

**ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ :**  
**ΑΚΡΩΤΗΡΙΑΝΑΚΗΣ ΜΙΧΑΗΛ**

# **ΑΝΑΛΥΣΗ SIP ΠΡΩΤΟΚΟΛΛΟΥ**

---

## ΠΕΡΙΕΧΟΜΕΝΑ

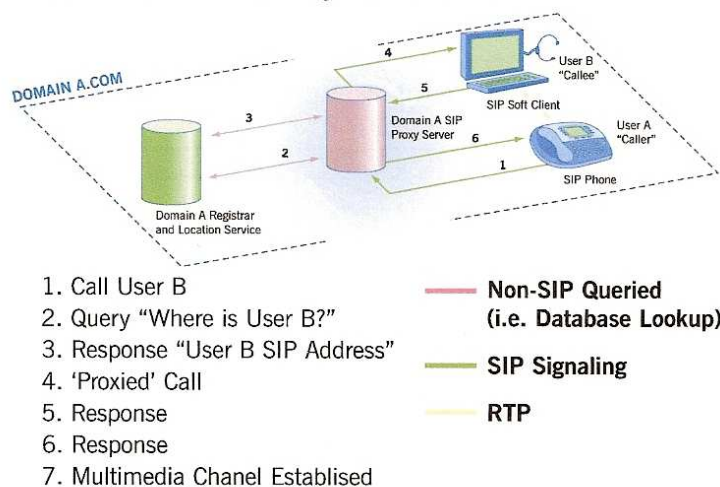
1. Εισαγωγή στις σύγχρονες τεχνολογίες με βάση το VoIP.....	σελ.3
2. Εισαγωγή στο SIP.....	σελ.6
3. Περίληψη SIP λειτουργικότητας.....	σελ.6
4. Ορολογία και περίληψη λειτουργίας.....	σελ.8
5. Δομή του Πρωτόκολλου.....	σελ.15
6. Ορισμοί.....	σελ.17
7. SIP μηνύματα.....	σελ.22
SIP αιτήσεις.....	σελ.22
SIP απαντήσεις.....	σελ.24
Κύρια πεδία.....	σελ.25
Σώματα.....	σελ.29
8. Γενική Συμπεριφορά Φορέα Χρήστη.....	σελ.30
Συμπεριφορά UAC.....	σελ.30
Συμπεριφορά UAS.....	σελ.42
Server προσανατολισμού.....	σελ.47
9. Ακύρωση αίτησης.....	σελ.48
Συμπεριφορά φορέα.....	σελ.49
Συμπεριφορά Server.....	σελ.49
10. Καταγραφή.....	σελ.50
Εισαγωγή στην καταγραφή.....	σελ.50
Δομή της αίτησης καταγραφής.....	σελ.51
Επεξεργασία της αίτησης REGISTER.....	σελ.52
11. Επίλογος.....	σελ.54

## ΕΙΣΑΓΩΓΗ ΣΤΟ VoIP

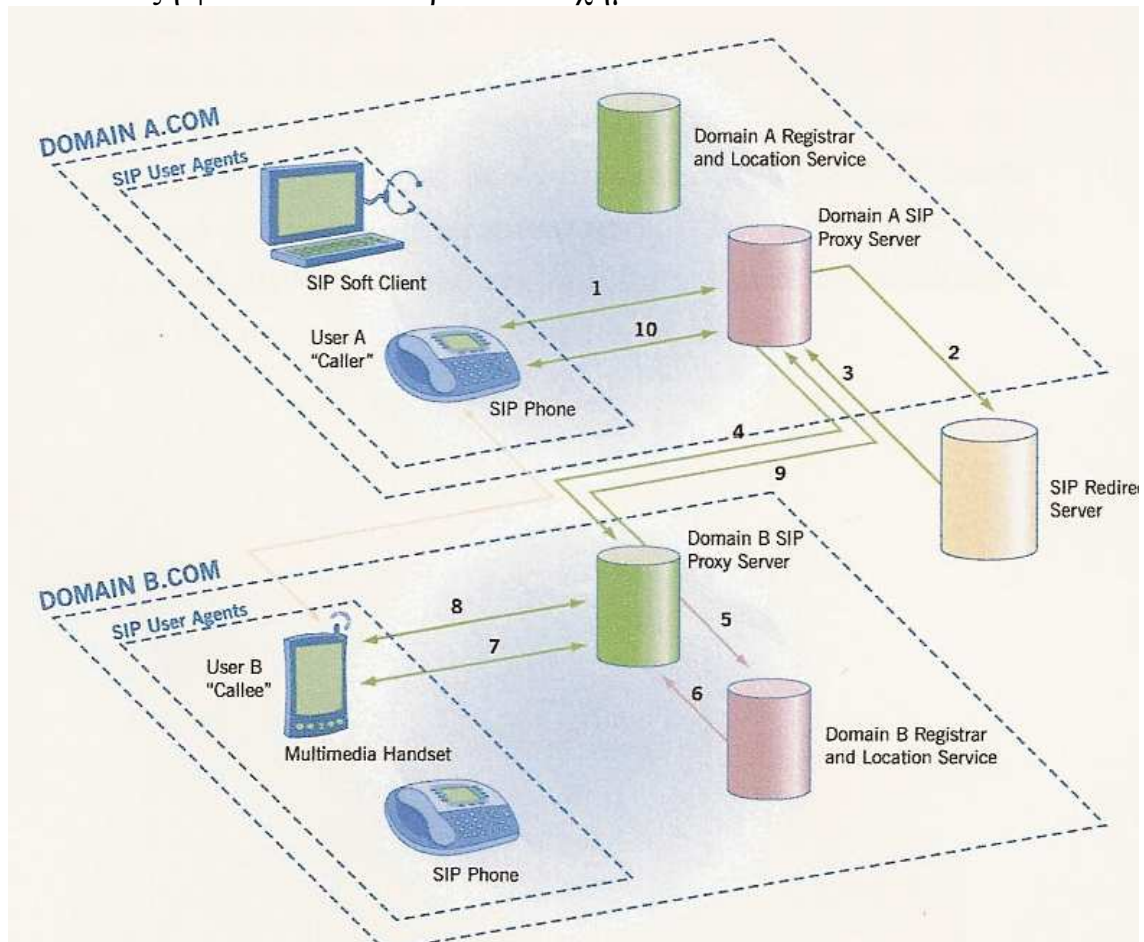
Η αλματώδης ανάπτυξη των τηλεπικοινωνιακών τεχνολογιών αναμφίβολα άνοιξε νέους ορίζοντες στην σύγχρονη τηλεφωνία, με μια πληθώρα εφαρμογών να την ακολουθούν. Ίσως η πιο άθουσα και εξελισσόμενη τεχνολογία είναι αυτή της τηλεφωνίας μέσω internet ή αλλιώς VoIP (Voice over IP). Ουσιαστικά πρόκειται για μια τεχνολογία που μας επιτρέπει την πραγματοποίηση τηλεφωνικών συνδιαλέξεων κάνοντας χρήση γραμμών ευρείας ζώνης. Το σημαντικότερο πλεονέκτημα αυτής της τεχνολογίας είναι ότι μειώνεται δραστικά το κόστος των τηλεφωνικών κλήσεων αφού στην ουσία ο χρήστης πληρώνει ένα πάγιο ανεξαρτήτως τον αριθμό των τηλεφωνικών κλήσεων που κάνει, ενώ σε πολλές περιπτώσεις αυτό το κόστος είναι μηδαμινό. Ταυτόχρονα παρέχει στους χρήστες ασφάλεια στην επικοινωνία, ευελιξία στις συνδιαλέξεις αμεσότητα και αξιοπιστία στην χρήση και πληθώρα εφαρμογών μέσω IP πρωτόκολλων. Στην συνέχεια θα παρουσιάσω τα βασικότερα πλεονεκτήματα αυτής της τεχνολογίας αναλυτικότερα καθώς και ένα μοντέλο υλοποίησης μιας τέτοιας συνδιάλεξης. Όπως ανέφερα νωρίτερα ένα από τα σημαντικότερα ατού που έχει να προσφέρει αυτή η τεχνολογία είναι το χαμηλό κόστος. Αυτό πραγματοποιείται διότι στην IP τηλεφωνία γίνεται απλά χρήση της broadband γραμμής (ADSL και όχι PSTN) που ήδη διαθέτει ο χρήστης έτσι ουσιαστικά ανάλογα με τον παροχέα υπηρεσιών ο χρήστης είναι σε θέση να πραγματοποιήσει απεριόριστο αριθμό κλήσεων πληρώνοντας απλά μια συνδρομή στο παροχέα. Ουσιαστικά δηλαδή ο χρήστης δεν δεσμεύεται από τον αριθμό και από την διάρκεια των κλήσεων αλλά μόνο από την διάρκεια παροχής υπηρεσιών από τον εκάστοτε προμηθευτή αυτών. Φυσικά εκτός από την broadband γραμμή και τον παροχέα υπηρεσιών απαιτείται και επιπλέον εξοπλισμός που όμως το κόστος του κυμαίνεται ανάλογα με τις ανάγκες του χρήστη. Σε αυτόν το εξοπλισμό περιλαμβάνονται τα IP τηλέφωνα ή αναλογικοί τηλεφωνικοί αντάπτορες καθώς και προγράμματα (softphones) για υπολογιστή ουσιαστικά έχοντας ακριβώς την ίδια αποτελεσματικότητα με τα παραπάνω και είναι πολύ πιο οικονομικά. Άρα γίνεται αντιληπτό ότι και ακόμα το κόστος του εξοπλισμού είναι κυμαινόμενο και αρκετές φορές μηδαμινό μιας και τα softphones μπορεί κάποιος να τα προμηθευτεί σχεδόν δωρεάν. Απ' τα παραπάνω γίνεται αντιληπτό ότι σε περίπτωση που κάποιος δεν θέλει να προχωρήσει στην αγορά IP τηλεφώνων μπορεί να κάνει χρήση του υπολογιστή του ως IP τηλέφωνο μέσω των softphones αν φυσικά διαθέτει μικρόφωνο και ηχεία.

Η παρούσα πτυχιακή μελέτη έχει ως σκοπό να παρουσιάσει και να αναλύσει ένα από τα πιο αναπτυσσόμενα πρωτόκολλα για την ανάπτυξη

VoIP εφαρμογών, το πρωτόκολλο SIP. Το SIP (Session Initiation Protocol) πρόκειται για ένα πρωτόκολλο σηματοδότησης για έναρξη, τροποποίηση και τερματισμό VoIP συνδιαλέξεων ενός ή περισσότερων ατόμων. Αφού όπως ανέφερα πιο πριν η VoIP είναι μια τεχνολογία βασισμένη σε IP το SIP έχει αρχιτεκτονική βασισμένη σε HTTP (Hyper Text Transfer Protocol). Ωστόσο η ανάπτυξή του είναι τέτοια ώστε σήμερα να καθιερώνεται ως ένα από τα μεγαλύτερα VoIP πρότυπα. Στην συνέχεια θα παρουσιάσω παραδείγματα πραγματοποίησης SIP συνδιάλεξης μεταξύ χρηστών που βρίσκονται σε ίδια ή και διαφορετικά τηλεπικοινωνιακά πεδία. Με τον όρο ίδιο τηλεφωνικό πεδίο εννοώ ότι οι χρήστες ανήκουν στον ίδιο ISP παροχέα άρα και στο ίδιο δίκτυο. Στην περίπτωση που οι χρήστες ανήκουν στο ίδιο πεδίο, η συνδιάλεξη πραγματοποιείται όπως φαίνεται και στο παρακάτω σχήμα. Στο παρακάτω παράδειγμα ο χρήστης A χρησιμοποιεί ένα SIP τηλέφωνο ενώ ο χρήστης B υπολογιστή με softphone λογισμικό που του παρέχει δυνατότητες μετάδοσης φωνής και εικόνας. Με την ενεργοποίηση και των δύο συσκευών οι χρήστες κατοχυρώνουν την διαθεσιμότητά τους και τις IP διευθύνσεις τους μέσω του SIP Proxy Server στο ISP δίκτυο. Ας υποθέσουμε ότι ο χρήστης A επιθυμεί να συνομιλήσει με τον χρήστη B. Αρχικά ο χρήστης A στέλνει μήνυμα στον SIP Proxy Server ότι επιθυμεί συνομιλία με το χρήστη B. Ακολούθως ο SIP Proxy Server ζητά και λαμβάνει την IP διεύθυνση του χρήστη B από τον Server τοποθεσίας και καταχώρησης. Μετά ο SIP Proxy Server ενημερώνει τον χρήστη B την επιθυμία του χρήστη A για εγκατάσταση συνομιλίας και αναλόγως ο χρήστης B ενημερώνει τον SIP Proxy Server ότι δέχεται την πρόσκληση και ότι είναι έτοιμος να δεχτεί το μήνυμα. Τότε ο SIP Proxy Server επικοινωνεί με τον χρήστη A εγκαθιστώντας την SIP συνδιάλεξη και έτσι οι χρήστες δημιουργούν μεταξύ τους μια συνδιάλεξη point to point. Το σχεδιάγραμμα μιας τέτοιας συνδιάλεξης όπως περιέγραψα φαίνεται παρακάτω σχεδιάγραμμα.



