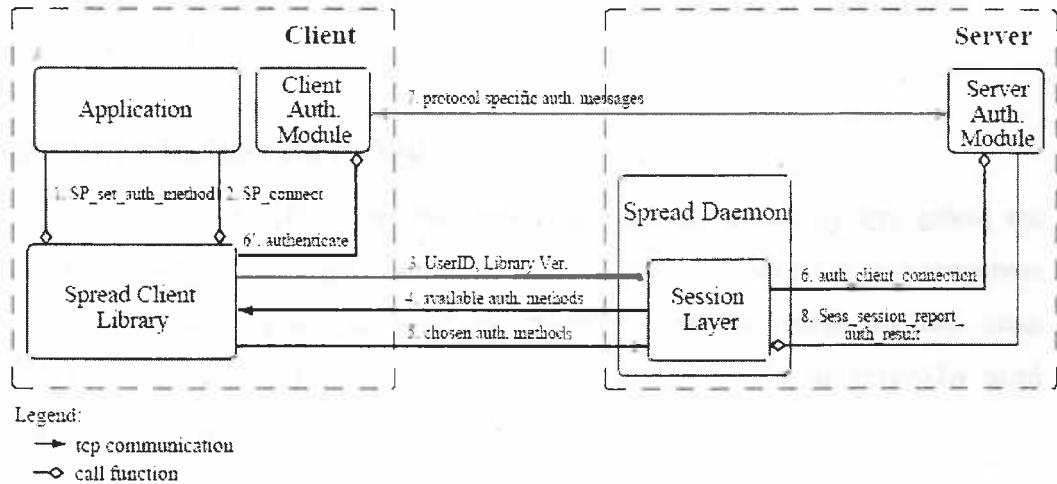


### Αυθεντικοποίηση και έλεγχος πρόσβασης στο Secure Spread

To secure spread δεν έχει υλοποιήσει κάποιο μηχανισμό αυθεντικοποίησης και ελέγχου πρόσβασης αλλά δίνει την δυνατότητα για την ενσωμάτωση ήδη υπαρχόντων μηχανισμών. Κυρίως μηχανισμών της μορφής "State – independent". Οι "State –

"independent" μηχανισμοί δεν απαιτούν την ύπαρξη γνώσης σχετικά με την τρέχουσα κατάσταση του group ή τα μηνύματα που στέλνονται στο group προκειμένου να πάρουν αποφάσεις αυθεντικοποίησης ή ελέγχου πρόσβασης. Το πλαίσιο αυθεντικοποίησης που υποστηρίζει το secure spread επιτρέπει στις εφαρμογές να εφαρμόσουν την δική τους πολιτική αυθεντικοποίησης και ελέγχου πρόσβασης.

Θα παρουσιάσουμε το πλαίσιο αυθεντικοποίησης του secure spread και θα εξηγήσουμε πως επιτυγχάνεται η αυθεντικοποίηση και ο έλεγχος πρόσβασης.



### Secure Spread Authentication framework

Για να χρησιμοποιήσει μία εφαρμογή το πλαίσιο αυθεντικοποίησης, πρέπει να δημιουργήσει το κατάλληλο client authentication module και server authentication module. Δεν απαιτείται όμως τροποποίηση του spread client library και του spread daemon. Το spread client library και το spread daemon υλοποιούν τις βασικές αρχές μετάδοσης.

Η αυθεντικοποίηση ολοκληρώνεται με την σύνδεση των μέλους στον server. Μετά την σύνδεση δεν απαιτείται αυθεντικοποίηση για καμία άλλη ενέργεια του μέλους στο σύστημα.

Το secure spread δεν υλοποιεί κάποια πολιτική πρόσβασης άλλα επιτρέπει στις εφαρμογές να υποστηρίζουν κάποια πολιτική προσαρμοσμένη στις επιμέρους ανάγκες τους. Ορισμένες μέθοδοι αυθεντικοποίησης και ελέγχου πρόσβασης που υποστηρίζει το spread και έχουν δοκιμασθεί στην πράξη είναι οι ακόλουθες:

- Πρόσβαση βασισμένη στην IP διεύθυνση (IP Access Control)

- Αυθεντικοποίηση με χρήση συνθηματικών (password authentication)
- Με την μέθοδο SecureID [11]
- Με την μέθοδο PAM [12]
- Ανώνυμη Αυθεντικοποίηση για συστήματα πληρωμών
- Αυθεντικοποίηση μέσω του group. Στη μέθοδο αυτή στην περίπτωση που ο server στον οποίο συνδέεται ένα μέλος δεν έχει πληροφορίες αυθεντικοποίησης για το συγκεκριμένο μέλος ζητάει από τους άλλους server του group να δώσουν πληροφορίες αυθεντικοποίησης για το υποψήφιο μέλος. Στην περίπτωση που έστω και ένας από τους server του group απαντήσουν θετικά, τότε το μέλος εισέρχεται στο group.

## Διαχείριση κλειδιού συνόδου

Η διαχείριση του κλειδιού συνόδου στο secure spread γίνεται με την χρήση της βιβλιοθήκης Clique. Το Clique toolkit όπως ήδη έχουμε αναφέρει έχει υλοποιήσει πέντε πρωτόκολλα διαχείρισης κλειδιού συνόδου, από τα οποία το ένα είναι κεντρικοποιημένο και τα υπόλοιπα τέσσερα κατανεμημένα. Τα πρωτόκολλα αυτά είναι τα:

- Centralized group key distribution – CDK
- Burmester- Desmedt – BD
- Group Diffie – Hellman – GDH
- Tree-Based Group Diffie-Hellman – TGDH
- STR

## Αρχιτεκτονικές του Secure Spread

Μέχρι τώρα περιγράψαμε ένα πλαίσιο εργασίας για την αυθεντικοποίηση και τον έλεγχο πρόσβασης στο Spread. Θα εξετάσουμε στην συνέχεια πως επιτυγχάνεται η εμπιστευτικότητα και η ακεραιότητα των πληροφοριών που ανταλλάσσονται μέσω του spread με χρήση του κλειδιού συνόδου.

To secure spread υποστηρίζει δύο αρχιτεκτονικές

- Layered Architecture (LA)
- Integrated Architecture (IA)

Στην IA αρχιτεκτονική η διαχείριση του μυστικού κλειδιού του group γίνεται από τους client. Κάθε χρήστης έχει την δυνατότητα να συμμετάσχει σε διαφορετικά groups με καθένα από τα οποία θα έχει ένα διαφορετικό μυστικό κλειδί. Στο καθένα από αυτά τα group μπορεί να χρησιμοποιήσει διαφορετικό πρωτόκολλο διαχείρισης του μυστικού κλειδιού του group.

Η IA αρχιτεκτονική έχει σαν φιλοσοφία η διαχείριση του μυστικού κλειδιού να μην γίνεται από τους client αλλά από τους servers. Οι servers είναι λιγότεροι σε αριθμό και ποιό σταθεροί από άποψη λειτουργίας, αυτό έχει σαν αποτέλεσμα να είναι πολύ ποιο εύκολη η σύναψη ενός μυστικού κλειδιού μεταξύ των servers. Για την σύναψη του μυστικού κλειδιού χρησιμοποιείται το πρωτόκολλο TGDH. Οι client θα συνδέονται με κάποιο ασφαλή τρόπο, για παράδειγμα με την χρήση SSL στους servers ή IPC (αν είναι εγκατεστημένοι στον ίδιο υπολογιστή με τον server).

Η διαδικασία σύναψης του μυστικού κλειδιού μεταξύ των server πραγματοποιείται μετά την ολοκλήρωση της διαδικασίας συμμετοχής. Κατά την διαδικασία συμμετοχής οι servers αυθεντικοποιούνται με χρήση πιστοποιητικών και κρυπτογραφίας δημοσίου κλειδιού.

Η IA αρχιτεκτονική έχει τρεις παραλλαγές. Σκοπός των παραλλαγών είναι η βελτιστοποίηση της απόδοσης του συστήματος ανάλογα με τον αριθμό των συνδεδεμένων ανά server clients. Οι αρχιτεκτονικές αυτές είναι οι ακόλουθες:

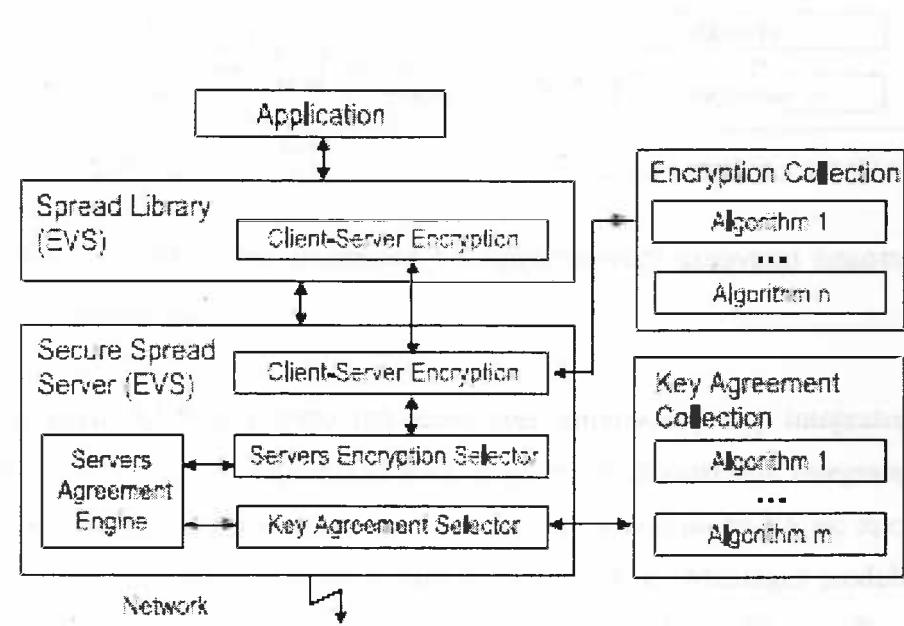
- Three step client – server
- Integrated VS
- Optimized EVS

Στην *three step client – server* αρχιτεκτονική υλοποιείται η βασική ιδέα της IA αρχιτεκτονικής όπως την περιγράψαμε παραπάνω. Οι servers έχουν εγκαθιδρύσει ένα μυστικό κλειδί συνόδου ενώ οι clients συνδέονται μέσο SSL ή IPC με τους servers. Η αρχιτεκτονική αυτή υποστηρίζει το VS και EVS μοντέλο. Βασικό μειονέκτημα της αρχιτεκτονικής είναι το μεγάλο κόστος κρυπτογράφησης στην περίπτωση που πολλοί clients συνδέονται μέσω SSL σε ένα server. Για την αποστολή ενός μηνύματος από client σε client απαιτούνται έξι βήματα.

1. Ο client κρυπτογραφεί το μήνυμα και το στέλνει με SSL
2. Ο server αποκρυπτογραφεί το μήνυμα

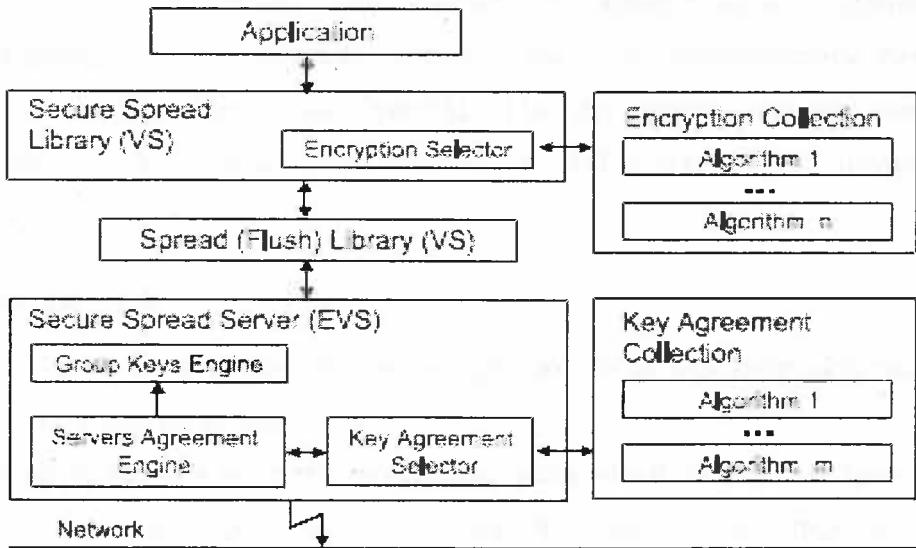
3. O server κρυπτογραφεί το μήνυμα με το μυστικό κλειδί συνόδου που μοιράζονται οι servers.
4. O server αποκρυπτογραφεί το μήνυμα
5. O server κρυπτογραφεί το μήνυμα και το αποστέλλει με SSL στον client.
6. O client αποκρυπτογραφεί το μήνυμα.

H three step client – server αρχιτεκτονική αναπαρίσταται στο παρακάτω σχήμα.