Ωστόσο παρόμοια συνδιάλεξη μπορεί να επιτευχθεί μεταξύ χρηστών που ανήκουν σε διαφορετικά πεδία. Μια τέτοιου τύπου συνδιάλεξη θα αναλύσω παρακάτω. Υποθέτουμε ότι ο χρήστης A επιθυμεί επικοινωνία με τον χρήστη B. Αρχικά ο χρήστης A εκφράζει την επιθυμία αυτή στον SIP Proxy Server. Στην συνέχεια ο SIP Proxy Server εντοπίζει τον χρήστη B μέσω ενός SIP Redirect Server αφού δεν γνωρίζει σε ποιο πεδίο βρίσκεται ο χρήστης B. Ο SIP Redirect Server δίνει τα στοιχεία του χρήστη B στον SIP Proxy Server του πεδίου A και στην συνέχεια αυτός επικοινωνεί με τον SIP Proxy Server του πεδίου B ο οποίος μεταφέρει την πρόσκληση του χρήστη A στον χρήστη B. Στην συνέχεια ο χρήστης B αποδέχεται την πρόσκληση και η επικοινωνία εγκαθίσταται όπως στο παραπάνω παράδειγμα. Το σχεδιάγραμμα μιας τέτοιου τύπου συνδιάλεξης φαίνεται στο παρακάτω σχήμα.



## **1. Εισαγωγή στο SIP**

Υπάρχουν πολλές εφαρμογές στο διαδίκτυο που απαιτούν την δημιουργία και την διαχείριση μιας συνδιάλεξης, όπου συνδιάλεξη θεωρείται η ανταλλαγή δεδομένων μεταξύ μιας ομάδας ατόμων. Ωστόσο η εκτέλεση αυτών των εφαρμογών περιπλέκεται απ' τις πρακτικές των χρηστών τους : οι χρήστες μπορεί να μετακινούνται μεταξύ των σημείων τερματισμού, μπορεί να είναι διευθετήσιμοι με διάφορα ονόματα και μπορεί να επικοινωνούν με διαφορετικά μέσα επικοινωνίας- πολλές φορές ταυτόχρονα. Διάφορα πρωτόκολλα έχουν δημιουργηθεί ώστε να μεταφέρουν δεδομένα όπως ήχος, εικόνα και κείμενο. Το SIP συνεργάζεται με αυτά τα πρωτόκολλα με το να επιτρέπει σε διαδικτυακά σημεία τερματισμού να εντοπίζουν το ένα το άλλο και να συμφωνούν σε ένα τύπο συνδιάλεξης που θα μπορούν να χρησιμοποιήσουν. Για τον εντοπισμό διαθέσιμων χρηστών καθώς και για άλλες λειτουργίες το SIP κάνει χρήση μιας υποδομής δικτυακών διακομιστών (Proxy Servers) στους οποίους οι χρήστες μπορούν να στέλνουν καταχωρήσεις, προσκλήσεις για συνδιαλέξεις και άλλες αιτήσεις. Το SIP είναι ένα ευέλικτο γενικής χρήσης εργαλείο για την δημιουργία, μετατροπή και τερματισμό συνδιαλέξεων το οποίο λειτουργεί ανεξάρτητα από άλλα πρωτόκολλα και χωρίς να εξαρτάται από τον τύπο της συνδιάλεξης που εγκαθιστάται.

## **2.Περίληψη SIP λειτουργικότητας**

Το SIP (Session Initiation Protocol) πρόκειται για ένα πρωτόκολλο σηματοδότησης για έναρξη, τροποποίηση και τερματισμό multimedia συνδιαλέξεων όπως διαδικτυακές τηλεφωνικές συνδιαλέξεις.. Το SIP μπορεί επίσης να προσκαλέσει χρήστες σε ήδη ενεργές συνδιαλέξεις. Το SIP επίσης υποστηρίζει λειτουργίες ονομαστικής απεικόνισης και αποστολής σε νέα διεύθυνση οι οποίες επιτρέπουν την κινητικότητα του χρήστη – οι χρήστες μπορούν να αναγνωρίζονται ανεξάρτητα της τοποθεσίας δικτύου που βρίσκονται.

Το SIP χρησιμοποιεί πέντε θεμελιώδεις αρχές για την εγκατάσταση και τερματισμό επικοινωνιών με πολυμέσα.

Εντοπισμός Χρήστη : Καθορισμός του τερματικού συστήματος που θα χρησιμοποιηθεί για την επικοινωνία.

Διαθεσιμότητα Χρήστη : Καθορισμός της επιθυμίας των προσκεκλημένων μελών να συμμετέχουν σε επικοινωνία.

Δυνατότητες Χρήστη : Καθορισμός των μέσων και των παραμέτρων αυτών που θα χρησιμοποιηθούν.

Ρυθμίσεις συνδιάλεξης : ‘κουδουνισμός’, εγκατάσταση των παραμέτρων συνδιάλεξης σε όλα τα μέλη της συνδιάλεξης.

Διαχείριση συνδιάλεξης : Περιλαμβάνει μεταφορά και τερματισμό συνδιαλέξεων και αλλαγές στις παραμέτρους της συνδιάλεξης.

Το SIP δεν είναι καθαρά ένα ολοκληρωμένο σύστημα επικοινωνιών, αλλά ένα εργαλείο το οποίο μπορεί να χρησιμοποιηθεί με άλλα IETF πρωτόκολλα για να δημιουργήσει μια ολοκληρωμένη αρχιτεκτονική πολυμέσων. Αυτή η αρχιτεκτονική εμπεριέχει πρωτόκολλα όπως το RTP (Real Time Protocol) για μεταφορά δεδομένων σε πραγματικό χρόνο, RTSP (Real Time Streaming Protocol) για έλεγχο παραλαβής μέσων ροής, MEGACO (Media Gateway Control Protocol) για τον έλεγχο πυλών για το PSTN δίκτυο, SDP (Session Description Protocol) για την περιγραφή συνδιαλέξεων με πολυμέσα. Άρα το SIP μπορεί να προσφέρει ολοκληρωμένες υπηρεσίες στους χρήστες σε συνεργασία με τα παραπάνω πρωτόκολλα. Ωστόσο η βασική λειτουργία και χρήση του SIP δεν εξαρτάται από κανένα από τα παραπάνω πρωτόκολλα.

Ουσιαστικά το SIP δεν παρέχει υπηρεσίες. Αλλά παρέχει τεχνικές που μπορούν να εφαρμοστούν για να πραγματοποιηθούν διάφορες υπηρεσίες. Για παράδειγμα το SIP μπορεί να εντοπίσει κάποιον χρήστη και να του στείλει πολύπλοκο πακέτο πληροφοριών. Αν η ίδια τεχνική χρησιμοποιηθεί για την αποστολή των χαρακτηριστικών μιας συνδιάλεξης βασισμένα σε SDP τότε τα σημεία τερματισμού (endpoints) μπορούν να συμφωνήσουν για τις παραμέτρους αυτής της συνδιάλεξης. Αν η ίδια τεχνική χρησιμοποιηθεί για την αποστολή της φωτογραφίας του ατόμου που πραγματοποιεί μια κλήση καθώς και τα χαρακτηριστικά της συνδιάλεξης που επιθυμεί να πραγματοποιήσει, τότε μια ρουτίνα “caller ID” μπορεί να πραγματοποιηθεί. Όπως φαίνεται μία απλή τεχνική χρησιμοποιείται για την παροχή ένα σωρό υπηρεσιών.

Το SIP δεν παρέχει υπηρεσίες ελέγχου συνδιάλεξης, ούτε προσδιορίζει τον τρόπο διαχείρισης της συνδιάλεξης. Ωστόσο το SIP μπορεί να χρησιμοποιηθεί για την έναρξη συνδιάλεξης η οποία χρησιμοποιεί ήδη κάποιο πρωτόκολλο ελέγχου.

Η φύση των υπηρεσιών που παρέχονται καθιστούν το θέμα της ασφάλειας ιδιαίτερα σημαντικό. Γι' αυτό το λόγο το SIP παρέχει ένα ολοκληρωμένο πακέτο υπηρεσιών ασφαλείας το οποίο περιέχει δυνατότητες , άρνησης υπηρεσιών, αναγνώρισης και εξακρίβωσης στοιχείων (μεταξύ χρηστών και μεταξύ χρήστη και proxy server), ακεραιότητα προστασίας και κωδικοποίηση καθώς και εμπιστευτικές λειτουργίες.

Το SIP είναι συμβατό με τα IPv4 και IPv6.

### 3.Ορολογία

Σε αυτό τι κείμενο οι λέξεις κλειδί «πρέπει» , «δεν πρέπει», «ΑΠΑΙΤΕΙΤΑΙ», «ΘΑ», «ΔΕΝ ΘΑ», «ΘΑ ΕΠΡΕΠΕ», «ΔΕΝ ΘΑ ΕΠΡΕΠΕ», «ΠΡΟΤΕΙΝΕΤΑΙ», «ΔΕΝ ΠΡΟΤΕΙΝΕΤΑΙ», «ΙΣΩΣ» και «ΠΡΟΑΙΡΕΤΙΚΑ» αποτελούν εντολές κώδικα του SIP και πρόκειται να χρησιμοποιηθούν στα παρακάτω παραδείγματα.

### 4.Περίληψη λειτουργίας

Αυτό το τμήμα εισάγει τις βασικές λειτουργίες του SIP : εντοπισμός τερματικού σημείου, σήμα επιθυμίας επικοινωνίας, διαπραγμάτευση παραμέτρων συνδιάλεξης και τερματισμός συνδιάλεξης μετά το πέρας επικοινωνίας.



Σχήμα 1



Το σχήμα 1 δείχνει ένα τυπικό παράδειγμα ανταλλαγής μηνυμάτων μεταξύ δύο χρηστών, της Μαρίας και του Γιώργου.( κάθε μήνυμα έχει τίτλο F και έναν αριθμό). Σε αυτό το παράδειγμα η Μαρία χρησιμοποιεί ένα SIP πρόγραμμα από τον υπολογιστή της (softphone) για να καλέσει τον Γιώργο στο SIP τηλέφωνό του μέσω του διαδικτύου. Στο σχήμα φαίνονται και οι δύο SIP Proxy Servers για το δίκτυο της Μαρίας και του Γιώργου αντίστοιχα που υλοποιούν την μεταξύ τους σύνδεση. Αυτού του τύπου η σύνδεση μερικές φορές λέγεται «SIP τραπέζιο» λόγω του σχήματος που δημιουργείται από τις κουκιδωτές γραμμές του σχήματος.

Η Μαρία «καλεί» τον Γιώργο κάνοντας χρήση της SIP του ταυτότητας, ένας τύπος ταυτότητας παρόμοιος με αυτόν των email διευθύνσεων, που ονομάζεται SIP URI. Αυτές οι ταυτότητες εμπεριέχουν το όνομα χρήστη και παροχέα. Για τον Γιώργο για παράδειγμα η ταυτότητά του είναι sip:Γιώργος@altecnet.com όπου altecnet.com είναι το πεδίο του παροχέα υπηρεσιών στον οποίο ανήκει ο Γιώργος. Το ίδιο ισχύει για την Μαρία της οποίας η δικιά της ταυτότητα είναι sip:Μαρία@forthnet.com. Το SIP επίσης παρέχει και μία ασφαλή ταυτότητα (URI) η οποία ονομάζεται SIPS URI. Ένα παράδειγμα μιας τέτοιας ταυτότητας για την Μαρία θα ήταν sips:Μαρία@forthnet.com. Μία κλήση σε μια SIPS URI θα εγγυόταν μια ασφαλή και κωδικοποιημένη μεταφορά μηνυμάτων από τον καλούντα προς τον κλησθέντα χρήστη. Ωστόσο οι μηχανισμοί ασφαλείας σε αυτήν την περίπτωση εξαρτώνται από τα χαρακτηριστικά του πεδίου του κλησθέντα.

Το SIP στηρίζεται πάνω σε ένα μοντέλο συναλλαγής ερώτησης/απάντησης παρόμοιο με το HTTP. Σε αυτό το παράδειγμα η συναλλαγή ξεκινά με την αίτηση «ΠΡΟΣΚΛΗΣΗ» της Μαρίας προς την SIP URI διεύθυνση του Γιώργου. Η αίτηση «ΠΡΟΣΚΛΗΣΗ» είναι ένα παράδειγμα SIP μεθόδου που υποδεικνύει το τι πράξη επιθυμεί ο καλώντας να πράξει ο κλησθέντας, (στην συγκεκριμένη περίπτωση να δεχτεί την κλήση). Η SIP αίτηση περιέχει ένα πλήθος κύριων πεδίων τα οποία περιέχουν επιπλέον πληροφορίες για το μήνυμα. Τα κύρια πεδία για την «ΠΡΟΣΚΛΗΣΗ» του παραδείγματος περιέχουν την διεύθυνση του Γιώργου (SIP URI), της Μαρίας και πληροφορίες για τον τύπο συνδιάλεξης που επιθυμεί η Μαρία να εγκαταστήσουν. Το μήνυμα «ΠΡΟΣΚΛΗΣΗ» έχει την παρακάτω δομή.

```
INVITE sip:Γιώργος@altecnet.com SIP/2.0
Via: SIP/2.0/UDP pc33.forthnet.com;branch=z9hG4bK776asdhds
Max-Forwards: 70
```

To Γιώργος <sip:Γιώργος@altecnet.com>  
From Μαρία <sip:Μαρία@forthnet.com>;tag=1928301774  
Call-IDa84b4c76e66710@pc33.forthnet.com  
CSeq: 314159 INVITE  
Contact: <sip:Μαρία@pc33.forthnet.com>  
Content-Type: application/sdp  
Content-Length: 142

Η πρώτη γραμμή αυτού του κωδικοποιημένου κειμένου περιέχει το όνομα της μεθόδου «ΠΡΟΣΚΛΗΣΗ». Οι γραμμές που ακολουθούν είναι μια λίστα των κύριων πεδίων.

Η γραμμή Via περιέχει την διεύθυνση (pc33.forthnet.com) στην οποία η Μαρία περιμένει να δεχτεί απαντήσεις στο αίτημά της. Επίσης περιέχει μια παράμετρο η οποία χρησιμοποιείται για την αναγνώριση αυτής της συναλλαγής.

Η γραμμή To περιέχει το όνομα απεικόνισης του χρήστη (Γιώργος) και την SIP URI (sip:Γιώργος@altecnet.com) ή SIPS URI (sips:Γιώργος@altecnet.com) στην οποία η αίτηση κατευθύνεται.

Η γραμμή From περιέχει το εικονικό όνομα της Μαρίας και την SIP URI (sip:Μαρία@forthnet.com) ή SIPS URI (sips:Μαρία@forthnet.com) ταυτότητα που δείχνει τον αποστολέα της αίτησης. Το πεδίο επίσης έχει μια παράμετρο με έναν τυχαίο αριθμό που προστέθηκε στο SIP της Μαρίας για λόγους αναγνώρισης.

Η γραμμή Call-ID περιέχει ένα μοναδικό για το υπόλοιπο δίκτυο στοιχείο αναγνωρισιμότητας για αυτήν τη κλήση και παράγεται από τον συνδυασμό μιας τυχαίας ακολουθίας και την ταυτότητα του καλλούντα (στην συγκεκριμένη περίπτωση της Μαρίας).

Η γραμμή CSeq (Command Sequence) περιέχει έναν ακέραιο και το όνομα της μεθόδου που ακολουθείται. Αυτός ο ακέραιος αυξάνεται για κάθε νέα αίτηση που πραγματοποιείται σε μια συνομιλία.

Η γραμμή Contact περιέχει μια SIP URI η οποία αντιπροσωπεύει μια απευθείας δίοδος επικοινωνίας με την Μαρία, η οποία συνήθως αποτελείται από ένα όνομα χρήστη καταχωρημένο στο πεδίο (FQDN). Αν και τα FQDN προτείνονται πολλά τερματικά δεν έχουν καταχωρήσει όνομα χρήστη και γι' αυτό επιτρέπονται και IP διευθύνσεις. Δηλαδή ενώ το πεδίο Via οδηγεί τα άλλα στοιχεία προς τα πού να στείλουν την

απάντηση, το πεδίο Contact λέει στα άλλα στοιχεία που να στέλνουν μελλοντικές αιτήσεις.

Το πεδίο Max-Forward περιορίζει τον αριθμό των αλμάτων μιας αίτησης μέχρι αυτή να φτάσει στον προορισμό της. Αποτελείται από έναν ακέραιο που μειώνεται σε κάθε άλμα.

Το πεδίο Content Type περιέχει την περιγραφή των περιεχομένων του μηνύματος

Το πεδίο Content-length περιέχει τον αριθμό σε byte του μήλους του μηνύματος.

Οι λεπτομέρειες της συνδιάλεξης όπως ο τύπος των μέσων που χρησιμοποιούνται, οι κωδικοποιητές και ο ρυθμός δειγματοληψίας δεν περιγράφονται με την χρήση του SIP. Ωστόσο το σώμα του SIP μηνύματος περιέχει μια περιγραφή της συνδιάλεξης, κωδικοποιημένη σε ενός διαφορετικού πρωτοκόλλου τύπο. Ένας τέτοιος τύπος είναι το Session Description Protocol (SDP). Αυτό το SDP μήνυμα μεταφέρεται από το SIP μήνυμα με τρόπο ανάλογο με αυτόν ενός κειμένου που μεταφέρεται από ένα email μήνυμα, ή μιας ιστοσελίδας η οποία μεταφέρεται από ένα HTTP μήνυμα.

Μιας και το softphone δεν γνωρίζει την θέση του Γιώργου ούτε του SIP Server στο altecnet.com τομέα, το softphone στέλνει πρόσκληση (INVITE) στον server στο πεδίο της Μαρίας, forthnet.com. Η διεύθυνση του SIP Server του forthnet.com μπορεί να είχε ρυθμιστεί από το softphone της Μαρίας, ή μπορεί να είχε ανακαλυφθεί από το DHCP για παράδειγμα.

Ο SIP Server του forthnet.com είναι ένας Proxy Server. Οι Proxy Servers δέχονται τις SIP αιτήσεις και τις προωθούν εκ μέρους του αιτών. Σε αυτό το παράδειγμα ο Proxy Server δέχεται την αίτηση ΠΡΟΣΚΛΗΣΗ και στέλνει 100(Προσπαθώ) απάντηση στο softphone της Μαρίας. Η απάντηση 100 σημαίνει πως η «ΠΡΟΣΚΛΗΣΗ» έχει φθάσει και ότι ο Proxy Server προσπαθεί να αποστείλει την «ΠΡΟΣΚΛΗΣΗ» στον προορισμό της. Οι απαντήσεις στον SIP κώδικα αποτελούνται από τριψήφιους κωδικούς ακολουθούμενοι από μια περιγραφική φράση. Αυτή η απάντηση περιέχει την ίδια παράμετρο που υπάρχει στο πεδίο Via (και στο INVITE) και έτσι το softphone της Μαρίας μπορεί να κάνει την αντιστοίχιση με την αίτηση «ΠΡΟΣΚΛΗΣΗ». Ο Proxy Server του forthnet.com εντοπίζει τον Proxy Server του altecnet.com πιθανώς χρησιμοποιώντας ένα DNS (Domain Name Service) ψάξιμο για να βρει

ποιος Server εξυπηρετεί τον altecnet.com τομέα. Αυτό έχει σαν αποτέλεσμα να βρίσκει το IP του Proxy Server του altecnet.com και προωθεί την αίτηση «ΠΡΟΣΚΛΗΣΗ» σε αυτόν. Προτού προωθήσει την αίτηση ο Proxy Server του forthnet.com βάζει επιπλέον ένα Via πεδίο που περιέχει την δική του διεύθυνση (η αίτηση «ΠΡΟΣΚΛΗΣΗ» ήδη περιέχει την διεύθυνση της Μαρίας στο πεδίο Via). Ο Proxy Server του altecnet.com δέχεται την αίτηση «ΠΡΟΣΚΛΗΣΗ» και απαντά με ένα 100 (προσπαθώ) πίσω στον Proxy Server του forthnet.com για δείξει ότι έχει λάβει την αίτηση και την επεξεργάζεται. Ο Proxy Server συμβουλευεται μια βάση δεδομένων, η οποία περιέχει την IP διεύθυνση του Γιώργου. Ο Proxy Server του altecnet.com προσθέτει άλλο ένα Via πεδίο με την δική του διεύθυνση στην «ΠΡΟΣΚΛΗΣΗ» και την προωθεί στο SIP τηλέφωνο του Γιώργου.

Το SIP τηλέφωνο του Γιώργου δέχεται την «ΠΡΟΣΚΛΗΣΗ» και ειδοποιεί τον Γιώργο για την εισερχόμενη κλήση από την Μαρία με κουδουνισμό. Το SIP τηλέφωνο του Γιώργου στέλνει μια απάντηση 180 (κουδουνισμός) μέσω των δύο Proxy Servers στην αντίθετη κατεύθυνση. Κάθε Proxy Server χρησιμοποιεί το πεδίο Via για να καθορίσει που θα στείλει την απάντηση και αφαιρεί την δικιά του διεύθυνση από πάνω. Έτσι η απάντηση 180(κουδουνισμός) γυρνάει πίσω στον καλούντα χωρίς να χρειάζεται ψάξιμο χάρις στα Via πεδία. Επίσης έτσι κάθε Proxy Server που βλέπει την «ΠΡΟΣΚΛΗΣΗ» βλέπει και τις απαντήσεις σε αυτήν.

Όταν το softphone της Μαρίας δεχτεί την 180 (κουδουνισμός) απάντηση δίνει αυτήν την πληροφορία στην Μαρία μέσω ενός ακουστικού σήματος ή εμφανίζοντας ένα μήνυμα στην οθόνη της Μαρίας.