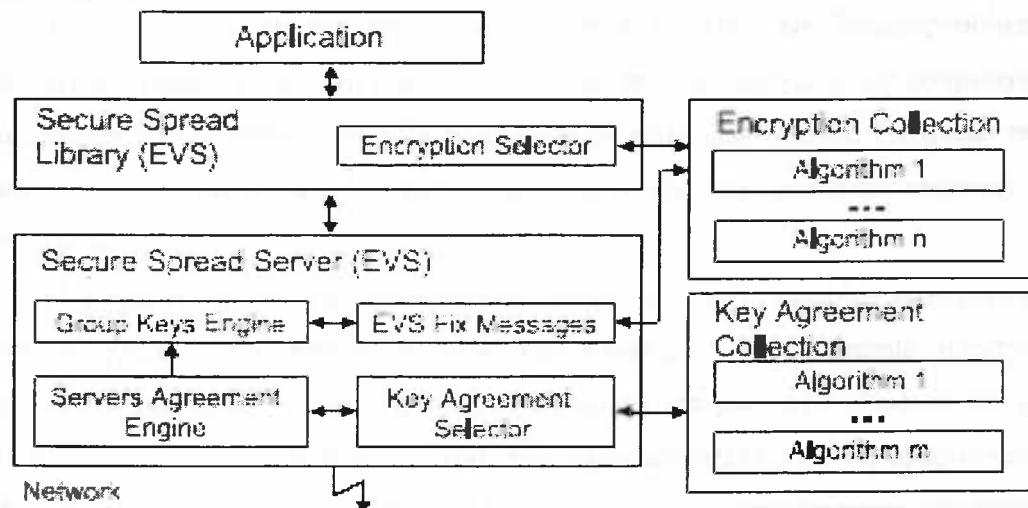
H Integrated VS αρχιτεκτονική υποστηρίζει το VS μοντέλο. H ιδέα είναι, οι clients να έχουν το δικό τους κλειδί συνόδου για κάθε group όπως και στην LA αρχιτεκτονική, το κλειδί αυτό όμως θα δημιουργείται από τον server στους οποίους συνδέονται οι clients που αποτελούν το group. Ο υπολογισμός του κλειδιού θα γίνεται βάση του μυστικού κλειδιού των servers, το όνομα του group και τον μοναδικό προσδιοριστή του group και θα διαμοιράζεται από τους servers στους clients. Οι server θα έχουν εγκαθιδρύσει ένα μυστικό κλειδί συνόδου με τρόπο ίδιο με αυτό της three step client – server αρχιτεκτονικής. Όταν θα υπάρχει αλλαγή στο view του group των clients, οι server είναι υπεύθυνοι για την αλλαγή του κλειδιού συνόδου του client group.

H Integrated VS αρχιτεκτονική απεικονίζεται στο παρακάτω σχήμα.



Όπως είναι φανερό με την Integrated VS αρχιτεκτονική μειώνεται δραστικά το κόστος κρυπτογράφησης.

Η Optimized EVS αρχιτεκτονική είναι μία παραλλαγή της Integrated VS αρχιτεκτονικής. Σκοπός της Optimized EVS είναι να συνδυάσει την Integrated VS αρχιτεκτονική και το EVS μοντέλο που πλεονεκτεί σε σχέση με το VS ως προς την απόδοση. Αυτό επιτυγχάνεται με την χρήση του EVS - Fix - Messages module που εξασφαλίζει την παράδοση των μηνυμάτων στο view στο οποίο στάλθηκαν. Ο τρόπος με τον οποίο δομείται η Optimized EVS αρχιτεκτονική φαίνεται στο παρακάτω σχήμα.



*Optimized EVS*

Σε όλες τις παραπάνω αρχιτεκτονικές ο προεπιλεγμένος αλγόριθμος κρυπτογράφησης είναι ο Blowfish. Μπορούν όμως να υποστηριχθούν όλοι οι αλγόριθμοι κρυπτογράφησης του OpenSSL. Για την αυθεντικοποίηση και την ακεραιότητα των δεδομένων χρησιμοποιείται η HMAC-SHA1 [20] συνάρτηση σύνοψης.

## Σύγκριση IA - LA

Οι δύο αρχιτεκτονικές έχουν σαν χαρακτηριστικό ότι τα πλεονεκτήματα της μίας είναι μειονεκτήματα της άλλης.

Η LA αρχιτεκτονική δίνει την δυνατότητα στους client να δημιουργήσουν ένα group στο οποίο η διαχείριση του κλειδιού συνόδου είναι αποκλειστικά δική τους ευθύνη, χωρίς την ανάμιξη των server. Η ιδιότητα αυτή συνεπάγεται μεγαλύτερη ασφάλεια, αλλά μικρότερη απόδοση.

Αντίθετα η IA αρχιτεκτονική μειώνει το κόστος για την διαχείριση του κλειδιού, αυξάνει όμως τις πιθανότητες παραβίασης, αφού χρησιμοποιείται το ίδιο κλειδί (το κλειδί συνόδου μεταξύ των servers) για περισσότερο χρόνο. Επιπλέον στην περίπτωση διάρρευσης του κλειδιού που μοιράζονται οι servers όλα τα groups είναι απροστάτευτα.

## Ensemble

### Ιδιότητες του Ensemble

Το Ensemble είναι ένα group communication σύστημα που δημιουργήθηκε από τα πανεπιστήμια Cornell και Hebrew. Επιτρέπει την δημιουργία group εφαρμογών με αξιόπιστη fifo – ordered, multicast και point to point επικοινωνία. Επιτρέπει επίσης και άλλες επικοινωνιακές ιδιότητες όπως casual και total multicast ordering, flow control κ.τ.λ.

Η αυθεντικοποίηση των νεοεισερχόμενων μελών στο Ensemble επιτυγχάνεται με την βοήθεια των συστημάτων PGP και Κέρβερος. Το Ensemble υποστηρίζει πολλαπλά partitions, αποτελεσματική αναθεώρηση του μυστικού κλειδιού του group (rekeying) και δυνατότητα δημιουργίας πολιτικών ασφάλειας από τις εφαρμογές. Μια δυνατότητα που παρέχει επίσης το Ensemble είναι η απομάκρυνση μη έμπιστων χρηστών από ένα group σε διάστημα λίγων millisecond.

To Ensemble μπορεί να χειριστεί αποτελεσματικά, περιπτώσεις αποτυχίας λόγο προβλήματος του δικτύου ή της εφαρμογής. Δεν ασχολείται όμως με αποτυχίες τύπου Byzantine failure επιπλέον δεν μπορεί να αντιμετωπίσει απειλές του τύπου traffic analysis και denial of service.

To Ensemble λοιπών διαθέτει ισχυρούς μηχανισμούς για την διαχείριση group partition και group merge, υιοθετεί το virtual synchrony μοντέλο που είναι αποτελεσματικότερο στην διαχείριση partition και merges. Ο αριθμός των χρηστών που μπορεί να υποστηριχθεί είναι της κλίμακας των λίγων εκατοντάδων. Στο Ensemble η ανοχή στα λάθη επιτυγχάνεται με την υλοποίηση failure – detection, membership, flow control και reliability πρωτοκόλλων.

Κάθε μέλος μπορεί να επιλέξει την δική του πολιτική εμπιστοσύνης. Σκοπός του Ensemble είναι να ενδυναμώσει αυτή την πολιτική δίνοντας την εγγύηση ότι μόνο αμοιβαία εμπιστευόμενα μέλη θα υπάρχουν σε ένα group.

Ιδιότητες όπως οι forward secrecy (FS) και backward secrecy (BS) υποστηρίζονται μερικός από το Ensemble. Η ενδυνάμωση τέτοιων ιδιοτήτων απαιτεί αλλαγή του κλειδιού συνόδου κάθε φορά που ένα μέλος εισέρχεται ή φεύγει από το group κάτι που συνεπάγεται υψηλό κόστος. Για το λόγο αυτό το Ensemble χαλαρώνει το PFS και το BS χωρίς όμως να δημιουργεί προβλήματα ασφάλειας. Κάτι τέτοιο επιτυγχάνεται με χρήση:

- Γρήγορων αλγόριθμων αναθεώρησης του κλειδιού συνόδου
- Αυτόματη ανανέωση κλειδιού από το σύστημα
- Ανανέωση κλειδιού από ένα νεοεισερχόμενο χρήστη που θυσιάζει την απόδοση χάρη της ασφάλειας.

Στο Ensemble υπάρχουν δύο είδη μηνυμάτων, αυτά που δημιουργούνται από τις εφαρμογές και ονομάζονται intra-groups ή regular messages και αυτά που δημιουργούνται από το Ensemble για επικοινωνία διαφορετικών components και partitions. Τα μηνύματα αυτά χρησιμοποιούνται για την επανένωση groups και ονομάζονται inter-component ή gossip messages.

Κάθε χρήστης μπορεί να δημιουργήσει τη δική του πολιτική ασφάλειας. Κάτι τέτοιο προϋποθέτει την ύπαρξη μηχανισμών και διαπιστευτηρίων για την αυθεντικοποίηση των μελών που συμμετέχουν σε ένα group. Τα διαπιστευτήρια μπορεί να είναι IP διευθύνσεις ή PGP πιστοποιητικά. Κάθε χρήστης δημιουργεί μία λίστα πρόσβασης (Access Control List - ACL) και την γεμίζει με τα διαπιστευτήρια των χρηστών που εμπιστεύεται. Το Ensemble με βάση την ACL κάθε μέλους

δημιουργεί group που περιέχουν μόνο αμοιβαία εμπιστευόμενους χρήστες. Όταν ένα χρήστης αλλάζει πολιτική ασφάλειας ζητά αναθεώρηση του κλειδιού συνόδου.

Στο Ensemble το κλειδί συνόδου παράγεται και διανέμεται στα μέλη από τον αρχηγό του group. Ο τρόπος με τον οποίο διανέμεται το κλειδί στα μέλη του group θα περιγραφεί στην συνέχεια.

Το Ensemble χρησιμοποιεί τον αλγόριθμο MD5 για ψηφιακή υπογραφή και τον αλγόριθμο RC4 για κρυπτογράφηση. Προεπιλεγμένο πρωτόκολλο για αυθεντικοποίηση είναι το PGP. Ανάλογα με τις απαιτήσεις της κάθε εφαρμογής για ασφάλεια και απόδοση δύναται να χρησιμοποιηθεί κάθε δυνατός συνδυασμός μηχανισμών κρυπτογράφησης, αυθεντικότητας και αυθεντικοποίησης αποστολέα.

Το Ensemble δεν υλοποιεί, υποστηρίζει όμως με κατάλληλες διεπαφές όλους τους μηχανισμούς κρυπτογράφησης του OpenSSL [13].

Η ακεραιότητα και η εμπιστευτικότητα στο Ensemble επιτυγχάνεται μέσω του μυστικού κλειδιού συνόδου. Το Ensemble δίνει την επιπλέον δυνατότητα στα μέλη ενός group να δημιουργούν ανά ζεύγη ασφαλή κανάλια επικοινωνίας με χρήση του πρωτοκόλλου χειραψίας Diffie – Hellman [14], τα κανάλια αυτά χρησιμοποιούνται για την διανομή του κλειδιού συνόδου.

## Συνένωση υπο – groups

Στο Ensemble για την επανένωση (merge) ενός group από τα τμήματα (partitions) που προκάλεσε κάποιο λάθος εφαρμογής ή πρόβλημα του δικτύου αναπτύχθηκε ένα πρωτόκολλο ασφαλούς επανένωσης. Κάθε τμήμα του αρχικού group μετά την κατάτμηση αποτελεί ένα ξεχωριστό group. Κάθε group στο Ensemble διαθέτει έναν αρχηγό. Το πρωτόκολλο επανένωσης εφαρμόζεται πάνω στους αρχηγούς των υπο-groups που επιθυμούν επανένωση. Για την αυθεντικοποίηση των αρχηγών γίνεται χρήση ασύμμετρης κρυπτογραφίας ενώ για την συμφωνία του μυστικού κλειδιού μεταξύ των δύο αρχηγών και κατά συνέπεια ολόκληρου του συνενωμένου group (το συμφωνηθέν κλειδί διανέμεται από τους αρχηγούς σε όλα τα μέλη του group) μπορούν να χρησιμοποιηθούν αλγόριθμοι της λογικής Diffie – Helman. Ο αλγόριθμος που χρησιμοποιεί το Ensemble είναι ο Bellare – Rogaway [15] που αντιμετωπίζει αποτελεσματικά επιθέσεις επανάληψης.

Δύο group μπορούν να συνενωθούν αν ο αρχηγός του ενός εμπεριέχεται στην ACL του άλλου. Αυτό το κριτήριο συνένωσης είναι προβληματικό στην περίπτωση

που δεν υπάρχουν αμοιβαίες σχέσεις εμπιστοσύνης μεταξύ των μελών των υπό ένωση groups. Το πλέον συνεπές κριτήριο συνένωσης θα ήταν ο έλεγχος από όλα τα μέλη των δύο group για την ύπαρξη αμοιβαίων σχέσεων εμπιστοσύνης μεταξύ τους. Μία τέτοια επιλογή θα δημιουργούσε μεγάλα προβλήματα απόδοσης. Για να σταθμιστεί το πρόβλημα ασφάλειας – απόδοσης το Ensemble επιτρέπει στις εφαρμογές να υλοποιήσουν μηχανισμούς για τον έλεγχο σχέσεων αμοιβαίας εμπιστοσύνης. Οι μηχανισμοί των εφαρμογών στηρίζονται στην ικανότητα του Ensemble να επιτρέπει στις εφαρμογές να αλλάζουν δυναμικά τις ACL τους και στην συνέχεια να απαιτούν από το Ensemble την αναθεώρηση του κλειδιού συνόδου.

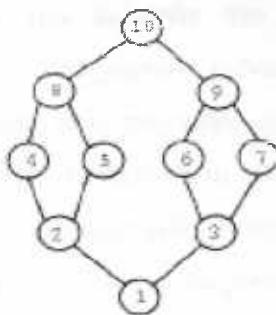
## Διανομή του κλειδιού συνόδου

Το Ensemble έχει υλοποιήσει τρις μηχανισμούς για αποτελεσματική διανομή του κλειδιού συνόδου.

Στον πρώτο μηχανισμό ο αρχηγός του group αποστέλλει το κλειδί με ασφαλή τρόπο σε κάθε μέλος του group και περιμένει θετικά acknowledgements. Σε περίπτωση που τα λάβει από όλα τα μέλη του group εγκαθιδρύει το νέο κλειδί. Ο τρόπος αυτός είναι αργός και δημιουργεί προβλήματα απόδοσης.

Στον δεύτερο μηχανισμό γίνεται χρήση ενός KDC και χρησιμοποιείται μία βελτιωμένη έκδοση του αλγόριθμου WGL [16,17,18] τον αλγόριθμο dWGL [19] για την διανομή των κλειδιών. Βασικό μειονέκτημα της χρήσης KDC είναι η κεντρικοποιημένη φύση του, με αποτέλεσμα να δημιουργείται στο σύστημα ένας σημείο αποτυχίας και συμφόρησης.

Ο τρίτος μηχανισμός χρησιμοποιεί την δομή ενός γράφου σε σχήμα διαμαντιού. Ο γράφος σχηματίζεται με τέτοιο τρόπο ώστε το ύψος του να είναι λογαριθμική συνάρτηση του αριθμού των κόμβων. Κάθε κόμβος αντιστοιχεί σε ένα μέλος ενώ κάθε ακμή που διασύνδει δύο κόμβους σε μία ασφαλή σύνδεση μεταξύ των κόμβων.