Ας υποθέσουμε ότι ο Γιώργος δέχεται να απαντήσει την κλήση. Όταν σηκώσει το ακουστικό το SIP τηλέφωνο στέλνει μία 200(OK) απάντηση για να υποδείξει ότι η κλήση έχει αποδεχτεί. Η απάντηση 200(OK) περιέχει ένα σώμα μηνύματος με την SDP περιγραφή της συνδιάλεξης που ο Γιώργος επιθυμεί να εγκαταστήσει με την Μαρία. Αυτό έχει ως αποτέλεσμα να υπάρχει μια ανταλλαγή SDP μηνυμάτων σε δύο φάσεις : Η Μαρία στέλνει ένα μήνυμα στον Γιώργο και ο Γιώργος ένα στην Μαρία. Αυτή η διπλή ανταλλαγή προσφέρει βασικές δυνατότητες διαπραγμάτευσης και βασίζεται σε ένα απλό μοντέλο προσφοράς/απάντησης SDP συναλλαγής. Αν ο Γιώργος δεν επιθυμούσε να απαντήσει ή το τηλέφωνο ήταν απασχολημένο μια απάντηση λάθους θα είχε σταλεί στην θέση της 200(OK), το οποίο θα σήμαινε ότι η σύνδεση δεν θα ήταν πραγματοποιήσιμη. Η απάντηση 200(OK) του Γιώργου μοιάζει κάπως έτσι:

SIP/2.0 200 OK

Via: SIP/2.0/UDP server10. altecnet.com  
;branch=z9hG4bKnashds8;received=192.0.2.3  
Via: SIP/2.0/UDP bigbox3.site3.forthnet.com  
;branch=z9hG4bK77ef4c2312983;received=192.0.2.2  
Via: SIP/2.0/UDP pc33.forthnet.com  
;branch=z9hG4bK776asdhds;received=192.0.2.1  
To: Γιώργος<sip:Γιώργος@altecnet.com>;tag=a6c85cf  
From: Μαρία< sip:Μαρία@forthnet.com>;tag=1928301774  
Call-ID: a84b4c76e66710@pc33.forthnet.com  
CSeq: 314159 INVITE  
Contact: <sip:Γιώργος@192.0.2.4>  
Content-Type: application/sdp  
Content-Length: 131

Η πρώτη γραμμή της απάντησης περιέχει τον κωδικό απάντησης (200) και την περιγραφική φράση (OK). Οι υπόλοιπες γραμμές αποτελούν τα κύρια πεδία. Τα πεδία Via, To, From, Call-ID και CSeq είναι αντιγραμμένα από την αίτηση INVITE (ΠΡΟΣΚΛΗΣΗ). (υπάρχουν τιμές για τρία πεδία Via: ένα από το softphone της Μαρίας, ένα από τον Proxy Server του forthnet.com και ένα από τον Proxy Server του altecnet.com). Επίσης το SIP τηλέφωνο του Γιώργου έχει προσθέσει μια παράμετρο στο πεδίο To. Η παράμετρος αυτή συγχωνεύεται από τα τερματικά σημεία στον διάλογο και θα συμπεριλαμβάνεται σε όλες τις μελλοντικές αιτήσεις και απαντήσεις σε αυτήν την κλήση. Το πεδίο Contact περιέχει μια διεύθυνση στην οποία ο Γιώργος μπορεί να είναι απ' ευθείας προσβάσιμος στο SIP τηλέφωνό του. Τα πεδία Content-Type και Content-Length περιέχουν στοιχεία του SDP μηνύματος του Γιώργου.

Εκτός από το DNS ψάξιμο, οι Proxy Servers μπορούν να παίρνουν «ευέλικτες» αποφάσεις αποστολής αιτήσεων και απαντήσεων. Για παράδειγμα αν το τηλέφωνο του Γιώργου έστειλε απάντηση 486(απασχολημένο), τότε ο proxy Server του altecnet.com μπορούσε να στείλει την «ΠΡΟΣΚΛΗΣΗ» στον Server που διαχειρίζεται το φωνοκιβώτιο του Γιώργου. Ένας Proxy Server μπορεί να στείλει μια «ΠΡΟΣΚΛΗΣΗ» σε πολλές διευθύνσεις ταυτόχρονα. Αυτός ο τύπος παράλληλης ανίχνευσης ονομάζεται Forking.

Στην συγκεκριμένη περίπτωση η απάντηση 200(OK) επιστρέφει πίσω μέσω των δύο Proxy Server και λαμβάνεται από το softphone της Μαρίας το οποίο σταματά το ακουστικό σήμα και υποδεικνύει ότι η κλήση έχει αποδεχθεί. Τέλος το softphone της Μαρίας στέλνει ένα μήνυμα ότι το έχει λάβει γνώση (ACK), στο τηλέφωνο του Γιώργου για να επιβεβαιώσει την λήψη της απάντησης 200(OK). Σε αυτό το παράδειγμα το ACK στέλνεται κατευθείαν από το softphone της Μαρίας στο SIP

τηλέφωνο του Γιώργου αγνοώντας τους δύο Proxy Servers. Αυτό συμβαίνει διότι τα τερματικά σημεία γνωρίζουν ήδη την διεύθυνσεις του ενός και του άλλου από τα πεδία Contact μέσω της <INVITE>/200(OK) ανταλλαγής η οποία δεν ήταν γνωστή μέχρι την αποστολή της αρχικής «ΠΡΟΣΚΛΗΣΗΣ». Έτσι ουσιαστικά δεν απαιτείται ψάξιμο από τους δύο Proxy Servers και έτσι μένουν εκτός από την ροή δεδομένων μεταξύ των δύο τερματικών. Με αυτήν την πράξη ολοκληρώνεται η τριπλή ανταλλαγή με τα INVITE/200(OK)/ACK με την οποία τελικά εγκαθίστανται οι SIP συνδιαλέξεις.

Η συνδιάλεξη μεταξύ της Μαρίας και του Γιώργου έχει αρχίσει και στέλνουν δεδομένα ο ένας στον άλλο σε μια μορφή που συμφωνήθηκε μέσω της ανταλλαγής των SDP. Γενικά τα πακέτα δεδομένων ακολουθούν διαφορετικό μονοπάτι από ότι τα SIP μηνύματα σηματοδοσίας.

Κατά την διάρκεια της συνδιάλεξης η Μαρία και ο Γιώργος μπορούν να αλλάξουν τα χαρακτηριστικά αυτής της συνδιάλεξης. Αυτό επιτυγχάνεται με το να στέλνουν μια αίτηση re-INVITE η οποία περιέχει τα νέα στοιχεία της συνδιάλεξης. Αυτή η re-INVITE πρόσκληση περιέχει στοιχεία τα οποία αναφέρονται στην υπάρχουσα συνδιάλεξη και έτσι το άλλο μέλος γνωρίζει ότι πρόκειται για αίτηση μεταβολής στοιχείων αυτής της συνδιάλεξης και όχι αίτηση για εγκατάσταση νέας. Το άλλο μέλος στέλνει το μήνυμα 200(OK) για να δεχτεί την αλλαγή. Ο αιτών απαντά στο 200(OK) με ένα ACK. Αν το άλλο μέλος δεν δεχτεί την αλλαγή στέλνει μήνυμα λάθους 488(MH ΑΠΟΔΕΧΤΟ) και επίσης δέχεται ένα ACK. Σε αυτήν την περίπτωση η συνδιάλεξη συνεχίζεται με τα υπάρχουσα χαρακτηριστικά.

Στο τέλος της κλήσης ο Γιώργος αποσυνδέεται (κατεβάζει το ακουστικό) πρώτος και παράγει ένα μήνυμα ANTI-O. Αυτό το ANTI-O στέλνεται απευθείας στο softphone της Μαρίας. Η Μαρία απαντά στο μήνυμα με ένα 200(OK) και έτσι τερματίζεται η συνδιάλεξη και η συναλλαγή ANTI-O. Δεν απαιτείται ACK αποστολή καθώς το ACK χρησιμοποιείται για να απαντήσει σε απαντήσεις της αίτησης INVITE. Οι λόγοι που συμβαίνει αυτό για την αίτηση INVITE θα αναλυθούν αργότερα αλλά έχει να κάνει με διάφορους μηχανισμούς του SIP. Και για αυτόν τον λόγο οι αιτήσεις χωρίζονται σε τύπου INVITE και μη-INVITE αιτήσεις.

Σε μερικές περιπτώσεις θα ήταν χρήσιμο για τους Proxy Servers να δούν όλα τα μηνύματα σηματοδοσίας. Για παράδειγμα αν ο Proxy Server του altecnet.com επιθυμούσε να παραμείνει στο μονοπάτι σηματοδοσίας πέρα από το αρχικό INVITE, θα πρόσθετε στην αίτηση INVITE ένα

πεδίο καθοδήγησης το οποίο θα περιείχε μια URI ή την IP διεύθυνση του Proxy Server. Αυτή η πληροφορία θα πήγαινε και στο SIP τηλέφωνο του Γιώργου αλλά και στο softphone της Μαρίας κατά την διάρκεια του διαλόγου. Τότε ο Proxy Server του altecnet.com θα έβλεπε τα μηνύματα ACK, ANTIΟ, και 200(OK) και θα τα καθοδηγούσε. Κάθε Proxy Server μπορεί ανεξάρτητα να επιλέξει το αν θα βλέπει τέτοιου τύπου μηνύματα και το αν αυτά τα μηνύματα θα προωθηθούν από αυτόν. Αυτή η δυνατότητα παρέχεται συχνά σε Proxy Servers που παρέχουν επιπλέον υπηρεσίες κατά την διάρκεια της κλήσης.

Η καταγραφή (Registration) είναι άλλη μια απλή λειτουργία του SIP. Η καταγραφή είναι ένας τρόπος για τον Server του altecnet.com να εντοπίσει την τωρινή θέση του Γιώργου. Κατά την ενεργοποίηση και κατά τακτά χρονικά διαστήματα το SIP τηλέφωνο του Γιώργου στέλνει μηνύματα ΚΑΤΑΧΩΡΗΣΗΣ (REGISTER) σε έναν Server του altecnet.com γνωστός και ως SIP ΚΑΤΑΓΡΑΦΕΑΣ (REGISTRAR). Τα μηνύματα καταγραφής συσχετίζουν το SIP ή SIPS URI του Γιώργου (sip:Γιώργος@altecnet.com) με το μηχάνημα με το οποίο έχει συνδεθεί. Ο ΚΑΤΑΓΡΑΦΕΑΣ αποθηκεύει αυτήν την συσχέτιση σε μια βάση δεδομένων η οποία μπορεί να χρησιμοποιηθεί από τον τομέα του altecnet.com. Συχνά ο ΚΑΤΑΓΡΑΦΕΑΣ ενός τομέα βρίσκεται στην ίδια θέση με τον Proxy Server του τομέα αυτού.

Ο Γιώργος δεν περιορίζεται στο να στέλνει σήματα καταγραφής από μία μόνο συσκευή. Μπορεί το SIP τηλέφωνο στο σπίτι του αλλά και αυτό στο γραφείο του να στέλνουν τέτοια σήματα. Αυτό επιτρέπει στον Proxy Server να εκτελεί διαφόρων τύπων αναζητήσεις για να εντοπίσει τον Γιώργο. Αυτό ισχύει και αντίστροφα. Μπορεί πολλοί χρήστες να χρησιμοποιούν μόνο μία συσκευή για να στέλνουν σήματα καταγραφής και μάλιστα την ίδια στιγμή.

Η υπηρεσία εντοπισμού περιέχει πληροφορίες που επιτρέπουν στον Proxy Server να εισάγει μια URI και να δεχτεί ένα πλήθος URI που του λένε που να στείλει μια αίτηση. Οι καταγραφές είναι ένας τρόπος για να παραχθούν αυτές οι πληροφορίες αλλά όχι ο μόνος τρόπος.

Τέλος είναι σημαντικό να σημειώσουμε πως ότι στο SIP η καταγραφή χρησιμοποιείται για την καθοδήγηση εισερχόμενων SIP αιτημάτων και δεν έχει κανένα ρόλο στην αποστολή εξερχόμενων αιτημάτων.

## **5.Δομή του Πρωτόκολλου**

Το SIP είναι δομημένο ως ένα επιφανειακό πρωτόκολλο, το οποίο σημαίνει πως η συμπεριφορά του περιγράφεται από ένα σετ ανεξαρτήτων τμημάτων επεξεργασίας τα οποία στάδια έχουν ελάχιστη σύνδεση το ένα με το άλλο. Η συμπεριφορά του πρωτόκολλου περιγράφεται ως στρώματα για τον λόγο της παρουσίας, επιτρέποντας την περιγραφή των λειτουργιών σε όλα τα στοιχεία ενός τομέα. Ωστόσο δεν επιβάλλουν τον τρόπο εφαρμογής τους με κανένα τρόπο. Όταν λέμε ότι ένα στοιχείο περιέχει ένα στρώμα, εννοούμε πως είναι συμβατό με το σέτ των κανόνων που ορίζονται από αυτό το στρώμα.

Βέβαια αυτό δεν σημαίνει πως όλα τα στοιχεία περιέχουν όλα τα στρώματα. Τα στοιχεία που διέπουν τον SIP κώδικα πρόκειται για λογικά στοιχεία και όχι φυσικά.

Το χαμηλότερο στρώμα SIP κώδικα είναι η σύνταξη και η κωδικοποίηση. Η κωδικοποίηση ορίζεται από ένα διάγραμμα μορφής Backus-Naur(BNF).

Το δεύτερο στρώμα είναι το στρώμα μεταφοράς. Ορίζει το πώς ένας χρήστης στέλνει αιτήσεις και δέχεται απαντήσεις και το πώς ένας Server κάνει το ίδιο στο δίκτυο. Όλα τα SIP στοιχεία περιέχουν στρώμα μεταφοράς.

Το τρίτο στρώμα είναι το στρώμα συναλλαγής. Οι συναλλαγές αποτελούν ένα κύριο συστατικό του SIP. Η συναλλαγή είναι μία αίτηση που στέλνεται από μία συναλλαγή χρήστη (κάνοντας χρήση του στρώματος συναλλαγής) σε μια συναλλαγή Server, μαζί με όλες τις απαντήσεις από την συναλλαγή Server στον χρήστη. Το στρώμα συναλλαγής διαχειρίζεται όλες τις αναμεταδόσεις εφαρμογών στρώματος, αντιστοιχώντας απαντήσεις σε αιτήματα. Κάθε βήμα που ολοκληρώνει ο φορέας χρήστη (UAC) γίνεται μέσω ενός συνόλου συναλλαγών. Οι φορείς χρήστη περιέχουν στρώμα συναλλαγής, όπως και οι Proxy καταστάσεως. Το στρώμα συναλλαγής περιέχει ένα συστατικό φορέα (το οποίο ονομάζεται συναλλαγή φορέα) και ένα στοιχείο Server (το οποίο ονομάζεται συναλλαγή Server) το καθένα από τα οποία αντιπροσωπεύεται από ένα καθορισμένης καταστάσεως μηχανήματα το οποί κατασκευάζεται ώστε να επεξεργάζεται μια συγκεκριμένη αίτηση.

Το στρώμα πάνω από το στρώμα συναλλαγής ονομάζεται Χρήστης Συναλλαγής (Transaction User or TU). Κάθε SIP οντότητα εκτός από τους Server δίχως κατάσταση είναι ένας TU. Όταν ένας TU επιθυμεί να στείλει μια αίτηση δημιουργεί μια συναλλαγή φορέα αίτηση την οποία προωθεί μαζί με την IP διεύθυνση του προορισμού, την θύρα (port) και το μέσω με το οποίο θα σταλεί το αίτημα. Ένα TU το οποίο δημιουργεί



μια συναλλαγή φορέα μπορεί επίσης να την καταργήσει. Όταν ένας φορέας καταργεί μια συναλλαγή, ζητά από τον Server να σταματήσει κάθε επεξεργασία και να επιστρέψει στην κατάσταση που βρισκόταν πριν την έναρξη της συναλλαγής, παράγοντας ένα μήνυμα σφάλματος για αυτήν στην συναλλαγή. Αυτό γίνεται με ένα αίτημα CANCEL (ΑΚΥΡΩΣΗ) το οποίο εμπεριέχεται στην συναλλαγή αλλά αναφέρεται στην ακύρωση αυτής.

Τα SIP στοιχεία όπως, οι φορείς χρήστη και Server, οι Proxy με κατάσταση ή χωρίς και οι καταγραφείς, περιέχουν ένα πυρήνα ο οποίος διαχωρίζει το ένα στοιχείο από το άλλο. Οι πυρήνες, εκτός για τους Proxy χωρίς κατάσταση, είναι χρήστες συναλλαγής. Ενώ η συμπεριφορά των UAC και UAS πυρήνων εξαρτάται από την μέθοδο, υπάρχουν μερικοί κοινοί κανόνες για όλες τις μεθόδους. Για έναν UAC αυτοί οι κανόνες έχουν να κάνουν με την κατασκευή ενός αιτήματος, ενώ για τους UAS με την επεξεργασία ενός αιτήματος και την παραγωγή μιας απάντησης. Μιας και οι καταγραφείς παίζουν σημαντικό ρόλο στο SIP, ένας UAS που διαχειρίζεται έναν REGISTER (ΚΑΤΑΓΡΑΦΕΑ) του δίνεται η ειδική ονομασία REGISTRAR.

Υπάρχουν και άλλου τύπου αιτήσεις οι οποίες στέλνονται σε ένα διάλογο. Ο διάλογος είναι μία SIP σχέση ισοτιμίας μεταξύ δύο φορέων χρηστών η οποία διαρκεί κάποιο χρόνο. Ο διάλογος διευκολύνει την συνεχή ροή μηνυμάτων και αιτημάτων μεταξύ των δύο φορέων. Η μέθοδος INVITE είναι ο μόνος τρόπος για την εγκατάσταση διαλόγου. Όταν ένα UAC στέλνει ένα αίτημα το οποίο είναι στο απόσπασμα ενός διαλόγου, τότε αυτό ακολουθεί τους κοινούς UAC κανόνες καθώς και τους κανόνες των αιτήσεων ενδοδιαλόγου.

Η πιο σημαντική μέθοδος στο SIP είναι η μέθοδος INVITE η οποία χρησιμοποιείται για την εγκατάσταση συνδιαλέξεων μεταξύ των συμμετεχόντων. Μια συνδιάλεξη είναι μια συγκέντρωση συμμετεχόντων με συνεχή ροή πληροφοριών μεταξύ τους, με σκοπό την επικοινωνία.

## **6.Ορισμοί**

Address Of Record (AOR): Το AOR είναι μια SIP ή SIPS URI η οποία δείχνει στην υπηρεσία εντοπισμού ενός τομέα μία URI την οποία αντιστοιχεί μια άλλη στην οποία μπορεί ο χρήστης να είναι διαθέσιμος. Γενικά η AOR θεωρείται ως η «γενική διεύθυνση» του χρήστη.

Back to Back User Agent (B2BUA): Το B2BUA είναι μια λογική οντότητα δέχεται μια αίτηση και την επεξεργάζεται σαν ένας Server

χρήστη (UAS). Για την απόφαση το πώς θα απαντήσει μια ερώτηση λειτουργεί σαν ένας φορέας χρήστη (UAC) και παράγει αιτήματα. Αντίθετα με έναν Proxy Server διατηρεί την κατάσταση διαλόγου και πρέπει να συμμετέχει σε όλα τα αιτήματα των διαλόγων που έχει εγκαταστήσει.

Call: Η κλήση (call) είναι ένας άτυπος όρος ο οποίος αναφέρεται σε ένα είδος επικοινωνίας μεταξύ ατόμων.

Call Leg: Μια άλλη ονομασία του διαλόγου,

Call Statefull: Ένας Proxy ονομάζεται stateful (με κατάσταση) όταν διατηρεί την κατάστασή του σε έναν διάλογο από το αρχικό INVITE μέχρι το τελικό BYE(ANTIO).

Client: Ο Client (Φορέας) είναι ένα στοιχείο του δικτύου το οποίο στέλνει SIP αιτήσεις και λαμβάνει SIP απαντήσεις. Οι φορείς μερικές φορές μπορούν να αλληλεπιδρούν απευθείας με χρήστες. Οι φορείς χρήστη και οι Proxy είναι φορείς.

Conference (Συνδιάλεξη): Μια επικοινωνία πολυμέσων με πολλούς συμμετέχοντες.

Core: Ο Core (πυρήνας) υποδεικνύει τις λειτουργίες μιας SIP οντότητας.

Dialog (Διάλογος): Ο διάλογος είναι μια SIP ισότιμη σχέση (επικοινωνία) μεταξύ δύο UA η οποία διαρκεί κάποιο χρονικό διάστημα. Ο διάλογος εγκαθίσταται από SIP μηνύματα, όπως μια 2XX απάντηση σε ένα αίτημα INVITE.

Downstream: Είναι η κατεύθυνση ενός μηνύματος το οποίο είναι αποτέλεσμα αίτησης που ζητά ροή δεδομένων από τον γορέα χρήστη στον φορέα server.

Final Response (Τελική Απάντηση): Είναι μια απάντηση που τερματίζει την SIP συναλλαγή, η οποία αντιτίθεται σε μια απάντηση που δεν το κάνει. Όλες οι απαντήσεις τύπου 2xx, 3xx, 4xx, 5xx, και 6xx είναι τελικές απαντήσεις.

Header(κύριο συστατικό): Το Header είναι ένα στοιχείο ενός Sip μηνύματος το οποίο περιέχει πληροφορίες για το μήνυμα. Κάθε μήνυμα είναι δομημένο από πεδία Header.

Header field (κύριο πεδίο): Αποτελούν την δομή του Header. Ένα Header field μπορεί να εμφανίζεται μόνο του ή σαν μία σειρά πεδίων. Μια σειρά header field αποτελείται από ένα όνομα και από μια σειρά μηδενικών ή περισσοτέρων Header field τιμών. Οι πολλαπλές τιμές header field διαχωρίζονται μεταξύ τους με κόμμα. Μερικά Header fields μπορεί να έχουν μόνο μία τιμή και αυτό έχει ως αποτέλεσμα να εμφανίζονται ως μονή σειρά header field.

Home Domain (Τομέας χρήστη): Είναι ο τομέας ο οποίος παρέχει υπηρεσίες στον SIP χρήστη. Τυπικά αυτός είναι ο τομέας που εμφανίζεται στο URI του χρήστη.

Caller (Καλλών): είναι ο χρήστης που ξεκινάει την συνδιάλεξη με την αίτηση INVITE. Ο Caller διατηρεί αυτόν τον ρόλο μέχρι και τον τερματισμό της συνδιάλεξης.

Callee (καλλούντας): Είναι το μέλος που δέχεται την αίτηση INVITE με σκοπό την εγκατάσταση νέας επικοινωνίας. Και αυτός διατηρεί τον ρόλο του μέχρι τον τερματισμό της συγκεκριμένης συνδιάλεξης.

Location Service (Υπηρεσία Εντοπισμού): Είναι μια υπηρεσία που χρησιμοποιείται από έναν κατευθυντικό ή proxy server για τον εντοπισμό και την ανάκτηση πληροφοριών για την πιθανή θέση ενός καλλούντα. Βασικά περιέχει μια λίστα με κλειδιά καταγραφής που συνδέονται με τις διευθύνσεις επαφής του χρήστη.

Loop (Κύκλος): Είναι μια αίτηση που στέλνεται σε έναν Proxy, προωθείται και μετά επιστρέφει στον ίδιο proxy. Όταν επιστρέφει το URI της αίτησης είναι το ίδιο με την πρώτη φορά και τα κύρια πεδία που επηρεάζουν την λειτουργία του proxy είναι επίσης τα ίδια με αποτέλεσμα ο proxy να παίρνει συνεχώς τις ίδιες αποφάσεις για την συγκεκριμένη αίτηση όπως έκανε και την πρώτη φορά. Οι Loop αιτήσεις είναι λάθη (errors) και οι διαδικασίες για την ανίχνευσή τους και διαχείριση τους περιγράφονται από το πρωτόκολλο.