Παράδειγμα ενός τέτοιου γράφου απεικονίζεται στο παραπάνω σχήμα.

Το κλειδί δημιουργείται από τη ρίζα του γράφου (κόμβος 10) και διανέμεται στους κόμβους που συνδέονται άμεσα με αυτόν (8,9). Οι κόμβοι που έλαβαν το κλειδί το στέλνουν στους επόμενους π.χ. (8→4,5) μέχρι να το λάβουν όλοι οι κόμβοι. Δεν απαιτείται χρήση acknowledgements αφού ο τελευταίος κόμβος (1) θα γνωρίζει αν έλαβαν το κλειδί όλοι οι προηγούμενοι. Με αυτό τον μηχανισμό ανανέωσης επιτυγχάνεται γρήγορη μετάδοση χωρίς την χρήση KDC.

## Secure Group Layer

Το SGL διαφοροποιείται σε σχέση με τα συστήματα τα οποία περιγράψαμε μέχρι τώρα στο ότι δεν ασχολείται με τους μηχανισμούς και το μέσο μετάδοσης και τον τρόπο με τον οποίο ένα group communication system θα τους υποστηρίζει αλλά προσπαθεί να δημιουργήσει ένα ενδιάμεσο επίπεδο μεταξύ συστήματος μετάδοσης και εφαρμογής με αποκλειστικό σκοπό την ασφαλή επικοινωνία. Το SGL σκοπό έχει να προστατέψει τους χρήστες από απειλές όπως eavesdropping και spoofing, με την ενοποίηση ενός αξιόπιστου group communication system, ενός μηχανισμού αυθεντικοποίησης και διαχείρισης των δικαιωμάτων πρόσβασης των χρηστών του group και ενός πρωτοκόλλου διαχείρισης του κοινού μυστικού κλειδιού συνόδου που θα παρέχει δυνατότητα διασφάλισης των μεταδιδόμενων μηνυμάτων εξασφαλίζοντας εμπιστευτικότητα, ακεραιότητα και αυθεντικότητα. Τέλος αναφέρουμε ότι το SGL δεν παρέχει προστασία από επιθέσεις του τύπου denial of service.

Για να λειτουργήσει το SGL απαιτείται από το group communication system να υποστηρίζει την αρχή του extended virtual synchrony και εξασφαλίζει ότι τα μηνύματα θα παραδίδονται πάντοτε με την σωστή σειρά στα μέλη του group. Πρέπει επίσης το SGL να έχει γνώση των αλλαγών στο τρέχων view του group, για παράδειγμα είσοδο νέων μελών, αποχώρηση μελών, αποτυχία μιας εφαρμογής, διαχωρισμό ή επανένωση του group που προέκυψε από πρόβλημα στο δίκτυο ή από την επιδιόρθωση αυτού αντίστοιχα. Επιπρόσθετα στις παραπάνω δυνατότητες του group communication system το SGL εξασφαλίζει στις εφαρμογές την ιδιότητα του sending view delivery που όπως έχει ήδη αναφερθεί, εγγυάται ότι ένα μήνυμα θα

διανεμηθεί στην όψη στην οποία αρχικά στάλθηκε και είναι ιδιαίτερα χρήσιμη στην υλοποίηση υπηρεσιών ασφάλειας σε ένα group communication system.

Η αρχιτεκτονική του SGL αποτελείται από τέσσερα επίπεδα κάθε ένα από τα οποία υλοποιεί ένα διαφορετικό πρωτόκολλο. Τα επίπεδα αυτά είναι τα ακόλουθα:

- **Record layer** που εξασφαλίζει τις βασικές υπηρεσίας ασφάλειας των μηνυμάτων (εμπιστευτικότητα, ακεραιότητα, αυθεντικότητα)
- **Access control protocol** που είναι υπεύθυνο για την εισαγωγή νέων μελών
- **Flush protocol** που υλοποιεί το view synchrony
- **Key agreement protocol** που είναι υπεύθυνο για την δημιουργία ενός μυστικού κλειδιού συνόδου από το οποίο απορρέει ένα κλειδί κρυπτογράφησης και ένα κλειδί αυθεντικοποίηση (MAC key).

Θα δώσουμε τώρα μία ποιο περιεκτική περιγραφή των λειτουργιών του κάθε επιπέδου.

To **record layer** είναι υπεύθυνο για την κρυπτογράφηση των προς αποστολή μηνυμάτων με το μυστικό κλειδί συνόδου, την εφαρμογή ενός αλγορίθμου ακεραιότητας με χρήση του MAC κλειδιού και την αποστολή των μηνυμάτων μέσω του group communication system. Η τρέχουσα υλοποίηση του SGL χρησιμοποιεί τον Rijndael [4] αλγόριθμο κρυπτογράφησης και την HMAC [5] μέθοδο για τον υπολογισμό του MAC.

To **flush protocol** καθορίζει το τέλος ενός view και εγγυάται ότι δεν θα μεταδοθούν πλέον και στο έξεις μηνύματα κρυπτογραφημένα με το κλειδί συνόδου του view που έπαψε να υφίσταται. Το πρωτόκολλο flush ενεργοποιείται από κάποιο γεγονός που αλλάζει το τρέχων view. Όπως ήδη αναφέραμε το πρωτόκολλο flush υλοποιεί το sending view delivery, για να επιτευχθεί κάτι τέτοιο το flush στέλνει όλα τα εικρεμή μηνύματα και μπλοκάρει όλα τα νέα μηνύματα πριν στείλει το flush μήνυμα. Στην συνέχεια περιμένει μέχρι να φτάσουν οι επιβεβαιώσεις από τα μέλη ότι έλαβαν το flush μήνυμα. Μία επιβεβαίωση είναι ουσιαστικά μία “υπόσχεση” του μέλους ότι δεν θα στείλει άλλα μηνύματα κρυπτογραφημένα με κλειδί συνόδου που αντιστοιχεί στο παλιό view. Στην περίπτωση που το group communication protocol υποστηρίζει την αρχή του sending view delivery το επίπεδο flush συνεχίζει να είναι απαραίτητο για την λειτουργία του συστήματος.

To *access control protocol* είναι υπεύθυνο για την υλοποίηση των μηχανισμών ελέγχου πρόσβασης κάποιου μέλους σε ένα group. Το πρωτόκολλο κάνει χρήση πιστοποιητικών που εξουσιοδοτούν μία οντότητα να εισέλθει σε ένα group. Τα πιστοποιητικά εκδίδονται από μία αρχή υπεύθυνη για τον έλεγχο πρόσβασης και ουσιαστικά σε κάθε πιστοποιητικό αντιστοιχείται ένα δημόσιο κλειδί με μία X.509 ταυτότητα χρήστη [6]. Η αρχή έκτος από την έκδοση των πιστοποιητικών συλλέγει και όλες τις πολιτικές των groups για να μπορεί να προσδιορίσει ποίος έχει δικαίωμα εισόδου σε ένα group. Ένα επιπλέον καθήκον του server είναι η διαχείριση των πιστοποιητικών. Ο server διατηρεί μία λίστα με όλα τα ισχύοντα πιστοποιητικά και μια λίστα με τα ανακληθέντα πιστοποιητικά. Γίνεται ανάκληση ενός πιστοποιητικού όταν αλλάζει η πολιτική ενός group. Ένας τέτοιος server έχει υλοποιηθεί και ονομάζεται Akenti.

Όταν ένας χρήστης κάνει μία αίτηση σύνδεσης στο group τότε η ενέργειά του προκαλεί για αλλαγή view και ενεργοποιείται το flush πρωτόκολλο. Κατά την διάρκεια του flush πρωτοκόλλου κάθε μέλος στέλνει το πιστοποιητικό του σε όλους τους άλλους χρήστες και γίνεται αμοιβαίως έλεγχος των πιστοποιητικών όλων των μελών. Όταν ο group controller επαληθεύσει όλα τα μέλη τότε αρχίζει η διαδικασία για δημιουργία και εγκαθίδρυση του μυστικού κλειδιού συνόδου (*group key agreement*). Ο group controller είναι ένα μέλος που επιλέγεται τυχαία και κύρια εργασία του είναι ο συντονισμός μεταξύ των μελών του group για την συμφωνία και διανομή του κλειδιού συνόδου. Για την δημιουργία του κλειδιού χρησιμοποιείται το πρωτόκολλο IKA.1 της σουίτας πρωτοκόλλων Cliques [7,8].

Κλείνοντας την περιγραφή του SGL θα αναφέρουμε πως γίνεται η επιλογή του group controller στην περίπτωση συνένωσης upo - groups που δημιουργήθηκαν από το αρχικό group εξαιτίας κάποιου προβλήματος (π.χ. αποτυχία του δικτύου).

Στην περίπτωση αυτή σαν νέος group controller επιλέγεται ο controller του upo - group με τα περισσότερα μέλη. Ο εντοπισμός του μεγαλύτερου upo - group μπορεί να γίνει με την βοήθεια του flush πρωτοκόλλου ως εξής. Κάθε μέλος προστεθεί στο flush.view όλες τις διεργασίες (μέλη) που τρέχουν στο δικό του upo-group και στέλνει το μήνυμα του στους άλλους χρήστες. Με τον τρόπο αυτό όλα τα μέλη μπορούν να βρουν το group με τα περισσότερα μέλη.



## **Συστήματα επικοινωνίας ομάδας που αντιμετωπίζουν τις εκ των έσω απειλές – Το σύστημα Secure Ring**

Τα τέσσερα συστήματα ασφαλούς επικοινωνίας ομάδας που περιγράψαμε παραπάνω δεν πραγματεύονται απειλές από το εσωτερικό του group. Θεωρούν ότι κανένα από τα μέλη του group δεν είναι διεφθαρμένο και δεν θα διαφθαρεί στην διάρκεια της επικοινωνίας.

Μία σύγχρονη τάση στην ασφάλεια στις τεχνολογίες πληροφορικής και επικοινωνιών είναι η εξασφάλιση της βιωσιμότητας [21] ενός συστήματος παρά την κατάρρευση μέρους του συστήματος που οφείλεται σε αποτυχία του υλικού, του λογισμικού ή του επικοινωνιακού μέσου. Η επίτευξη της βιωσιμότητας ενός συστήματος αποκτά άλλη διάσταση όταν το σύστημα αυτό χρησιμοποιείται για την διαχείριση κρίσιμων υποδομών όπως υποδομές ενέργειας, ύδρευσης κ.τ.λ.

Το **Secure Ring** είναι ένα σύστημα επικοινωνίας ομάδας που εστιάζει στην βιωσιμότητα εξασφαλίζοντας προστασία από λάθη του τύπου “Byzantine faults”. Στο Secure Ring οι εφαρμογές επικοινωνούν με κατάλληλα modules, τους processors. Οι processors έχουν την ευθύνη της αποστολής και λήψης των μηνυμάτων που παράγουν οι δέχονται αντίστοιχα οι εφαρμογές.

Προκειμένου να βελτιώσει την απόδοση του το Secure Ring ακολουθεί μία αρχιτεκτονική που πετυχαίνει μεγάλη απόδοση όταν δεν ανιχνεύεται κάποιο Byzantine fault ενώ η απόδοση μειώνεται δραστικά όταν αποτυχία τέτοιου τύπου ανιχνευτεί. Σε αντίθεση με τα συνήθη πρωτόκολλα επικοινωνίας ομάδας που απαιτούν ψηφιακή υπογραφή κάθε μηνύματος που στέλνεται, στο Secure Ring τα μηνύματα στέλνονται χωρίς πρώτα να υπογραφούν ψηφιακά, από κάθε μήνυμα δημιουργείται μία σύνοψη, όλες οι συνόψεις εμφυτεύονται σε ένα token το οποίο κατέχει η οντότητα που κάθε φορά στέλνει μηνύματα στο group, στην συνέχεια το token υπογράφεται ψηφιακά με το ιδιωτικό κλειδί της οντότητας και αποστέλλεται στο group.

### **Θεώρηση του συστήματος**

Θεωρούμε ένα ασύγχρονό κατανεμημένο σύστημα που αποτελείται από κάποιον αριθμό processors που επικοινωνούν μέσω μηνυμάτων σε ένα πλήρως συνδεδεμένο δίκτυο. Κάθε processor έχει ένα μοναδικό προσδιοριστή (identifier). Το σύστημα

είναι ασύγχρονο οπότε δεν μπορεί να τεθούν χρονικοί περιορισμοί για επεξεργασία ή επικοινωνία. Οι processors μπορεί να έχουν πρόσβαση σε τοπικά ρολόγια, τα οποία όμως δεν είναι συγχρονισμένα. Κάθε group processor μπορεί να κάνει πολλαπλή αποστολή (multicast) μηνυμάτων στους άλλους processor του group και μπορεί να λαμβάνει τα μηνύματα που έστειλε.

Η επικοινωνία στηρίζεται σε ένα λογικό δακτύλιο με ένα token που ρυθμίζει ποιος processor έχει κάθε στιγμή το δικαίωμα αποστολής μηνυμάτων. Ο δακτύλιος (ring) είναι ο βασικός μηχανισμός πάνω στον οποίο λειτουργούν τα πρωτόκολλα που εξασφαλίζουν την σωστή σειρά μετάδοσης των μηνυμάτων.

Το σύνολο των μελών που συμμετέχουν στο group προσδιορίζεται από τους identifiers των processor που συμμετέχουν στο group.

Υπάρχει διαχωρισμός μεταξύ των όρων generate, originate, receive και deliver ενός μηνύματος.