Message (Μήνυμα): Είναι δεδομένα τα οποία στέλνονται μεταξύ SIP στοιχείων σαν μέρος του πρωτόκολλου. Τα SIP μηνύματα είναι είτε αιτήσεις είτε απαντήσεις.

Method (Μέθοδος): η μέθοδος είναι μια βασική λειτουργία την οποία μια αίτηση έχει ως σκοπό να πραγματοποιήσει. Ένα παράδειγμα μεθόδου είναι οι αιτήσεις INVITE και BYE.

**Outbound Proxy (Proxy εκτός ορίων):** Είναι ένας Proxy που δέχεται αιτήσεις από έναν φορέα ακόμα και αν αυτός δεν συμπεριλαμβανόταν στο URI της αίτησης.

**Parallel Search (Παράλληλο ψάξιμο):** Στο παράλληλο ψάξιμο ένας Proxy στέλνει διάφορες αιτήσεις σε πιθανές τοποθεσίες του χρήστη μέχρι αυτός να δεχτεί μια εισερχόμενη αίτηση. Ουσιαστικά από το να περιμένει να στείλει μία αίτηση και να περιμένει να δεχτεί απάντηση μέχρι να στείλει την επόμενη, τώρα στέλνει ένα πακέτο αιτήσεων σε όλους του πιθανούς προορισμούς χωρίς πρώτα να περιμένει την απάντηση.

**Provisional Response (Προσωρινή Απάντηση):** Είναι μια απάντηση που χρησιμοποιείται από τον server για να υποδείξει πρόοδος αλλά δεν τερματίζει την συνομιλία. Τέτοιου τύπου απαντήσεις είναι οι 1xx.

**Proxy Server:** Μια οντότητα που δρα και σαν server και σαν φορέας. Με σκοπό την δημιουργία αιτήσεων εκ μέρους άλλων φορέων και την αποστολή αιτήσεων και απαντήσεων στους κατάλληλους χρήστες. Οι Proxy είναι επίσης υπεύθυνοι για τον έλεγχο και των χρηστών (αν για παράδειγμα ένας χρήστης έχει το δικαίωμα να αποστείλει μια αίτηση ή να πραγματοποιήσει μια κλήση). Αν χρειαστεί Ο Proxy μπορεί να αλλάξει κάποια στοιχεία μιας αίτησης προτού την προωθήσει.

**Redirect Server (Server Νέας Κατεύθυνσης):** πρόκειται για έναν server φορέα χρήστη ο οποίος παράγει απαντήσεις τύπου 3xx, κατευθύνοντας τον φορέα να επικοινωνήσει με ένα εναλλακτικό σετ URI.

**Registrar:** Πρόκειται για έναν server που αποδέχεται αιτήσεις REGISTER και τοποθετεί τις πληροφορίες που δέχεται από αυτές τις αιτήσεις στην υπηρεσία εντοπισμού για τον τομέα στον οποίο ανήκει.

**Regular Transaction (τακτική συναλλαγή):** Η τακτική συναλλαγή είναι μια συναλλαγή με μια μέθοδο διαφορετική από τις INVITE, ACK και CANCEL.

**Request (Αίτηση):** Είναι ένα SIP μήνυμα το οποίο στέλνεται από έναν φορέα σε έναν server με σκοπό να πράξει μια συγκεκριμένη λειτουργία.

**Response (Απάντηση):** Είναι ένα SIP μήνυμα το οποίο στέλνεται από έναν server σε έναν φορέα για να υποδείξει την κατάσταση της αίτησης που στάλθηκε από τον φορέα στον server.

Ringback (τόνος καλλούντα): πρόκειται για τον ηχητικό τόνο που παράγει η συσκευή του ατόμου που έστειλε την αρχική αίτηση για να υποδείξει ότι το άλλο μέλος έχει ειδοποιηθεί.

Route Set (Σετ κατευθύνσεων): Είναι μια συλλογή από SIP ή SIPS URI που δείχνουν μια λίστα από proxy που πρέπει να διασχιστούν για την αποστολή μιας συγκεκριμένης αίτησης. Αυτό το σετ κατευθύνσεων συνήθως καθορίζεται από τα κύρια πεδία (Header Fields).

Server: Είναι ένα στοιχείο του δικτύου το οποίο δέχεται αιτήσεις με σκοπό να τις εκτελέσει και στέλνει απαντήσεις σε αυτές τις αιτήσεις.

Sequential Search (Διαδοχική Αναζήτηση): Σε αυτού του τύπου την αναζήτηση ένας proxy προσπαθεί να επικοινωνήσει με κάθε διεύθυνση διαδοχικά, πηγαίνοντας στην επόμενη διεύθυνση μόνο όταν η προηγούμενη έχει στείλει τελική απάντηση. Οι απαντήσεις τύπου 2xx και 6xx πάντα τερματίζουν τις διαδοχικές αναζητήσεις.

Session (Συνομιλία): Μία συνομιλία πολυμέσων είναι ένα σετ από αποστολές και δέκτες με συνεχή ροή δεδομένων από τους αποστολές στους δέκτες. Αυτό σημαίνει ότι ένα άτομο μπορεί να έχει διάφορες προσκλήσεις για συνομιλία από διαφορετικά άτομα θγια την ίδια συνομιλία πάντα).

SIP Transaction (SIP Συναλλαγή): Μια SIP συναλλαγή γίνεται μεταξύ ενός φορέα και ενός server και περιλαμβάνει όλα τα μηνύματα από την πρώτη αίτηση έως και την τελική απάντηση.

Spiral (Σπείρα): Είναι μια αίτηση η οποία οδηγείται σε έναν proxy, προωθείται και αργότερα επιστρέφει στον ίδιο proxy, αλλά αυτή την φορά με αλλαγμένα στοιχεία έτσι ώστε αυτήν την φορά ο proxy να παίρνει διαφορετικές αποφάσεις για την αίτηση αυτήν την φορά. Το spiral δεν είναι error όπως το loop.

Target Refresh Request: (Αίτηση ανανέωσης στόχου): Μια τέτοια αίτηση η οποία στέλνεται σε ένα διάλογο έχει ως σκοπό να αλλάξει τον στόχο (τερματικό)του διαλόγου.

Transaction User (Χρήστης συναλλαγής): Είναι το στρώμα της επεξεργασίας πρωτόκολλου που βρίσκεται πάνω από το στρώμα συναλλαγής

Upstream: Είναι η κατεύθυνση ενός μηνύματος από μια συναλλαγή η οποία αναφέρεται στην κατεύθυνση των απαντήσεων από τον φορέα server στον φορέα χρήστη.

URL-encoded (URL-κωδικοποιημένα): Είναι μια σειρά χαρακτήρων κωδικοποιημένοι σύμφωνα με το RFC-2396.

User Agent Client (Φορέας Χρήστη): Είναι μια λογική οντότητα που παράγει απαντήσεις σε SIP αιτήματα. Η απάντηση δέχεται απορρίπτει ή καθοδηγεί ξανά το αίτημα. Ο ρόλος αυτός διατηρείται μέχρι το πέρας αυτής της συναλλαγής.

UAS Core (UAS πυρήνας): Το σέτ των λειτουργιών επεξεργασίας που απαιτούνται από έναν UAS (User Agent Server) για να κάνει μια συναλλαγή και να παράγει στρώματα μεταφοράς.

User Agent: μια λογική οντότητα που μπορεί να συμπεριφέρεται και σαν φορέας χρήστη και σαν φορέας server.

Ο ρόλος των UAC και UAS, όπως και ο ρόλος των proxy και redirect server καθορίζεται από μια συνεχή βάση συναλλαγής. Για παράδειγμα ο φορέας χρήστη ο οποίος ξεκινά μια κλήση, δρά ως UAC όταν στέλνει την αίτηση INVITE και ως UAS όταν δέχεται την αίτηση BYE από το άλλο μέλος. Ομοίως το ίδιο λογισμικό μπορεί να δρα proxy server για την μία αίτηση και σαν redirect server για την άλλη.

## **7.SIP Μηνύματα**

Ένα SIP μήνυμα είναι είτε μια αίτηση από έναν φορέα σε έναν server είτε μια απάντηση από ένα server σε ένα φορέα.

Κάθε μήνυμα απάντησης και κάθε μήνυμα αιτήματος χρησιμοποιούν το βασικό σχέδιο του RFC 2822, αν και οι συντακτικά διαφέρουν στο σετ χαρακτήρων και στις συντακτικές οδηγίες. Και οι δύο τύποι μηνυμάτων αποτελούνται από μία αρχική γραμμή, ένα ή περισσότερα κύρια πεδία, μια κενή γραμμή που δείχνει το τέλος των κύριων πεδίων και ένα επιπλέον σώμα μηνύματος.

```
Generic-message=start line
    *message-header
    CRLF
    [message body]
Start-line=Request-line/status line
```

Τα ,start line (αρχική γραμμή), message-header line και οι κενές γραμμές πρέπει να τερματίζονται από από ένα CRLF (carriage return line-feed sequence). Σημαντικό είναι η κενή γραμμή να υπάρχει όταν δεν υπάρχει το σώμα μηνύματος.

Εκτός από την παραπάνω διαφορά στο σετ των χαρακτήρων, μεγάλο μέρος του SIP μηνύματος και του κύριου πεδίου είναι πανομοιότυπο είναι πανομοιότυπα με αυτά του HTTP/1.1. Παρόλα αυτά το SIP δεν είναι προέκταση του HTTP.

## **7.1SIP Αιτήσεις**

Οι SIP αιτήσεις διαχωρίζονται με το να έχουν Request-γραμμή αντί για Start-γραμμή. Μια Request-γραμμή περιέχει το όνομα μεθόδου, μια Request-URI και την έκδοση του πρωτοκόλλου που διαχωρίζεται από ένα χαρακτήρα κενού.

Η Request-γραμμή τελειώνει με έναν CRLF.

Request-γραμμή = Μέθοδος SP Request-URI SP SIP- Version CRLF

Method (Μέθοδος): Αυτός ο προσδιορισμός καθορίζει έξι μεθόδους: REGISTER για την καταγραφή πληροφοριών επαφής, INVITE, ACK και CANCEL για την εγκατάσταση συνδιαλέξεων, BYE για τον τερματισμό συνδιαλέξεων και OPTIONS για ρωτήσει για τις δυνατότητες των server. Κάποιες SIP προεκτάσεις ορίζουν και επιπλέον μεθόδους.

Request-URI (URI-Αίτησης): Το Request-URI είναι μια SIP ή SIPS URI που δείχνει τον χρήστη στον οποίο προορίζεται η αίτηση.

Τα SIP στοιχεία MAY (ΙΣΩΣ) υποστηρίζουν Request-URIs οι οποίες δεν έχουν «sip» ή «sips» διάταξη, όπως για παράδειγμα την διάταξη «tel». Τα SIP στοιχεία MAY ερμηνεύουν τις μη-SIP αιτήσεις χρησιμοποιώντας οποιονδήποτε διαθέσιμο μηχανισμό, καταλήγοντας σε αιτήσεις με SIP ή SIPS URI, ή σε άλλο συμβατό τύπο.

SIP-Version (SIP-Έκδοση): Τα μηνύματα αιτήσεων καθώς και απαντήσεων και τα δύο περιέχουν την έκδοση του SIP που χρησιμοποιούν και ακολουθούν την εκάστοτε έκδοση (το HTTP αντικαθίσταται από το SIP, το HTTP/1.1 αντικαθίσταται από το SIP/2.0)

προβαίνοντας σε συμβιβαστικές απαιτήσεις και αναβαθμίζοντας τους αριθμούς έκδοσης.

## **7.2SIP Απαντήσεις**

Οι απαντήσεις SIP διαχωρίζονται από τις SIP αιτήσεις με το να έχουν μια γραμμή-Status στην αρχική τιμή. Η γραμμή Status αποτελείται από την έκδοση πρωτοκόλλου η οποία ακολουθείται από έναν αριθμητικό κώδικα Status και την σχετική με αυτήν γραφική φράση, του οποίου το κάθε στοιχείο διαχωρίζεται από έναν SP χαρακτήρα.