- Μία εφαρμογή παράγει (generate) ένα μήνυμα.
- Ένας processor αρχικοποιεί (originate) το μήνυμα και το μεταδίδει
- Ένας processor λαμβάνει (receive) ένα μήνυμα και το διανέμει (deliver) σε μία εφαρμογή

Μερικές βασικές ιδιότητες του Secure Ring είναι οι παρακάτω:

- Το Secure Ring θεωρεί την επικοινωνία μεταξύ των processors αναξιόπιστη. Μπορεί να υπάρχει αυθαίρετη καθυστέρηση στην μετάδοσης αλλά θεωρείται ότι δεν υπάρχουν group partitions.
- Το σύστημα χρησιμοποιεί κρυπτογραφία δημοσίου κλειδιού προκειμένου οι processors να έχουν δυνατότητα να υπογράψουν ψηφιακά τα μηνύματα που αποστέλλουν. Κάθε processor έχει την δυνατότητα να μάθει τα δημόσια κλειδιά των άλλων processors του group προκειμένου να επιβεβαιώσει τις ψηφιακές υπογραφές.
- Η μη λειτουργία “σκάσιμο” ενός processor θεωρείται Byzantine fault και αντιμετωπίζεται από το σύστημα Secure Ring ως τέτοια.
- Αν το group αποτελείται από n processors τότε πρέπει να λειτουργούν τουλάχιστον  $(2n+1)/3$  σωστά προκειμένου το σύστημα στην ολότητά του να συνεχίσει να λειτουργεί σωστά.

## Πρωτόκολλα του Secure Ring

To Secure Ring αποτελείται από τέσσερα πρωτόκολλα τα:

- Message deliver protocol
- Primary component membership protocol
- Unreliable Byzantine fault detector protocol
- Message diffusion protocol

Θα δώσουμε μία σύντομη περιγραφή του τι κάνει το κάθε πρωτόκολλο και πως επιτυγχάνει τους στόχους του.



### *Message deliver protocol*

Το πρωτόκολλο αυτό είναι υπεύθυνο για την διανομή δύο είδους μηνυμάτων. Τα μηνύματα εφαρμογής (regular message) και τα μηνύματα αλλαγής view (configuration change message). Το πρωτόκολλο παρέχει:

- Ακεραιότητα των μηνυμάτων (integrity)
- Μοναδικότητα μηνύματος (uniqueness) δηλαδή δεν υπάρχουν μηνύματα με τον ίδιο identifier και διαφορετικό περιεχόμενο.
- Self delivery δηλαδή κάθε processor θα λάβει τα μηνύματα που έστειλε
- Ατομικότητα (atomicity). Ένας processor p διανέμει ένα μήνυμα σε μία view αν και μόνο αν o processor q που ανήκει στο ίδιο view με τον p διανέμει ένα τέτοιο μήνυμα.
- Total order δηλαδή o p διανέμει το μήνυμα m<sub>1</sub> πριν το μήνυμα m<sub>2</sub> αν και μόνο αν o q στο ίδιο view διανέμει το m<sub>1</sub> πριν το m<sub>2</sub>

To message deliver πρωτόκολλο μπορεί να εξασφαλίσει τις παραπάνω αρχές όταν τουλάχιστον  $(2n + 1)/3$  processors λειτουργούν σωστά. Η αρχή του total order που είναι η βάση για την εξασφάλιση ασφαλούς και αξιόπιστης μετάδοσης μηνυμάτων επιτυγχάνεται με την χρήση ενός λογικού δακτυλίου με ένα token που περιέχει πληροφορίες για την σειρά μετάδοσης και τον έλεγχο ροής των μηνυμάτων. Μόνο ο processor που κατέχει κάθε φορά το token έχει δικαίωμα να στείλει μηνύματα. Η επικεφαλίδα κάθε μηνύματος περιέχει ένα μοναδικό αριθμό σειράς.



To message deliver πρωτόκολλο κάνει χρήση συνόψεων των μηνυμάτων, οι συνόψεις εμφυτεύονται στο token το οποίο ο processor μετά το τέλος της αποστολής υπογράφει ψηφιακά. Με τον τρόπο αυτό επιτυγχάνεται καλύτερη απόδοση του συστήματος μίας και δεν απαιτείται η ψηφιακή υπογραφή κάθε μηνύματος. Γίνεται φανερό ότι η αυθεντικοποίηση των μηνυμάτων δεν γίνεται σε πραγματικό χρόνο άλλα όταν φθάσει το token μετά την αποστολή όλων των μηνυμάτων από κάποιο processor.

Κάθε regular message περιέχει τον identifier του processor που αρχικοποίησε (originate) το μήνυμα, τον identifier του ring, τον αριθμό σειράς του μηνύματος (message sequence number) και το περιεχόμενο που προέρχεται από την εφαρμογή.

Για την δημιουργία ενός regular μηνύματος απαιτείται η ύπαρξη ενός regular token. Ο processor πρέπει να είναι κάτοχος του regular token προκειμένου να μπορεί να κάνει originate τα regular messages. Κάθε token περιέχει έναν αριθμό σειράς, τον ring identifier την σύνοψη του προηγούμενου token και μία λίστα με τις συνόψεις των μηνυμάτων που έστειλε ο τρέχον κάτοχος του token. Τα στοιχεία αυτά είναι σημαντικά για την εξασφάλιση της ασφάλειας παράδοσης. Όταν ένας processor κατέχει το regular token μπορεί να στείλει κάποιον προκαθορισμένο αριθμό μηνυμάτων. Επιπλέον το token περιέχει ένα πεδίο το aru (all received up to) που χρησιμοποιείται για δηλώσει ότι ο processor που έστειλε το token έχει λάβει κάθε μήνυμα με identifier ίσο ή μικρότερο του aru. Ένα άλλο πεδίο που χρησιμοποιείται είναι το acu (all checked up to) που δηλώνει ότι ο processor έχει ελέγξει κάθε μήνυμα με identifier ίσο ή μικρότερο του acu. Το πεδίο token\_aru δηλώνει ότι ο processor έχει λάβει κάθε token με identifier μικρότερο ή ίσο του token\_aru ενώ το πεδίο token\_acu δηλώνει ότι έχει ελέγξει κάθε token με identifier μικρότερο ή ίσο του token\_acu. Τέλος τα πεδία rtl\_list και rtg\_list περιέχουν τους identifier των μηνυμάτων που ο κάτοχος του token έχει ζητήσει για επαναμετάδοση και έχει επαναμεταδόσει αντίστοιχα, ενώ τα token\_rtl\_list και token\_rtg\_list περιέχουν τους identifier των token που ο κάτοχος του token έχει ζητήσει για επαναμετάδοση ή έχει επαναμεταδόσει αντίστοιχα.

Όταν ένας processor λαμβάνει ένα regular μήνυμα και το αντίστοιχο token, ελέγχει αν το μήνυμα αντιστοιχεί στην σύνοψη που έχει εμφυτευτεί στο token, στην περίπτωση μη αντιστοιχίας ο processor αγνοεί το μήνυμα. Ο processor αγνοεί επίσης κάθε token που δεν έχει μία έγκυρη υπογραφή.

Όταν ένας processor γίνεται κάτοχος του token τότε δημιουργεί μία σύνοψη του token του προκατόχου του και την εμφυτεύει στο token που θα στείλει στο group. Επίσης για κάθε μήνυμα που αποστέλλει δημιουργεί μία σύνοψη και την εμφυτεύει στο token. Όταν ο processor τελειώσει την μετάδοση μηνυμάτων αυξάνει τον αύξων αριθμό του token το υπογράφει ψηφιακά και το αποστέλλει στο group.

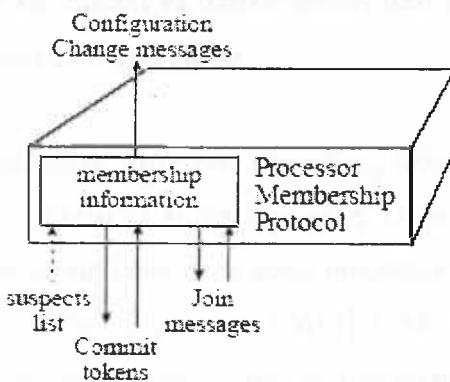
Ένας processor διανέμει ένα μήνυμα όταν έχει διανείμει όλα τα μηνύματα με αριθμό σειράς μικρότερο του αριθμού σειράς του προς μετάδοση μηνύματος και έχει διανείμει και όλα τα tokens αυτών των μηνυμάτων.

### ***Primary component membership protocol***

Το πρωτόκολλο αυτό είναι υπεύθυνο για την αναδιαμόρφωση του group όταν εντοπιστεί ένα Byzantine fault. Το membership protocol δημιουργεί ένα νέο δακτύλιο με τους σωστούς processors και εξασφαλίζει:

- Μοναδικότητα των identifiers του view
- Self – inclusion δηλαδή ο processor που δημιουργεί το νέο view εμπεριέχει σε αυτό και τον εαυτό του.
- Eventual inclusion - Αν δύο processors είναι σωστοί κάποια στιγμή θα δημιουργηθεί ένα view στο οποίο θα εμπεριέχονται και οι δύο
- Eventual exclusion - Αν υπάρχει ένας σωστός processor έστω  $p$  και ένας processor που έχει υποπέσει σε ένα Byzantine fault έστω  $q$  τότε μετά από κάποιο χρονικό διάστημα, ο  $p$  θα δημιουργήσει ένα view που θα αποκλείει τον  $q$  και από εκεί και έπειτα ο  $p$  δεν θα δημιουργήσει ποτέ ένα view που θα περιέχει τον  $q$ .
- Total order of configurations - Αν δύο σωστοί processors έστω  $p$  και  $q$  δημιουργήσουν δύο view  $C_1$  και  $C_2$  τότε ο  $p$  θα εγκαταστήσει το  $C_2$  πριν το  $C_1$  αν και μόνο αν ο  $q$  εγκαταστήσει το  $C_2$  πριν το  $C_1$

Στο παρακάτω σχήμα απεικονίζονται τα μηνύματα και tokens που δέχεται και αποστέλλει το membership protocol. Όπως παρατηρούμε τα μηνύματα αλλαγής view μεταβιβάζονται στο message deliver protocol ενώ η λίστα με τους ύποπτους processors λαμβάνεται από το Byzantine fault detector.



The membership protocol

To membership protocol περιέχει επτά ειδικούς τύπους μηνυμάτων και δομές δεδομένων. Αυτά είναι τα:

- **Join message** – Τα μηνύματα αυτά ανταλλάσσονται μεταξύ των processors κατά την διαδικασία δημιουργίας νέου view. Κάθε μήνυμα αυτού του τύπου περιέχει πεδία για την συμπλήρωση του identifier του sender, τον identifier του δακτυλίου, τον αριθμό σειράς του μηνύματος, το σύνολο των processor που ο sender πιστεύει ότι μπορεί να δικαιούνται να συμμετάσχουν στο νέο δακτύλιο και τέλος το σύνολο των processor που ο fault detector του sender θεωρεί ύποπτους. Κάθε ένα από αυτά τα μηνύματα υπογράφεται ψηφιακά από τον αποστολέα του και μεταδίδεται μέσω του message diffusion protocol.
- **Commit token** – Το commit token χρησιμοποιείται για να επιβεβαιώσει ότι κάθε processor δεσμεύεται στο νέο συμφωνηθέν σύνολο μελών. Το commit token περιέχει πεδία στα οποία συμπληρώνονται ο identifier του sender, ο identifier του δακτυλίου, ο αριθμός σειράς του token και μία λίστα με τους identifiers των νέων μελών. Επίσης περιέχει ένα πεδίο στο οποίο συμπληρώνεται ένας αριθμός που δηλώνει ότι ο processor έχει ελέγξει κάθε μήνυμα με αριθμό σειράς μικρότερο ή ίσο από τον αριθμό που εμπεριέχεται στο πεδίο. Κάθε commit token υπογράφεται ψηφιακά και μεταδίδεται μέσω του message diffusion protocol.
- Κάθε processor διαχειρίζεται μία σειρά τοπικών μεταβλητών που περιέχουν πληροφορίες σχετικά με το membership του τρέχοντος δακτυλίου, τους processor που μπορούν να γίνουν πιθανά μέλη, τους processor που

θεωρούνται ύποπτοι και πρέπει να αποκλειστούν από μελλοντικά view καθώς και μεταβλητές με άλλες πληροφορίες.

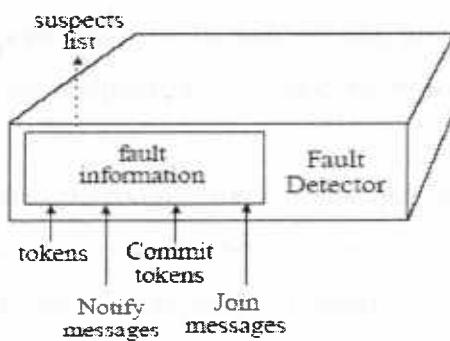
Το πρωτόκολλο ενεργοποιείται όταν ένα νέο μέλος θέλει να μπει στο group ή όταν συμβεί μία αλλαγή στην λίστα με τα ύποπτα μέλη. Οι processor καταλήγουν σε συμφωνία δημιουργίας νέου group όταν στον κάθε processor η διαφορά μεταξύ των αποδεκτών και ύποπτων μελών δεν ξεπερνά το  $(2n+1)/3$ . Με τον τρόπο αυτό μπορεί να δημιουργηθεί ένα group που περιέχει και αναξιόπιστους processor. Για να αντιμετωπιστεί αυτή η αδυναμία, με την δημιουργία του group το commit token περιστρέφεται στο δακτύλιο προκειμένου να επιβεβαιωθεί ότι όλοι οι processors είναι σωστοί.

### *Unreliable Byzantine fault detector protocol*

Σκοπός αυτού του πρωτοκόλλου είναι να εντοπιστούν οι εσφαλμένοι processors. Το πρωτόκολλο παρέχει τις αρχές:

- Eventual strong Byzantine completeness - Η αρχή αυτή σημαίνει ότι μετά από κάποιο χρονικό διάστημα κάθε processor που εκτελεί ένα Byzantine fault που το Secure Ring μπορεί να εντοπίσει θα είναι μόνιμα ύποπτος από κάθε σωστό processor.
- Eventual strong accuracy – Μετά από κάποιο χρονικό διάστημα κανένας σωστός processor δεν θα θεωρείται ύποπτος από τους άλλους σωστούς processors.

Η αρχιτεκτονική του fault detector φαίνεται στο παρακάτω σχήμα.



The Byzantine fault detector

Ο Byzantine fault detector δημιουργεί την λίστα των ύποπτων processor και την μεταβιβάζει στο membership protocol. Προκειμένου ο fault detector να έχει τις απαραίτητες πληροφορίες ώστε να δημιουργήσει την λίστα των ύποπτων processors τα token, commit token και join messages μεταβιβάζονται σ' αυτόν.

Ο fault detector χρησιμοποιεί ένα ιδικό τύπο μηνυμάτων που ονομάζονται Notify messages για ανταλλάξει πληροφορίες που είναι απαραίτητες για τον εντοπισμό ενός εσφαλμένου processor που έχει στείλει μεταλλαγμένα token. Τα notify messages διαβιβάζονται μέσο του message diffusion protocol που θα αναλύσουμε παρακάτω.

Τα λάθη που ο Byzantine fault detector μπορεί να εντοπίσει στηρίζονται σε:

- **Αποστολή μεταλλαγμένων μηνυμάτων (*mutant messages*)**. Ένα μήνυμα θεωρείται μεταλλαγμένο αν η σύνοψη του προηγούμενου token που εμπεριέχεται στο μήνυμα δεν ανταποκρίνεται στην σωστή σύνοψη. Στην περίπτωση που ένας processor λάβει ένα τέτοιο μήνυμα δημιουργεί και διανέμει περιοδικά ένα notify message μέχρι να γίνει αλλαγή στο view. Αν κάποιος processor που στο τρέχων view δεν έχει στείλει notify message λάβει ένα notify message με κατάλληλη δομή από κάποιον άλλο processor, τότε δημιουργεί ένα δικό του notify message που περιέχει το notify message που έλαβε, επιπλέον προσθέτει το χρήστη που έστειλε το μεταλλαγμένο μήνυμα στην λίστα των ύποπτων μελών.
- **Αποστολή μηνυμάτων με μη κατάλληλη μορφή (*improperly formed messages*)**. Όταν ένας processor λάβει ένα token, join message, commit token ή notify message που δεν έχει κατάλληλη μορφή τότε προσθέτει τον αποστολέα του στην λίστα των ύποπτων processor.  
Ένα token δεν θεωρείται σε κατάλληλη μορφή όταν ο αριθμός σειράς του δεν είναι κατά ένα μεγαλύτερος του προηγούμενου token.  
Ένα notify message θεωρείται σε ακατάλληλη μορφή όταν δεν περιέχει τα δύο token που δεν συμβαδίζουν ως προς τις συνόψεις ή ένα άλλο notify message.
- **Αποστολή παραπλανητικών Token (*improper reporting of messages*)**. Ένας διεφθαρμένος processor έχοντας σκοπό να διασπάσει την σωστή σειρά των μηνυμάτων θα μπορούσε να στείλει ένα token στο οποίο θα είχε αυξήσει τον αύξων αριθμό του token χωρίς όμως να στείλει ποτέ το μήνυμα. Αν δεν υπήρχε κατάλληλος μηχανισμός ανίχνευσης τέτοιου είδους σφάλματος κάποιοι σωστοί processors θα αποκλείονταν από το

group σαν αδύναμοι να λάβουν μηνύματα. Για να αντιμετωπιστεί αυτή η απειλεί ο fault detector ελέγχει πόσοι processors έχουν ζητήσει την επαναμετάδοση του μηνύματος. Αν ο αριθμός τους είναι τουλάχιστον  $(2n+1)/3$  τότε το μήνυμα λαμβάνεται σαν ανύπαρκτο και ο αποστολέας του token που αναφέρεται στο μήνυμα προστίθεται στην λίστα των ύποπτων μελών.

- **Αποτυχία στην επιβεβαίωση λήψης μηνύματος (failure to acknowledge messages)**) Δεν επιτρέπεται σε έναν processor να μην στέλνει επιβεβαιώσεις των μηνυμάτων που λαμβάνει γιατί σε μια τέτοια περίπτωση όλοι οι υπόλοιποι processors θα αναγκάζονταν να διατηρούν στους buffer τους αυτά τα μηνύματα. Για το λόγο αυτό αν ένας processor μεταδίδει πάνω από έναν προκαθορισμένο αριθμό μηνυμάτων στα οποία ζητά για επαναμετάδοση το ίδιο μήνυμα τότε προστίθεται στους αναξιόπιστους processors.
- **Αποτυχία αποστολής μηνύματος (failure to send message).** Κάθε φορά που ένας processor λαμβάνει ένα token ένα μηχανισμός time out ενεργοποιείται, αν μέσα στον προκαθορισμένο χρόνο ο κάτοχος του token δεν στείλει το νέο token τότε προστίθεται στους αναξιόπιστους processors. Το ίδιο ισχύει και στην περίπτωση των commit tokens.
- **Μετακίνηση processor από την λίστα ύποπτων processors (removal from the fault set ).** Όποιος processor προστεθεί στην λίστα ύποπτων επειδή έστειλε ένα μήνυμα σε ακατάλληλη μορφή ή ένα μεταλλαγμένο μήνυμα δεν μετακινείται ποτέ από την λίστα ύποπτων. Κάποιος processor όμως που δεν κατάφερε να στείλει ή να λάβει ένα μήνυμα λόγο λήξης του χρόνου μπορεί να είναι απλά αργός και όχι διεφθαρμένος. Για να γίνει με κάποιο τρόπο διαχωρισμός μεταξύ διεφθαρμένων και αργών processors, κάθε φορά που ένας processor προστίθεται στην λίστα ύποπτων λόγο λήξης κάποιου time out κρατιέται σε ένα ιδικό πεδίο ένας μετρητής. Αν ο μετρητής αυτός ξεπεράσει κάποιο προκαθορισμένο όριο τότε ο processor δεν έχει δυνατότητα να μετακινηθεί από την λίστα ύποπτων. Αν ο μετρητής που αντιστοιχεί στον processor δεν έχει ξεπεράσει το προκαθορισμένο όριο και ο processor καταφέρει να λάβει ή να στείλει κάποιο μήνυμα τότε μετακινείται από την λίστα ύποπτων. Επίσης μετακινείται από την λίστα ακόμα και αν ο μετρητής έχει περάσει το

προκαθορισμένο όριο στην περίπτωση που η διαφορά μεταξύ μη διεφθαρμένων και διεφθαρμένων processors είναι μικρότερη ή ίση του  $(2n+1)/3$ .

### *The message diffusion protocol*

To membership protocol και o fault detector χρησιμοποιούν το message diffusion protocol για την μετάδοση μηνυμάτων ειδικού τύπου. Το πρωτόκολλο αυτό εξασφαλίζει της αρχές του:

- Self – receipt. Αν ένας σωστός processor διανέμει ένα μήνυμα μέσο του diffusion protocol τότε θα το λάβει και ο ίδιος.
- Uniform receipt. Αν ένα σωστός processor λάβει ένα μήνυμα μέσω του diffusion protocol τότε θα λάβει το μήνυμα αυτό και κάθε σωστός processor.

### Αρχή λειτουργίας

Στο diffusion protocol κάθε φορά που ένας processor λαμβάνει ένα μήνυμα τότε το αποστέλλει σε όλους τους άλλους processors και ύστερα το ξαναλαμβάνει. Το πρωτόκολλο απαιτεί αξιόπιστο μέσο μετάδοσης, αν το μέσο δεν μπορεί να εγγυηθεί αξιοπιστία το πρωτόκολλο δεν μπορεί να εγγυηθεί τις δύο παραπάνω αρχές.

### **Κριτήρια επιλογής ενός συστήματος – Σύγκριση συστημάτων**

Τα κριτήρια βάση των οποίων μπορεί να γίνει η επιλογή ενός group communication συστήματος είναι τα ακόλουθα:

- Η εξασφάλιση αξιόπιστης επικοινωνίας
- Η ύπαρξη αποτελεσματικού μηχανισμού αυθεντικοποίησης
- Η ύπαρξη αποτελεσματικών μηχανισμών ελέγχου πρόσβασης
- Η ύπαρξη αποτελεσματικού μηχανισμού εγκαθίδρυσης και διαχείρισης του κλειδιού συνόδου του group.
- Η ευκολία με την οποία μπορούμε να δημιουργήσουμε εφαρμογές που θα στηρίζονται στο group communication σύστημα.

- Η αρχιτεκτονική του συστήματος. Αν το σύστημα ακολουθεί μία modular αρχιτεκτονική και δίνει την δυνατότητα ενσωμάτωσης modules με εύκολο τρόπο, ώστε να είναι σχετικά απλές, μελλοντικές τροποποιήσεις του συστήματος. π.χ. προσθήκη αλγορίθμων κρυπτογράφησης, μηχανισμών ελέγχου πρόσβασης και αυθεντικοποίησης
- Το μέγεθος του group που μπορεί το σύστημα να υποστηρίξει
- Η δυνατότητα υλοποίησης πολιτικών

Με βάση τις παραπάνω ιδιότητες και όσα αναφέρθησαν στις προηγούμενες ενότητες, παρατηρούμε ότι κανένα από τα ήδη υπάρχοντα group communication συστήματα δεν καλύπτει εξολοκλήρου τις απαιτήσεις μας.

Συγκριτικά με τα υπόλοιπα group communication συστήματα το Secure Spread είναι αυτό που ικανοποιεί αρκετές από τις ιδιότητες που αναφέρθησαν παραπάνω. Το Secure Spread χρησιμοποιεί το clique toolkit για την εγκαθίδρυση και διαχείριση του κλειδιού συνόδου με ένα κατανεμημένο συνεργατικό τρόπο. Αντίθετα στο Ensemble και το Antigone το κλειδί διανέμεται από τον αρχηγό του group στα υπόλοιπα μέλη.

Ένα σημείο που φαινομενικά μειονεκτεί το Secure Spread σε σχέση με το Ensemble και το Antigone, είναι το μέγεθος του group που μπορεί να υποστηρίξει. Όπως έχουμε ήδη αναφέρει τα κατανεμημένα συνεργατικά πρωτόκολλα διαχείρισης του κλειδιού συνόδου ενός group, όταν ο αριθμός των συνδεδεμένων μελών κυμανθεί περί τα εκατό γίνονται ακατάλληλα λόγο του μεγάλου επικοινωνιακού και υπολογιστικού τους κόστους. Το πρόβλημα αυτό μπορεί να ξεπεραστεί στο Secure Spread με την υιοθέτηση κάποιας, από τις Integrated αρχιτεκτονικές που υλοποιεί. Η επιλογή της αρχιτεκτονικής εξαρτάτε από τον σκοπό για τον οποίο θα χρησιμοποιηθεί το Secure Spread και από την πολιτική ασφάλειας των εφαρμογών, που θα κληθεί να υποστηρίξει (οι integrated αρχιτεκτονικές του Secure Spread είναι μόνο εμπορικά διαθέσιμες).

Το Secure Spread παρέχει τα κατάλληλα APIs ώστε η δημιουργία εφαρμογών να είναι ιδιαίτερα απλή και δεν απαιτεί από τους δημιουργούς των εφαρμογών γνώση της αρχιτεκτονικής ολόκληρου του συστήματος.

Σε αντίθεση με το Antigone και το Ensemble, το Secure Spread ακολουθεί μία modular αρχιτεκτονική που του δίνει την δυνατότητα για εύκολες μελλοντικές τροποποιήσεις.

Η “αχύλειος πτέρνα” του Secure Spread είναι η έλλειψη μηχανισμού αυθεντικοποίησης και έλεγχου πρόσβασης των νέων μελών στο group. Το Secure Spread αφήνει στην διάθεση του χρήστη την επιλογή κατάλληλων μηχανισμών αυθεντικοποίησης και ελέγχου πρόσβασης. Οι μηχανισμοί αυτοί μπορούν να ενσωματωθούν στο Secure Spread με την χρήση δύο modules, των client και server authentication modules. Η ενσωμάτωση λοιπόν ενός κατάλληλου μηχανισμού αυθεντικοποίησης και ελέγχου πρόσβασης στο Secure Spread θα το καθιστούσε αδιαφιλονίκητο “φαβορί” μεταξύ των επιλογών μας.

Κατά καιρούς έχουν ενοποιηθεί διάφοροι μηχανισμοί αυθεντικοποίησης και έλεγχου πρόσβασης με το Secure Spread όπως IP Access Control, password authentication κ.α., η ποίο ενδιαφέρουσα περίπτωση όμως είναι αυτή της ενοποίησης του Secure Spread με τον Bouncer. Η περίπτωση αυτή θα παρουσιαστεί αναλυτικά στην συνέχεια.

Το SGL και το Secure Ring είναι δύο συστήματα που δεν αναφέραμε στην παραπάνω σύγκριση. Αυτά τα δύο συστήματα έχουν διαφορετικά χαρακτηριστικά και για αυτό αποφασίσαμε να αναλυθούν ξεχωριστά.

Το Secure Ring όπως ήδη έχουμε αναφέρει έχει σαν βασικό του στόχο την διασφάλιση του συστήματος από τις εκ’ των έσω απειλές. Το μοντέλο απειλών που αναλύεται στην παρούσα εργασία και που υιοθετούν τα υπόλοιπα τέσσερα συστήματα που περιγράψαμε, δεν περιέχει τις εκ’ των έσω απειλές και κατά συνέπεια οποιαδήποτε σύγκριση θα ήταν άσκοπη.

Όσον αφορά το SGL, η βασική του διαφορά είναι ότι δεν υλοποιεί το υποκείμενο επικοινωνιακό σύστημα άλλα παρέχει μηχανισμούς διασφάλισης ενός τέτοιου συστήματος. Το SGL όπως και το secure spread κάνει χρήση του clique toolkit για την συμφωνία του μυστικού κλειδιού συνόλου με ένα κατανεμημένο συνεργατικό τρόπο. Τα χαρακτηριστικά του SGL που το κάνουν να μειονεκτεί σε σχέση με το secure spread είναι τα ακόλουθα.

- Η ενοποίηση του SGL με ένα group communication σύστημα που δεν υλοποιεί κάποια αρχιτεκτονική παρόμοια με τις παραλλαγές της Integrated αρχιτεκτονικής του secure spread θα έχει σαν αποτέλεσμα ένα σύστημα με προβλήματα απόδοσης και επεκτασιμότητας
- Το SGL για τον έλεγχο πρόσβασης κάνει χρήση ενός server (Akenti). Ο server μπορεί είναι κοινός για πολλά groups κάτι που τον κάνει ελκυστικό

στόχο για επιθέσεις και αποτελεί μοναδικό σημείο συμφόρησης και αποτυχίας όλου του συστήματος.