Status-γραμμή=SIP-Version---SP---Status Code---SP---Reason Phrase  
CRLF

Ο κωδικός Status είναι ένας ακέραιος 3<sup>ων</sup> ψηφίων ο οποίος δείχνει το αποτέλεσμα μιας προσπάθειας κατανόησης και ικανοποίησης μιας αίτησης. Η Reason Phrase έχει ως σκοπό να δώσει μια μικρή περιγραφή του Status Code (κωδικού). Ο Status Code προορίζεται για αυτόματη χρήση ενώ η Reason Phrase προορίζεται για τον χρήστη. Ο φορέας δεν απαιτείται για την επίδειξη ή την επεξεργασία της Reason Phrase.

Το πρώτο ψηφίο του Status Code ορίζει τον τύπο (κλάση) της απάντησης. Τα δύο τελευταία ψηφία δεν έχουν κανένα ρόλο στην κατηγοριοποίηση. Για αυτόν το λόγο κάθε απάντηση με Status Code μεταξύ «100» και «199» αναφέρονται ως απαντήσεις κλάσης «1xx». Το ίδιο ισχύει για τις απαντήσεις «200» ως «299» (κλάση 2xx) και ούτο καθεξής. Η έκδοση SIP/2.0 επιτρέπει την χρήση έξι τιμών για το πρώτο ψηφίο.

1xx(Προσωρινή): Η αίτηση έχει φτάσει, συνέχεια σε επεξεργασία της αίτησης

2xx(Επιτυχία): Η πράξη έχει κατανοηθεί και έχει γίνει αποδεχτή.

3xx(Αποστολή σε νέα κατεύθυνση): Επιπλέον κινήσεις πρέπει να γίνουν ώστε η αίτηση να ολοκληρωθεί

4xx(Σφάλμα φορέα): Η αίτηση περιέχει λάθη στην σύνταξη ή δεν μπορεί να εκπληρωθεί σε αυτόν τον server.

5xx(Σφάλμα Server): Ο server δεν κατάφερε να εκπληρώσει μια φαινομενικά σωστή αίτηση.



δxx(Γενικό Σφάλμα): Η αίτηση δεν μπορεί να πραγματοποιηθεί σε κανένα server.

### **7.3 Κύρια Πεδία (Header Fields)**

Τα κύρια πεδία στο SIP είναι πανομοιότυπα με αυτά του HTTP και στην σύνταξη αλλά και στην σημασιολογία. Πιο συγκεκριμένα στο SIP τα κύρια πεδία ακολουθούν τις προδιαγραφές του [H4.2] στην σύνταξη καθώς και τους κανόνες επέκτασης των Header field σε πολλαπλές γραμμές. Το [H4.2] επίσης ορίζει ότι πολλαπλά κύρια πεδία με κοινό όνομα πεδίου του οποίου η τιμή διαχωρίζεται από κόμμα, μπορούν να συνδυαστούν σε ένα κύριο πεδίο. Αυτό ισχύει επίσης και για το SIP αλλά με μια διαφορά στο κανόνα καθώς εδώ έχουμε διαφορετική γραμματική. Συγκεκριμένα για κάθε SIP πεδίο που η γραμματική του είναι της μορφής:

```
Header="Header Name"HCOLON header value*(COMMA header value)
```

Επιτρέπεται ο συνδυασμός των πεδίων με ίδιο όνομα πεδίου του οποίου η τιμή διαχωρίζεται με κόμμα, σε ένα κύριο πεδίο.

#### **7.3.1 Μορφή Κύριων Πεδίων (Header Fields Format)**

Τα κύρια πεδία ακολουθούν τον ίδιο γενικό τύπο σχεδόν όλων των κύριων πεδίων. Αποτελούνται από ένα όνομα πεδίου (field name) το οποίο ακολουθείται από το σύμβολο ":" και την τιμή του πεδίου (field value).

field name: field value

Η τυπική γραμματική ενός πεδίου-μήνυμα επιτρέπει ένα αυθαίρετο αριθμό από κενά διαστήματα και στις δύο πλευρές του συμβόλου ":". Ωστόσο συνήθως αποφεύγονται οι κενοί χαρακτήρες μεταξύ του name και του ":" και χρησιμοποιείται ένα μόνο κενό

```
Subject:    lunch
Subject   :  lunch
Subject    :lunch
Subject: lunch
```

Από τα παραπάνω παραδείγματα, όλα είναι σωστά αλλά η τελευταία μορφή είναι αυτή που συνηθίζεται.

Τα κύρια πεδία μπορεί να προεκτείνονται σε πολλές γραμμές με το να ξεκινούν κάθε επιπλέον γραμμή με κενό χαρακτήρα (SP). Το παρακάτω παράδειγμα δείχνει το πώς ένα κύριο πεδίο μπορεί να χωριστεί σε επιπλέον γραμμές.

Subject: Σήκωσε το ακουστικό, ξέρω ότι είσαι εκεί  
Subject: Σήκωσε το ακουστικό,  
ξέρω  
ότι είσαι εκεί

Η σειρά των πεδίων Header με διαφορετικά ονόματα δεν είναι ιδιαίτερα σημαντική. Ωστόσο συνιστάται τα πεδία Header τα οποία απαιτούνται για την επεξεργασία proxy (Via, Route, Record Route...) να εμφανίζονται στην κορυφή του μηνύματος για να γίνεται γρηγορότερα και η επεξεργασία του. Αντίθετα η σειρά των πεδίων Header με το ίδιο όνομα έχει μεγάλη σημασία. Πολλαπλά κύρια πεδία μπορούν να εμφανίζονται σε ένα μήνυμα μόνο όταν ολόκληρη η τιμή του πεδίου ορίζεται σαν μια λίστα διαχωρισμένη από κόμμα. Θα πρέπει να είναι δυνατό να ενωθούν αυτά τα πεδία σε ένα ζευγάρι τύπου

field name: field value

χωρίς να αλλάξει η σημασιολογία του μηνύματος με το να προσαρμόζεται η επόμενη τιμή πεδίου (field value) στην πρώτη διαχωρίζοντας τις μεταξύ τους με κόμμα. Οι εξαιρέσεις σε αυτόν τον κανόνα είναι τα κύρια πεδία των μηνυμάτων Επικύρωσης, Έγκρισης, Proxy-επικύρωσης, Proxy-έγκρισης. Τα κύρια πεδία των παραπάνω μηνυμάτων δεν μπορούν να συνδυαστούν σε ένα λόγω διαφορετικής γραμματικής και σύνταξης.

Οι εφαρμογές θα πρέπει να είναι σε θέση να επεξεργαστούν πολλαπλές σειρές κύριων πεδίων με το ίδιο όνομα, σε οποιονδήποτε συνδυασμό της μιας τιμής ανά γραμμή ή των μορφών που διαχωρίζονται με κόμμα.

Τα παρακάτω γκρουπ σειρών κύριων πεδίων είναι σωστά και ισότιμα μεταξύ τους:

Route: <sip:Μαρία@forthnet.com>  
Subject: Lunch  
Route: <sip:Γιώργος@altecnet.com>  
Route: <sip:Μάριος@vivodi.com>

Route: <sip:Μαρία@forthnet.com>, <sip:Γιώργος@altecnet.com>

Route: <sip:Μάριος@vivodi.com>  
Subject: Lunch

Subject: Lunch  
Route: <sip:Μαρία@forthnet.com>, <sip:Γιώργος@altecnet.com>,  
<sip:Μάριος@vivodi.com>

Τα παρακάτω γκρουπ σειρών κύριων πεδίων είναι σωστά αλλά όχι ισότιμα μεταξύ τους:

Route: <sip:Μαρία@forthnet.com>  
Route: <sip:Γιώργος@altecnet.com>  
Route: <sip:Μάριος@vivodi.com>

Route: <sip:Γιώργος@altecnet.com>  
Route: <sip:Μαρία@forthnet.com>  
Route: <sip:Μάριος@vivodi.com>

Route: <sip:Μαρία@forthnet.com>, <sip:Μάριος@vivodi.com>,  
<sip:Γιώργος@altecnet.com>

Η μορφή μιας τιμής κυρίου πεδίου ορίζεται από το όνομα πεδίου. Είτε θα είναι πάντα μια δυσνόητη ακολουθία από μια οκτάδα χαρακτήρων κειμένου, είτε ένας συνδυασμός από κενούς χαρακτήρες, σύμβολα, διαχωριστές και αποσπάσματα ακολουθιών. Πολλά υπάρχουν κύρια πεδία εμμένουν στην γενική μορφή μιας τιμής η οποία ακολουθείται από μία ακολουθία με το ζευγάρι της παραμέτρου ονόματος και της παραμέτρου τιμής, διαχωρισμένη από ένα δίστιγμα. Δηλαδή της παρακάτω μορφής:

field-name: field-value \*(;parameter-name=parameter-value)

Παρόλο που ένας αυθαίρετος αριθμός από ζευγάρια παραμέτρων μπορούν να προσκολληθούν στην τιμή ενός κύριου πεδίου, μια παράμετρος ονόματος δεν πρέπει να εμφανίζεται παραπάνω από μία φορά σε κάθε τέτοια τιμή.

Όταν συγκρίνουμε κύρια πεδία, τα ονόματα των πεδίων είναι πάντα **case-insensitive**. Εκτός και αν έχει οριστεί διαφορετικά οι τιμές πεδίων, οι παράμετροι ονομάτων και οι παράμετροι τιμών είναι **case-insensitive**. Τα

σύμβολα είναι πάντα **case-insensitive**. Επίσης εκτός και αν έχει οριστεί διαφορετικά, οι τιμές που εκφράζονται σαν απόσπασμα ακολουθιών είναι **case-insensitive**. Για παράδειγμα:

Contact: <sip:Μαρία@forthnet.com>;expires=3600  
είναι ίσο με το:

CONTACT: < sip:Μαρία@forthnet.com>;ExPiReS=3600

Και το:

Content-Disposition: session;hanling=optional

Είναι ίσο με το:

content-disposition: Session;HANDLING=OPTIONAL

Τα παρακάτω δύο κύρια πεδία δεν είναι ίσα μεταξύ τους:

Warning: 370 devnull “Choose a bigger pipe”

Warning: 370 devnull “CHOOSE A BIGGER PIPE”

### **7.3.2 Κατηγοριοποίηση Κύριων Πεδίων**

Μερικά κύρια πεδία έχουν νόημα σε αιτήματα και σε απαντήσεις. Αυτά λέγονται κύρια πεδία αιτημάτων και απαντήσεων αντίστοιχα. Αν ένα κύριο πεδίο εμφανίζεται σε ένα μήνυμα άλλου τύπου (για παράδειγμα ένα κύριο πεδίο αιτήματος εμφανίζεται σε μια απάντηση) τότε πρέπει να αγνοείται.

### **7.3.3 Συμπυκνωμένη Μορφή**

Το SIP έχει μηχανισμό για να παρουσιάζει κοινά ονόματα κυρίων πεδίων σε μια συντομευμένη μορφή. Αυτό είναι χρήσιμο όταν τα μηνύματα είναι πολύ μεγάλα ώστε να γίνεται δύσκολη έως αδύνατη η μεταφορά τους από το εκάστοτε μέσο (π.χ όταν υπερβαίνουν το μέγιστο μέγεθος μεταφοράς όταν γίνεται χρήση UDP). Αυτές οι συμπυκνωμένες μορφές μπορούν να αντικατασταθούν από τις μεγαλύτερες εκδοχές τους οποιαδήποτε στιγμή χωρίς να αλλάξει η σημασία του μηνύματος. Επίσης το όνομα ενός κύριου πεδίου μπορεί να υπάρχει σε ένα μήνυμα και με την συμπυκνωμένη και με την μεγάλη εκδοχή του ταυτόχρονα.