## Ενότητα 4<sup>η</sup>: Ενοποίηση Secure Spread – Bouncer

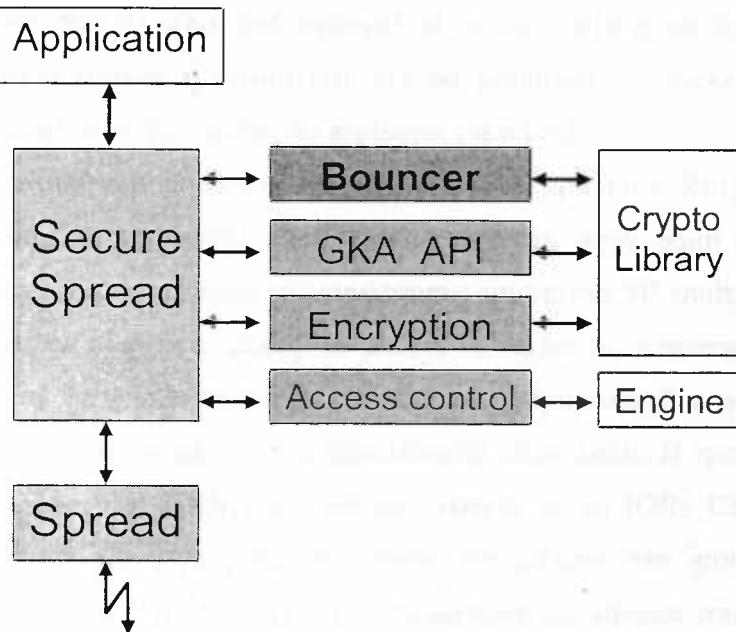
### Αρχιτεκτονική ενοποίησης

Η ενοποίηση του secure spread με τον Bouncer έγινε από τους δημιουργούς του Bouncer με σκοπό να μετρηθεί η απόδοση των διάφορων σχημάτων ψηφιακών υπογραφών σε ένα σύγχρονο group communication σύστημα. Για την μέτρηση της απόδοσης σε ένα ασύγχρονο σύστημα έγινε ενοποίηση του Bouncer με το δημοφιλές peer to peer σύστημα Gnutella.

Ουσιαστικά ο Bouncer ενοποιήθηκε με το υποκείμενο επικοινωνιακό σύστημα spread πάνω στο οποίο στηρίζεται το secure spread. Το ενοποιημένο σύστημα spread – Bouncer μπορεί να εγκατασταθεί και να λειτουργήσει χωρίς να είναι απαραίτητη η εγκατάσταση των βιβλιοθηκών που παρέχουν ασφάλεια στο spread και συνθέτουν το σύστημα secure spread. Βέβαια η χρήση του συστήματος κατά αυτό τον τρόπο δεν παρέχει ασφάλεια αφού δεν εμποδίζει κάποιον επίβουλο να υποκλέψει την επικοινωνία μεταξύ των μελών του group.

Στην περίπτωση του secure spread έχουν ενοποιηθεί μόνο τα κεντρικοποιημένα πρωτόκολλα ελέγχου πρόσβασης, που έχουμε ήδη αναλύσει στην ενότητα περιγραφής του Bouncer. Το γεγονός αυτό μας αναγκάζει να κάνουμε χρήση ενός on – line GAUTH που θα πιστοποιεί τα GMCs. Η χρήση του GAUTH αποτελεί βέβαια μία παραβίαση της κατανεμημένης φύσης των group communication συστημάτων, ο ρόλος του όμως δεν είναι η λήψη απόφασης πρόσβασης άλλα απλά η δημιουργία ενός πιστοποιητικού αφού πρώτα το υποψήφιο μέλος έχει συγκεντρώσει των απαιτούμενο αριθμό ψήφων από τα υπόλοιπα μέλη του group. Έξαλλον τόσο το secure spread όσο και ο Bouncer είναι project ανοικτού κώδικα κάτι που δίνει την δυνατότητα σε όποιον επιθυμεί να ενοποιήσει και τα κατανεμημένα πρωτόκολλα του Bouncer στο secure spread.

Στο σχήμα που ακολουθεί απεικονίζεται η ενοποίηση του secure spread με τον Bouncer.



Για την παραπάνω ενοποίηση χρειάστηκε η αλλαγή και η τροποποίηση κάποιων συναρτήσεων του spread. Συγκεκριμένα προστέθηκε η συνάρτηση SP\_GAC\_join (maibox mbox, const char \*group) που δηλώνεται στο sp.h του δένδρου κώδικα του spread. Η συνάρτηση αυτή χρησιμοποιείται για την είσοδο ενός μέλους σε ένα group κάνοντας χρήση των μηχανισμών ελέγχου πρόσβασης του Bouncer. Το όνομα του group που το μέλος επιθυμεί να συνδεθεί μπαίνει σαν δεύτερο όρισμα κατά την κλήση της συνάρτησης, το πρώτο όρισμα είναι η διεύθυνση του δαίμονα στον οποίο το μέλος θα συνδεθεί. Αν το group δεν υπάρχει ανάμεσα στους δαίμονες του spread τότε δημιουργείται, διαφορετικά αρχίζει η διαδικασία εισόδου του μέλους στο ήδη υπάρχον group. Η συνάρτηση επιστρέφει 0 αν η σύνδεση έγινε με επιτυχία ή ένα από τα ακόλουθα λάθη

- **ILLEGAL\_GROUP.** Το μήνυμα αυτό εμφανίζεται όταν η συμβολοσειρά που επιλέχθηκε σαν όνομα του group δεν είναι σωστή. Συνήθως το λάθος αυτό εμφανίζεται αν δεν έχει δοθεί όνομα για το group ή το μήκος του ονόματος ξεπερνά το μέγιστο επιτρεπτό μέγεθος.
- **ILLEGAL\_SESSION.** Το μήνυμα αυτό εμφανίζεται όταν δεν μπορεί να επιτευχθεί σύνδεση στον δαίμονα, συνήθης αιτία είναι ότι ο δαίμονας είναι ανενεργός.
- **CONNECTION\_CLOSED.** Το μήνυμα εμφανίζεται στην περίπτωση κάποιου επικοινωνιακού λάθους.

Στην περίπτωση που το μέλος που επιθυμεί να συνδεθεί στο group δεν λάβει τον απαιτούμενο αριθμό ψήφων η συνάρτηση δεν θα μπορέσει να ολοκληρώσει την λειτουργία της και το υποψήφιο μέλος θα περιμένει για πάντα.

Για την αποστολή των μηνυμάτων σύνδεσης στο group (JOIN\_REQ), προς όλα τα μέλη του group τα μηνύματα αυτά ενσωματώνονται στην δομή των spread μηνυμάτων και αποστέλλονται μέσω της συνάρτησης του spread SP\_multicast.

Προκειμένου το υποψήφιο μέλος να μπορεί να λάβει τις απαντήσεις από τα μέλη του group που θα ψηφίσουν για την είσοδο του, χρησιμοποιείται η SP\_receive συνάρτηση του spread αφού πρώτα έχει τροποποιηθεί κατάλληλα. Η τροποποιημένη SP\_receive όταν λάβει ένα JOIN\_REQ μήνυμα απαντά με το JOIN\_CMT μήνυμα που είναι ουσιαστικά μία ψήφος για την είσοδο του μέλους στο group. Για την αποστολή του JOIN\_CMT γίνεται και πάλι ενσωμάτωση σε μήνυμα του spread και χρησιμοποιείται ο spread unicast μηχανισμός. Για την αποστολή unicast μηνυμάτων στο spread χρησιμοποιείται η SP\_multicast συνάρτηση αλλά με ορίσματα το όνομα του μέλους και το όνομα του δαίμονα στον οποίο συνδέεται.

Με την συλλογή των απαιτούμενου αριθμού ψήφων, το υποψήφιο μέλος απευθύνεται στον GAUTH. Με την έκδοση από τον GAUTH του GMC η διαδικασία ελέγχου πρόσβασης ολοκληρώνεται. Στην συνέχεια οι δαίμονες ανανεώνουν τις membership πληροφορίες τους και αρχίζει η διαδικασία ανανέωσης του κλειδιού συνόδου του group.

## Εγκατάσταση και λειτουργία

### Εγκατάσταση

Η εγκατάσταση του secure spread ενοποιημένο με τον Bouncer μπορεί να γίνει σε κάποιο από τα λειτουργικά συστήματα Unix ή Linux. Τα βήματα για μία επιτυχή εγκατάσταση είναι τα ακόλουθα.

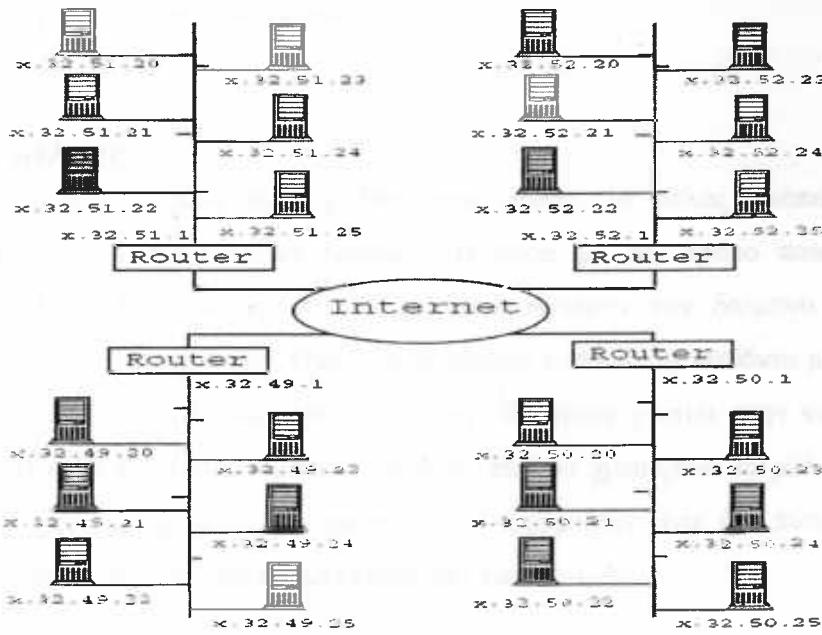
1. Αφού κατεβάσουμε τον πηγαίο κώδικα του ενοποιημένου συστήματος spread-Bouncer (ο πηγαίος κώδικας με τις βιβλιοθήκες του secure spread δεν περιλαμβάνεται), από την επίσημη ιστοσελίδα του Bouncer [60], εγκαθιστούμε αρχικά το σύστημα spread σύμφωνα με το README αρχείο του. Αφού ολοκληρωθεί με επιτυχία η εγκατάσταση του spread.

Εγκαθιστούμε στην συνέχεια τον Bouncer που περιέχεται στο φάκελο με το όνομα gac-0.5.0 σύμφωνα με το README αρχείο του.

2. Για την εγκατάσταση των βιβλιοθηκών του secure spread απαιτεί έκτος από την εγκατάσταση του spread και η εγκατάσταση μίας έκδοσης του OpenSSL toolkit μεγαλύτερης ή ίση της 0.9.4. Το OpenSSL toolkit μπορούμε να το προμηθευτούμε από την επίσημη ιστοσελίδα του OpenSSL project [65] και να το εγκαταστήσουμε σύμφωνα με το README αρχείο του.
3. Στο βήμα αυτό θα γίνει η εγκατάσταση των βιβλιοθηκών του secure spread 2.1.0. Μπορούμε να κατεβάσουμε τον πηγαίο κώδικα του secure spread από την επίσημη ιστοσελίδα του secure spread project [66] και να τον εγκαταστήσουμε.

## Παραμετροποίηση συστήματος

Για να αρχίσει η λειτουργία του secure spread πρέπει να διαμορφωθεί κατάλληλα το configuration αρχείο του spread. Η βασικότερη ρύθμιση σε αυτό το αρχείο είναι η καταχώρηση της τοπολογίας των spread δαιμόνων. Ουσιαστικά εκείνο που καταχωρούμε είναι η IP διεύθυνση ή το domain name του υπολογιστή που έχει εγκατασταθεί ένας δαίμονας και την διεύθυνση του υποδικτύου στο οποίο ανήκει. Στα παρακάτω σχήματα απεικονίζεται η τοπολογία ενός δικτύου και οι συνδεδεμένοι δαίμονες σ' αυτό καθώς και η δομή του configuration file για αυτή την τοπολογία.



```

Spread_Segment x.32.49.255:3333 {
    machine1 x.32.49.20
    machine2 x.32.49.21
    machine3 x.32.49.22
    machine4 x.32.49.23
    machine5 x.32.49.24
    machine6 x.32.49.25
}
Spread_Segment x.32.50.255:3333 {
    machineB1 x.32.50.20
    machineB2 x.32.50.21
    machineB3 x.32.50.22
    machineB4 x.32.50.23
    machineB5 x.32.50.24
    machineB6 x.32.50.25
}
Spread_Segment x.32.51.255:3333 {
    machineC1 x.32.51.20
    machineC2 x.32.51.21
    machineC3 x.32.51.22
    machineC4 x.32.51.23
    machineC5 x.32.51.24
    machineC6 x.32.51.25
}
Spread_Segment x.32.52.255:3333 {
    machineD1 x.32.52.20
    machineD2 x.32.52.21
    machineD3 x.32.52.22
    machineD4 x.32.52.23
    machineD5 x.32.52.24
    machineD6 x.32.52.25
}

```

### *Configuration*

Αναλυτικές οδηγίες για την συμπλήρωση του configuration file του spread μπορούν αντληθούν από αντίστοιχα εγχειρίδια [67].

Αφού συμπληρωθεί το configuration file όλων των δαιμόνων της τοπολογίας θέτουμε τους δαίμονες σε λειτουργία.

Το τελευταίο βήμα είναι η ενεργοποίηση του GAUTH προκειμένου να υπογράφει τα GMC των νεοεισερχόμενων μελών.

## Σύνδεση μέλους

Για την σύνδεση ενός νέου μέλους στο group, το μέλος πρέπει να έχει εγκαταστήσει το σύστημα secure spread – Bouncer με τον τρόπο που έχει ήδη περιγραφεί. Το μέλος δεν χρειάζεται να ενεργοποιήσει τον δαίμονα που έχει εγκατασταθεί στο σύστημα του. Όπως είναι λογικό η σύνδεση κάποιου μέλους στο group θα γίνεται μέσω κάποιων εφαρμογών που θα έχουν χτιστεί στην κορυφή του secure spread – Bouncer συστήματος. Το μόνο που θα χρειάζεται το μέλος για την είσοδό του σε ένα group μέσω των κατάλληλων εφαρμογών είναι ένα πιστοποιητικό δημοσίου κλειδιού που θα έχει προμηθευτεί από κάποια CA.

## Ανασκόπηση

Στην εργασία αυτή παρουσιάσαμε αναλυτικά τα ζητήματα ασφάλειας ενός group communication συστήματος. Περιγράψαμε πρωτόκολλα αυθεντικοποίησης, ελέγχου πρόσβασης και πρωτόκολλα διαχείρισης του μυστικού κλειδιού συνόδου ενός group. Επιπλέον περιγράψαμε κάποια από τα υπάρχοντα group communication συστήματα που πραγματεύονται ζητήματα ασφάλειας, με στόχο να γίνει κατανοητός ο τρόπος με τον οποίο τα παραπάνω πρωτόκολλα μπορούν να ενοποιηθούν στα πλαίσια ενός συστήματος, άλλα και για να δούμε τις υπάρχουσες δυνατές τεχνικές λύσεις στο χώρο των ασφαλών group communication συστημάτων.

Βασικός άξονας της ανάλυσης που κάναμε ήταν η διασφάλιση ενός group communication συστήματος με κατανεμημένα πρωτόκολλα και τεχνικές που δεν παραβιάζουν την κατανεμημένη φύση του υποκείμενου συστήματος.

Το βασικότερο πρόβλημα που παραμένει ανοικτό και αναζητά λύση είναι το πρόβλημα της αυθεντικοποίησης χωρίς την χρήση κάποιας έμπιστης τρίτης οντότητας.

Από τα υπάρχοντα ασφαλή group communication συστήματα επιλέξαμε το secure spread ενοποιημένο με το εργαλείο ελέγχου πρόσβασης Bouncer. Στο σύστημα αυτό τα βασικά ζητήματα που αναζητούν λύση είναι η αυθεντικοποίηση χωρίς την χρήση έμπιστης τρίτης οντότητας και η έλλειψη μηχανισμού ανάκλησης των GMCs.

## Περίληψη

Η παρούσα εργασία πραγματεύεται προβλήματα ασφάλειας των συστημάτων επικοινωνίας ομάδας.

Λέγοντας σύστημα επικοινωνίας ομάδας, εννοούμε ένα σύγχρονο σύστημα επικοινωνίας τόσο για WAN όσο και LAN. Το σύστημα θα πρέπει να είναι ανεξάρτητο από τεχνολογίες μετάδοσης όπως το IP multicast.

Το μοντέλο απειλών που υιοθετούμε δεν πραγματεύεται απειλές που προέρχονται από διεφθαρμένα μέλη του group άλλα προσπαθεί να διασφαλίσει την επικοινωνία της ομάδας από οποιονδήποτε εξωτερικό επίβουλο.

Τα ζητήματα ασφάλειας που πραγματεύομαστε κατά συνέπεια σε αυτή την εργασία είναι η αυθεντικοποίηση των υποψήφιων μελών, οι μηχανισμοί έλεγχου πρόσβασης ενός υποψήφιου μέλους στο group και τέλος η συμφωνία ενός μυστικού κλειδιού συνόδου μεταξύ των μελών του group, έτσι ώστε ή επικοινωνία μεταξύ των μελών να μην μπορεί να υποκλαπεί. Λόγο της κατανεμημένης φύσης των group communication συστημάτων επικεντρώνουμε την προσοχή μας σε κατανεμημένα πρωτόκολλα και αρχιτεκτονικές που δεν παραβιάζουν την φύση του υποκείμενου συστήματος.

Στο πρόβλημα της αυθεντικοποίησης παρουσιάζουμε και αναλύουμε τις υπάρχουσες λύσεις που συνοψίζονται σε πρωτόκολλα που κάνουν χρήση μίας on-line έμπιστης τρίτης οντότητας, πρωτόκολλα που κάνουν χρήση μίας έμπιστης τρίτης οντότητας μόνο κατά την φάση εγγραφής νέου χρήστη και το πρωτόκολλο PGP που δεν κάνει χρήση μίας έμπιστης τρίτης οντότητας.

Από τα πρωτόκολλα αυτά σαν πλέον κατάλληλα κρίνουμε τα πρωτόκολλα που κάνουν χρήση μίας έμπιστης τρίτης οντότητας μόνο κατά την φάση εγγραφής νέου χρήστη. Τα πρωτόκολλα αυτά φαίνονται τα καταλληλότερα στην περίπτωση σχετικά μικρών groups όπως για παράδειγμα τα group communication συστήματα μεταξύ των εργαζομένων μίας πολυεθνικής εταιρίας ή των μελών ενός πανεπιστημίου. Ορισμένα προβλήματα των συστημάτων που στηρίζονται σε μία έμπιστη τρίτη οντότητα κατά την φάση εγγραφής νέου χρήστη, είναι η ανάκληση “πιστοποιητικών” και η επεκτασιμότητα αυτών των συστημάτων.

Το πρωτόκολλο PGP αν και κατανεμημένο δεν κρίνεται αρκετά αξιόπιστο ώστε να χρησιμοποιηθεί σε συστήματα με υψηλές απαιτήσεις ασφάλειας.

Τα πρωτόκολλα που κάνουν χρήση μίας on-line έμπιστης τρίτης οντότητας και ειδικότερα αυτά που στηρίζονται στην χρήση ψηφιακών πιστοποιητικών, φαίνεται ότι αποτελούν την μόνη επιλογή για τα group communication συστήματα με χρήστες που βρίσκονται διάσπαρτοι στο διαδίκτυο και οι οποίοι δεν μπορούν να εξασφαλίσουν διαπιστευτήρια από μία κοινή για όλους έμπιστη τρίτη οντότητα.

Το δεύτερο κρίσιμο ζήτημα ασφάλειας των group communication συστημάτων είναι ο έλεγχος πρόσβασης. Στον έλεγχο πρόσβαση οι αρχιτεκτονικές μπορούν να διαχωριστούν σε δύο βασικές κατηγορίες, σ' αυτές που κάνουν χρήση στατικών ACL και σε αυτές που υιοθετούν κάποιο σχήμα ψηφοφορίας. Οι αρχιτεκτονικές που στηρίζονται στην ψηφοφορία μεταξύ των μελών πλεονεκτούν έναντι των στατικών ACL μίας και μπορούν να λάβουν απόφαση πρόσβασης για μέλη για τα οποία δεν διέθεταν κάποια προηγούμενη πληροφορία.

Στην παρούσα εργασία περιγράφουμε και αναλύουμε δύο σχήματα ψηφιακών υπογραφών. Το ένα από αυτά στηρίζεται στην χρήση πιστοποιητικών δημοσίου κλειδί ενώ το δεύτερο είναι ένα ID – based σχήμα. Το ID – based σχήμα είναι το πλέον κατάλληλο για τον έλεγχο πρόσβασης σε ένα group communication σύστημα μίας και ακολουθεί μία κατανεμημένη αρχιτεκτονική και υλοποιεί ένα ευέλικτο μηχανισμό ανάκλησης της ιδιότητας μέλους.

Το τελευταίο κρίσιμο ζήτημα ασφάλειας των group communication συστημάτων είναι η καθιέρωση και η διαχείριση του μυστικού κλειδιού συνόδου ενός group. Για την διαχείριση του μυστικού κλειδιού συνόδου έχουν προταθεί κεντρικοποιημένες, μη κεντρικοποιημένες και κατανεμημένες αρχιτεκτονικές. Οι κατανεμημένες αρχιτεκτονικές παρουσιάζουν το μεγαλύτερο ενδιαφέρον για τα group communication συστήματα μίας και δεν παραβιάζουν την κατανεμημένη φύση τους. Οι κατανεμημένες αρχιτεκτονικές μπορούν να διαχωριστούν σε αυτές που παρέχουν αυθεντικοποίηση του μυστικού κλειδιού συνόδου και σε αυτές που δεν παρέχουν τέτοια δυνατότητα.

Για να γίνει κατανοητός ο τρόπος με τον οποίο τα παραπάνω πρωτόκολλα και αρχιτεκτονικές αυθεντικοποίησης, ελέγχου πρόσβασης και συμφωνίας του μυστικού κλειδιού συνόδου μπορούν να ενοποιηθούν στα πλαίσια ενός group communication συστήματος, δίνουμε την περιγραφεί τεσσάρων συστημάτων, εστιάζοντας κυρίως στις απαιτήσεις ασφάλειας και στις επικοινωνιακές τους ιδιότητες.

Με την μελέτη των παραπάνω συστημάτων και την βοήθεια των συμπερασμάτων που προέκυψαν κατά την ανάλυση των απαιτήσεων ασφάλειας δημιουργούμε μία

λίστα κριτηρίων για την επιλογή μία ολοκληρωμένης αρχιτεκτονικής ασφάλειας για ένα group communication σύστημα.

Βάση των κριτηρίων που αποτελούν το απόσταγμα της συνολικής μας μελέτης στο χώρο της ασφάλειας των group communication συστημάτων και έχοντας κατά νου τις υπάρχουσες υλοποιήσεις στο χώρο αυτό. Επιλέγουμε το σύστημα Secure Spread ενοποιημένο με το εργαλείο πρόσβασης Bouncer σαν την πλέον κατάλληλη λύση.

Το Secure Spread είναι ένα σύστημα που ακολουθεί μία client server αρχιτεκτονική, έχει υιοθετήσει ένα ευέλικτο σχήμα συμφωνίας και διαχείρισης του μυστικού κλειδιού συνόδου με κατανεμημένα συνεργατικά πρωτόκολλα άλλα δεν έχει υλοποιήσει κάποια αρχιτεκτονική για τον έλεγχο πρόσβασης. Το εργαλείο Bouncer αποτελεί ουσιαστικά μία υλοποίηση της αρχιτεκτονικής έλεγχου πρόσβασης μέσω ψηφοφορίας που κάνει χρήση πιστοποιητικών δημόσιου κλειδιού.

Η ενοποίηση του Secure Spread με τον Bouncer μας παρέχει ένα αξιόπιστο σύστημα άμεσα διαθέσιμο για χρήση, που καλύπτει σε μεγάλο βαθμό τις απαιτήσεις ασφάλειας ενός κατανεμημένου group communication συστήματος. Τα βασικότερα ελαττώματα της ενοποιημένης αρχιτεκτονικής είναι η έλλειψη ενός μηχανισμού ανάκλησης πιστοποιητικών μέλους καθώς και το γεγονός ότι η αυθεντικοποίηση των υποψηφίων μελών γίνεται με χρήση πιστοποιητικών δημόσιου κλειδιού, κάτι που μας οδηγεί στην ανάγκη ύπαρξης μία on – line έμπιστης τρίτης οντότητας.

Τέλος πρέπει να αναφέρουμε ότι στην παρούσα εργασία έγινε η περιγραφή ενός συστήματος που πραγματεύεται τις εκ' των έσω απειλές. Το σύστημα που περιγράψαμε είναι το Secure Ring. Αν και από την αρχή τονίσαμε ότι η παρούσα εργασία υιοθετεί ένα μοντέλο απειλών που δεν πραγματεύεται τις εκ' των έσω απειλές θεωρήσαμε την περιγραφή του Secure Ring αναγκαία έτσι ώστε ο αναγνώστης να έχει μία ολοκληρωμένη οπτική του χώρου της ασφάλεια των group communication συστημάτων .

# Executive Summary

The present work treats problems of safety of group communication systems.

By group communication system, we mean a synchronous communication system, so for WAN as for LAN. The system should be independent from technologies of transmission as the IP multicast.

The model of threats that we adopt doesn't treat threats that emanate from corrupted group members but it tries to ensure the communication of group from anyone exterior insidious.

The issues of safety which we accordingly treat in this work are the authentication of candidate members, the mechanisms of access control of group's candidate members and finally the agreement of session secret key between the group members, such as the communication between the group members cannot wiretapped. Because of the distributed nature of group communication systems we focus our attention in distributed protocols and architectures such us do not force the nature of amenable system.

In authentication problem we present and analyze the current solutions, summarised in protocols that make use of an on-line trusted third party, protocols that make use of a trusted third party only at the new user registration phase and protocol PGP that does not make use of a trusted third party.

From this protocols as best suitable we judge the protocols that make use of a trusted third party only at the new user registration phase. This protocols appear more suitable in the case of relatively small groups, as group communication systems between employees of a multinational company, or the members of a university. Certain problems of those systems are the lack of a revocation mechanism and stretchable problems of these systems.

Protocol PGP even if it is a distributed system it is not consider reliable enough so it can be used in systems with high safety requirements.

Protocols that make use of an on-line trusted third party and more specifically the protocols which are based on the use of digital certificates, appear as the only choice for the group communication systems with users scattered in the internet who aren't able to ensure credentials from a common trusted third party.

The second critical issue of group communication systems safety is the access control. The access control architectures can be separated in two basic categories, in

those that make use of static ACL and in those that adopt some form of voting. The architectures that are based on the voting between the group members have an advantage against the static ACL architectures because they can take access decisions for members who did not allocate certain previous information.

In the present work we describe and analyze two schemes of digital signatures. The one of them is based on the use of public key certificates while the second is a ID-based architecture. The ID – based scheme is most suitable for the access control of a group communication system because it follows a distributed architecture and implements a flexible membership revocation mechanism.

The last critical safety issue of a group communication systems is the group secret session key establishment and management. For the secret session key management have been proposed centralized, not centralized and distributed architectures. The distributed architectures are the most interesting for group communication systems because they don't force their distributed nature. The distributed architectures can be separated in those that provide authentication of secret session key and those that don't have this capability.

In order to become comprehensible, the way in which the above authentication, access control and secret session key agreement protocols and architectures can be unified in the frame of group communication system, we describe four systems, focusing mainly in their safety requirements and their communication attributes.

With the study of systems above and the help of conclusions that resulted from the analysis of safety requirements, we create a list of criteria, which can help the choice of a completed architecture for a secure group communication system.

Based on the above criteria which constitutes the distillation of our total study in the secure group communication systems and having in mind the existing concretisations in this space. We select the Secure Spread system unified with the Bouncer access control toolkit as the most suitable solution.

The Secure Spread is a system that follows a client server architecture, it has adopted a flexible secret session key agreement and management framework using distributed collaborative protocols. Unfortunately the Secure Spread has not implemented any architecture for the access control problem. The Bouncer toolkit constitutes substantially an implementation of access control architecture via voting making use of public key certificates.



The unification of Secure Spread with Bouncer provides us with a reliable system immediately available for use, that covers in large percent the safety requirements of a distributed group communication system. The basic disadvantage of unified architecture are the lack of a membership revocation mechanism as well as the usage of public key certificates for candidate members authentication, something that leads us to the need of on-line trusted third party existence.

Finally we should report that in the present work we described a system that treats the interior threats. The system we described is the Secure Ring. Even if we marked from the beginning that the present work adopts a model of threats that don't treat the interior threats, we considered the description of Secure Ring necessary so as the reader could have a completed vision of secure group communication systems area.



## Βιβλιογραφία

- [1] D. Denning, and G. Sacco, Timestamps in key distribution protocols, Communications of the ACM 24,8 (1981), pp 533-536, 1981
- [2] Woei-Jiunn Tsaurf Shi-Jinn Horngt, Chia-Ho Chen. An authentication-combined access control scheme using a geometric approach in distributed systems. Proceedings of the 1997 ACM symposium on Applied computing, April 1997
- [3] I-Lung Kao and Randy Chow. An Efficient and Secure Authentication Protocol Using Uncertified Keys. ACM SIGOPS Operating Systems Review, pp 14-21, 1995
- [4] J. Daemen and V. Rijmen. The Rijndael Block Cipher. In AES Proposal, 2000 Available at <http://csrc.nist.gov/encryption/aes/>.
- [5] H. Krawczyk, M. Bellare, and R. Canetti. HMAC: Keyed-Hashing for Message Authentication. Internet RFC 2104, February 1997
- [6] ITU-Recommendation X.509. The Directory - Authentication Framework, 1998
- [7] G. Ateniese, M. Steiner, and G. Tsudik. New multi-party authentication services and key agreement protocols. IEEE JSAC, special issue on Secure Communication, May 2000
- [8] M. Steiner, G. Tsudik, and W. Waidner. Key agreement in dynamic peer groups. IEEE Transactions on Parallel and Distributed Systems, 2000
- [9] T. Leighton and S. Micali. Secret-key Agreement without Public-Key Cryptography. In Proceedings of Crypto 93, pp 456–479, August 1994
- [10] S. Floyd, V. Jacobson, C. Liu, S. McCanne, and L. Zhang. A Reliable Multicast Framework for Light-weight Sessions and Application Level Framing. IEEE/ACM Transactions on Networking, pp 784–803, December 1997



- [11] O. Rodeh, K. Birman, M. Hayden, Z. Xiao, and D. Dolev. Ensemble security. Tech. Rep. TR98-1703, Cornell University, Department of Computer Science, September 1998
- [12] M. A. Hiltunen, R. D. Schlichting, and C. Ugarte. Enhancing survivability of security services using redundancy. in Proceedings of The International Conference on Dependable Systems and Networks, June 2001
- [13] Cox, M. J., Engelschall, R. S., Henson, S., Laurie, B., Young, E. A., and Hudson, T. J. Open SSL, 2000
- [14] Diffie, W. and Hellman. M. New directions in cryptography. IEEE Transactions on information Theory, IT-22:644–654, November 1976
- [15] Bellare, M. and Rogaway, P. Entity authentication and key distribution. IEEE Computer Society Press, pp 232–249, 1993
- [16] Canetti, R., Garay, J., Itkis, G., Micciancio, D., M. Naor, and B. Pinkas. Multicast security: A taxonomy and some efficient constructions. In INFOCOM, volume 2, pp 708–716, March 1999
- [17] Wallner, D., Harder, E., and Agee, R. Key management for multicast: Issues and architectures. Internet Draft draft-wallner-key-arch-01.txt, IETF, Network Working Group, September 1998
- [18] Wong, C.K., Gouda, M., and Lam. S.S. Secure group communication using key graphs. In SIGGCOM, New York, USA, September 1998
- [19] Rodeh, O., Birman, K. P., and Dolev, D. Optimized group rekey for group communication systems. In Symposium on Network and Distributed System Security, Febuary 2000



- [20] National Institute for Standards and Technology (NIST). The Keyed-Hash Message Authentication Code (HMAC), 2002  
<http://csrc.nist.gov/publications/fips/index.html>
- [21] Ellison, R. J., Fisher, D. A., Linger, R. C., Lipson, H. F., Longstaff, T., and Mead, N. R. Survivable network systems: An emerging discipline. Technical Report. Software Engineering Institute, Carnegie Mellon University, November 1997
- [22] Jennifer G. Steiner, Clifford Neuman, Jeffrey I. Schiller. Kerberos: An Authentication Service for Open Network Systems, January 1988
- [23] Re k Molva, Gene Tsudik, Els Van Herreweghen and Stefano Zatti. KryptoKnight Authentication and Key Distribution System, November 1992
- [24] National Institute of Standards and Technology (NIST). Entity Authentication Using Public Key Cryptography, February 1997
- [25] Adi Shamir. Identity based cryptosystems and signature schemes. Springer-Verlag New York, 1985
- [26] Wen-Her Yang, and Hun-Min Sun. An Authentication Protocol Without Trusted Third Party. IEEE Communications Letters, vol. 1, No. 3, MAY 1997
- [27] P. Zimmermann. Pretty Good Privacy User's Guide. Volume I and II. Distributed with the PGP software, June 1993
- [28] J. Feigenbaum, M. Blaze, J. Lacy. Decentralized Trust Management. Proceedings of the IEEE Conference on Security and Privacy. Oakland, May 1996
- [29] Y. Kim, D. Mazzocchi, and G. Tsudik. Admission control in peer groups. In IEEE International Symposium on Network Computing and Applications (NCA), April 2003



- [30]Y. Desmedt and Y. Frankel. Threshold cryptosystems. In CRYPTO'89, August 2000
- [31]Y. Frankel, P. Gemmell, P. MacKenzie, and M. Yung. Optimal-resilience proactive public-key cryptosystems. In FOCS'97, pp 384–393, 1997
- [32]J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang. Providing robust and ubiquitous security support for MANET. In IEEE ICNP'01, 2001
- [33]K. Ohta, S. Micali, and L. Reyzin. Accountable - subgroup multisignatures. In ACM CCS, pp 245–254, November 2001
- [34]C. Schnorr. Efficient signature generation by smart cards. Journal of Cryptology, pp 161–174, 1991
- [35]D. Chaum and E. van Heyst. Group signatures. In EUROCRYPT' 91, May 1991
- [36]J. Camenisch. Efficient and generalized group signatures. In EUROCRYPT'97, pp 465–479, May 1997
- [37]G. Ateniese and G. Tsudik. Some Open Issues and New Directions in Group Signatures. In Financial Cryptography, 1999
- [38]Harney, H. and Muckenheim, C. Group Key Management Protocol (GKMP) Architecture. RFC 2094, 1997
- [39]Wallner, D., Harder, E., and Agee, R. Key Management for Multicast: Issues and Architectures. RFC 2627, 1999
- [40]Wong, C. K., Gouda, M. G., and Lam, S. S. Secure group communications using key graphs. IEEE/ACM Trans. Netw. 8, 1, pp 16–30, February 2000



- [41] McGrew, D. A. and Sherman, A. T. Key establishment in large dynamic groups using one way function trees. Tech. Rep. No. 0755 , TIS Labs at Network Associates, Inc., Glenwood, Md, May 1998
- [42] Canetti, R., Garay, J., Itkis, G., Micciancio, D., Naor, M., and Pinkas, B. Multicast Security: A Taxonomy and Some Efficient Constructions. In Proceedings of the IEEE INFOCOM. Vol. 2. (New York, N.Y.), pp 708–716, 1999
- [43] Canetti, R., Malkin, T., and Nissim, K. Efficient communication storage tradeoffs for multicast encryption. In Advances in Cryptology—EUROCRYPT '99, J. Stern, Ed. Lectures Notes in Computer Science, vol. 1599. Springer-Verlag, New York, pp. 459–474, 1999
- [44] LI, M., Poovendran, R., and Berenstein, C. Optimization of key storage for secure. In Proceedings of the 35th Annual Conference on Information Sciences and Systems (CISS), March 2001
- [45] Waldvogel, M., Caronni, G., Sun, D., Weiler, N., and Plattner, B. The VersaKey framework: Versatile group key management. IEEE J. Sel. Areas Commun, pp 1614–1631, August 1999
- [46] Perrig, A., Song, D., and Tsigas, J. D. ELK, A new protocol for efficient large-group key distribution. In Proceedings of the IEEE Symposium on Security and Privacy. (Oakland, Calif., May). IEEE Computer Society Press, Los Alamitos, Calif, 2001
- [47] Mitra, S. Iolus: A framework for scalable secure multicasting. In Proceedings of the ACM SIGCOMM. Vol. 27, 4 (New York, Sept.) ACM, New York, pp 277–288, 1997
- [48] Dondeti, L., Mukherjee, S., and Samal, A. Scalable secure one-to-many group communication using dual encryption. Comput. Commun. 23, pp 1681–1701, November 1999



[49]Decleene, B., Dondeti, L., Griffin, S., Hardjono, T., Kiwior, D., Kurose, J., Towsley, D., Vasudevan, S., and Zhang, C. Secure group communications for wireless networks. In Proceedings of the MILCOM, June 2001

[50]Rafaeli, S. and Hutchison, D. Hydra: A decentralised group key management. In Proceedings of the 11th IEEE International WETICE: Enterprise Security Workshop, A. Jacobs, Ed. (Pittsburgh, Pa., June). IEEE Computer SocietyPress, Los Alamitos, 2002

[51]Wen-Her Yang and Shiu-Pyng Shieh. Secure key agreement for group communications. International Journal of Network Management, pp 365-374, 2001

[52]Li Ming, Gu Dawu, Wang Yong and Bai Yingcai. Research on Authenticated Key Agreement in Group Settings. Infosecu '04, November 2004

[53]Yongdae Kim, Adrian Perrig, Gene Tsudik. Tree – Based Group Key Agreement. ACM Transactions on Information and System Security, Vol. 7, No. 1, pp 60-96, February 2004

[54]J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang. Providing robust and ubiquitous security support for MANET. In IEEE 9th International Conference on Network Protocols (ICNP), 2001

[55]Nitesh Saxena, Gene Tsudick, Jeong Hyun Yi. Admission Control in Peer-to-Peer: Design and Evaluation. ACM, 2003

[56]J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang. Providing robust and ubiquitous security support for MANET. In IEEE 9th International Conference on Network Protocols (ICNP), 2001

[57]H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang. Self-securing Ad Hoc Wireless Networks. In Seventh IEEE Symposium on Computers and Communications (ISCC '02), 2002

[58]J. Kong, H. Luo, K. Xu, D. L. Gu, M. Gerla, and S. Lu. Adaptive Security for Multi-level Ad-hoc Networks. In Journal of Wireless Communications and Mobile Computing (WCMC), volume 2, pp 533–547, 2002

[59]R. Gennaro, S.Jarecki, H.Krawczyk, and T.Rabin. Robust Threshold DSS Signatures. In U. Maurer, editor, EUROCRYPT '96, number 1070 in LNCS, pp 354–371, 1996

[60]<http://sconce.ics.uci.edu/gac/download.html>

[61]<http://sconce.ics.uci.edu/cliques/>

[62]D. Steer, L. Strawczynski, W. Diffie and M. Wiener. A Secure Audio Teleconference System. Crypto'88, 1988

[63]M. Burmester and Y. Desmedt. A Secure and Efficient Conference Key Distribution System. EUROCRYPT '94, 1994

[64] M. Steiner, G. Tsudik and M. Waidner. Key Agreement in Dynamic Peer Groups. IEEE Transaction on Parallel and Distributed Systems, August 2000

[65]<http://www.openssl.org/>

[66][http://www.dsn.jhu.edu/research/group/secure\\_spread/](http://www.dsn.jhu.edu/research/group/secure_spread/)

[67]Jonathan R. Stanton. A Users Guide to Spread Version 0.11, October 2002

[68]Boldyreva, A. Efficient threshold signatures, multisignatures and blind signatures based on the gap-diffie-hellman-group signature scheme. In: Proceedings of International Workshop on Practice and Theory in Public Key Cryptography. Volume 2567 of LNCS, pp 31-46, 2003

[69]Boneh, D., Lynn, B., Shacham, H. Short Signatures from the Weil Pairing. In Boyd, C., ed.: ASIACRYPT '01. Number 2248 in LNCS, pp 514-532, 2001

- [70]Gennaro, R., Jarecki, S.; Krawczyk, H., Rabin, T. Secure Distributed Key Generation for Discrete-Log based Cryptosystems. In: Eurocrypt 99, 1999
- [71]Balfanz, D., Durfee, G., Shankar, N., Smetters, D., Staddon, J., Wong, H.C. Secret Handshakes from Pairing-Based Key Agreements. In: IEEE Symposium on Security and Privacy, pp 180-196, 2003
- [72]Cha, J., Cheon, J. An ID-based signature from Gap-Diffie-Hellman Groups. In: Proceedings of International Workshop on Practice and Theory in Public Key Cryptography. Volume 2567 of LNCS, pp 18-30, 2003
- [73]Nitesh Saxena, Gene Tsudik and Jeong Hyun Yi. Identity-based Access Control for Ad Hoc Groups. In International Conference on Information Security and Cryptology (ICISC), 2004
- [74]A. Herzberg, S. Jarecki, H. Krawczyk, and M. Yung. Proactive Secret Sharing, Or How To Cope With Perpetual Leakage. In D. Coppersmith, editor, CRYPTO '95, number 963 in LNCS, pp 339–352, 1995