## **7.4 Σώματα**

Οι αιτήσεις μπορεί να περιέχουν σώματα μηνυμάτων εκτός και αν έχει προσδιοριστεί το αντίθετο. Η ερμηνεία αυτών των σωμάτων εξαρτάται από την μέθοδο αίτησης.

Για τα μηνύματα απαντήσεων η μέθοδος αιτήσεως και ο κώδικας καταστάσεως απάντησης καθορίζουν τον τύπο της ερμηνείας για το κάθε σώμα μηνύματος. Επίσης όλες οι αιτήσεις μπορούν να περιλαμβάνουν σώματα.

### **7.4.1 Τύπος Σωμάτων**

Ο τύπος του σώματος μηνύματος πρέπει να καθορίζεται από το Content-Type κύριο πεδίο. Αν το σώμα έχει υποστεί κάποιου είδους κωδικοποίηση όπως συμπίεση, τότε ο τύπος του σώματος θα πρέπει να καθορίζεται από το Content-Encoding κύριο πεδίο, σε διαφορετική περίπτωση το Content-Encoding κύριο πεδίο πρέπει να παραλείπεται. Αν αυτό είναι εφαρμόσιμο, το σέτ χαρακτήρων του σώματος μηνύματος θεωρείται σαν μέρος της τιμής του Content-Type κύριου πεδίου. Τα SIP μηνύματα μπορεί να περιέχουν δυαδικά σώματα ή τμήματα σωμάτων.

### **7.4.2 Μήκος Σώματος Μηνύματος**

Το μήκος σώματος σε byte παρέχεται από το πεδίο Content-Length. Η «χοντροκομμένη» κωδικοποίηση μεταφοράς του HTTP/1.1 δεν πρέπει να χρησιμοποιηθεί στο SIP. (Σημείωση η «χοντροκομμένη» κωδικοποίηση αλλάζει το σώμα του μηνύματος με σκοπό να το μεταφέρει σε κομμάτια με ξεχωριστό μέγεθος το καθένα)

## **7.5 Σχηματισμός SIP Μηνύματος**

Σε αντίθεση με το HTTP, οι SIP εφαρμογές μπορούν να χρησιμοποιήσουν UDP ή και άλλα πρωτόκολλα για διάγραμμα δεδομένων. Κάθε τέτοιο διάγραμμα δεδομένων μεταφέρει μία αίτηση ή απάντηση.

Οι εφαρμογές που επεξεργάζονται SIP μηνύματα σε ρεύμα – προσανατολισμένες μεταφορές πρέπει να αγνοούν κάθε CRLF το οποίο εμφανίζεται πριν από κάθε γραμμή εκκίνησης.

Η τιμή του πεδίου Content-Length χρησιμοποιείται για να εντοπίσει το τέλος κάθε SIP μηνύματος σε ένα ρεύμα. Θα είναι πάντα παρών όταν τα SIP μηνύματα στέλνονται μέσω ρεύμα – προσανατολισμένων μεταφορών.

## **8. Γενική Συμπεριφορά Φορέα Χρήστη**

Ένας φορέας χρήστη αντιπροσωπεύει ένα τερματικό σύστημα. Περιέχει πληροφορίες έναν UAC, οποίος παράγει αιτήσεις, και έναν UAS ο οποίος τις απαντά. Ο UAC είναι ικανός να δημιουργεί μια αίτηση βασισμένη σε κάποιο εξωτερικό ερέθισμα (πχ ο χρήστης πατά ένα πλήκτρο) και επεξεργάζεται μια απάντηση. Ο UAS είναι ικανός να δέχεται μια αίτηση και να παράγει μια απάντηση βασισμένη σε εξωτερικά ερεθίσματα, δεδομένα που εισάγει ο χρήστης σε αποτελέσματα από εκτέλεση ενεργειών, ή από άλλου τύπου μηχανισμό.

Όταν ο UAC στέλνει μια αίτηση, η αίτηση περνά από κάποιους proxy servers οι οποίοι προωθούν την αίτηση στον UAS. Όταν ο UAS παράγει την απάντηση, αυτή αποστέλλεται πίσω στον UAC.

Οι UAC και UAS διαδικασίες εξαρτώνται από δύο παράγοντες. Πρώτον από το γεγονός αν η αίτηση ή απάντηση είναι εντός ή εκτός του διαλόγου και δεύτερον βασίζονται στην μέθοδο της αίτησης. Οι διάλογοι εκφράζουν μια ομότιμη σχέση μεταξύ των χρηστών και εγκαθίσταται από συγκεκριμένες SIP μεθόδους όπως την INVITE. Σε αυτό το τμήμα θα μιλήσουμε για τους κανόνες (ανεξαρτήτου μεθόδου) για την συμπεριφορά των UAC και UAS όταν επεξεργάζονται αιτήσεις που δεν ανήκουν σε κάποιον διάλογο. Αυτό συμπεριλαμβάνει και τις αιτήσεις που ως σκοπό έχουν την εγκατάσταση διαλόγου. Οι διαδικασίες ασφαλείας για αιτήσεις και απαντήσεις εκτός διαλόγου στηρίζονται σε υπάρχοντες μηχανισμούς για αμοιβαία επικύρωση.

### **8.1. Συμπεριφορά UAC**

Αυτό το τμήμα έχει σκοπό να καλύψει την UAC συμπεριφορά όταν είναι εκτός ενός διαλόγου.

#### **8.1.1 Δημιουργώντας την Αίτηση**

Μια σωστή SIP αίτηση διατυπωμένη από έναν UAC πρέπει να περιέχει τουλάχιστον τα παρακάτω κύρια πεδία: To, From, CSeq, Call-ID, Max-Forwards και το Via. Όλα αυτά τα πεδία είναι υποχρεωτικά σε όλες τις SIP αιτήσεις. Αυτά τα έξι κύρια πεδία είναι θεμελιώδη τμήματα ενός SIP μηνύματος καθώς όλα μαζί παρέχουν τις πιο ζωτικής σημασίας πληροφορίες για την δρομολόγηση υπηρεσιών όπως, την αποστολή απαντήσεων, την ελαχιστοποίηση της διάδοσης μηνυμάτων, την αναγνώριση των συναλλαγών.

Παραδείγματα αιτήσεων που αποστέλλονται εκτός διαλόγου περιλαμβάνουν μια αίτηση INVITE που ως σκοπό έχει την εγκατάσταση ενός διαλόγου, καθώς και μια αίτηση OPTIONS που έχει ως σκοπό να διερευνήσει τις δυνατότητες ενός server.

### **8.1.1.1 Request-URI**

Το αρχικό Request-URI του μηνύματος θα πρέπει να είναι ρυθμισμένο στην τιμή του URI του πεδίου To. Μια εξαίρεση σε αυτόν τον κανόνα είναι για την αίτηση REGISTER. Επίσης για λόγους μυστικότητας αυτά τα πεδία δεν θα πρέπει να έχουν την ίδια τιμή (ειδικά όταν ο φορέας χρήστη που στέλνει την αίτηση περιμένει πως η Request-URI θα αλλάξει κατά την διέλευση).

Σε μερικές ειδικές περιπτώσεις, η παρουσία μια ήδη υπάρχουσας διαδρομής-σετ (route-set) μπορεί να επηρεάσει την Request-URI ενός μηνύματος. Μια προϋπάρχουσα διαδρομή-σετ (route-set) είναι ένα σετ URI που αναγνωρίζουν μια αλυσίδα servers, στους οποίους ένας UAC στέλνει εξερχόμενες αιτήσεις οι οποίες είναι εκτός του διαλόγου. Συνήθως αυτές ρυθμίζονται από έναν χρήστη ή έναν παροχέα υπηρεσιών χειροκίνητα ή από κάποιον μη-SIP μηχανισμό.

### **8.1.1.2 Πεδίο «To»**

Το κύριο πεδίο «To» είναι το πρώτο που διευκρινίζει τον επιζητούμενο «λογικό» παραλήπτη της αίτησης, ή την διεύθυνση του στόχου της αίτησης. Ο αποδέκτης αυτός μπορεί να μην είναι και ο καλύτερος για αυτήν την αίτηση. Το κύριο πεδίο «To» μπορεί να περιέχει μια SIP ή SIPS URI αλλά επίσης μπορεί να κάνει χρήση άλλων σχημάτων URI όταν αυτό απαιτείται. Όλες οι SIP εφαρμογές πρέπει να υποστηρίζουν το σχήμα των SIP URI. Κάθε εφαρμογή που υποστηρίζει το σχήμα TLS

πρέπει και να υποστηρίζει το SIPS URI σχήμα. Επίσης το πεδίο «To» επιτρέπει την χρήση ονόματος επίδειξης.

Ένας UAC μπορεί να μάθει να παράγει το πεδίο «To» για μια αίτηση με πολλούς τρόπους. Συνήθως ο χρήστης εισάγει την URI χειροκίνητα ή επιλέγοντας την από μία λίστα διευθύνσεων. Συνήθως ο χρήστης δεν εισάγει την URI αυτούσια αλλά εισάγει μια σειρά ψηφίων ή χαρακτήρων (για παράδειγμα "Γιώργος"). Έτσι είναι στην κρίση του UA για το πώς θα ερμηνεύσει αυτό το δεδομένο. Χρησιμοποιώντας αυτήν την σειρά ψηφίων για να φτιάξουμε SIP URI υπονοεί πως ο UA επιθυμεί το όνομα να προστεθεί στο δεξιό τμήμα του SIP URI πριν το domain name. Αυτό έχει και το πλεονέκτημα ότι επιτρέπει στον τομέα να αναγνωρίζει εύκολα τον χρήστη που αποστέλλει μια αίτηση και να την επεξεργάζεται ανάλογα. Αυτό επίσης είναι χρήσιμο και σε λειτουργίες όπως «γρήγορη κλήση» η οποία απαιτεί ερμηνεία για το όνομα χρήστη από τον τομέα (domain). Επίσης μπορεί να γίνει και χρήση της URL όταν ο UA δεν επιθυμεί τον τομέα που θα ερμηνεύσει τον τηλεφωνικό αριθμό που έχει επιλέξει και εισάγει ο χρήστης. Έτσι σε κάθε τομέα (domain) από τον οποίο θα περνά η αίτηση θα του δίνεται η παραπάνω δυνατότητα. Για παράδειγμα ας θεωρήσουμε έναν χρήστη που βρίσκεται σε ένα αεροδρόμιο και επιθυμεί να συνδεθεί και να στείλει αιτήσεις μέσω ενός proxy ο οποίος δεν ανήκει στον τομέα (domain) που ανήκει ο χρήστης. Αν ο χρήστης πληκτρολογήσει "411" (ο οποίος αριθμός αντιπροσωπεύει τον στις υπηρεσίες καταλόγου των Ηνωμένων Πολιτειών) τότε αυτός ο αριθμός πρέπει να ερμηνευτεί και να από τον proxy εκτός ορίων και όχι από proxy του τομέα του χρήστη.

Μια αίτηση που βρίσκεται εκτός ενός διαλόγου δεν θα πρέπει να περιέχει την ετικέτα «To» μέσα στο πεδίο. Αυτή η ετικέτα προσδιορίζει τον αποδέκτη ενός διαλόγου. Από την στιγμή που δεν υπάρχει διάλογος, αυτή η ετικέτα δεν υπάρχει.

Το παρακάτω παράδειγμα είναι ένα σωστό (συντακτικά και νοηματικά) πεδίο «To».

To: Ελένη <sip:Ελένη@Αθήνα.com>

### **8.1.1.3 Πεδίο «From»**

Το κύριο πεδίο «From» προσδιορίζει την λογική ταυτότητα του αποστολέα της αίτησης, πιθανώς την διεύθυνση του χρήστη. Όπως και το πεδίο «To» το πεδίο «From» περιέχει μια URI καθώς και προαιρετικό όνομα επίδειξης. Χρησιμοποιείται από SIP στοιχεία για τον



προσδιορισμό των κανόνων επεξεργασίας που θα εφαρμόσουν σε μια αίτηση (για παράδειγμα αυτόματη απόρριψη κλήσης). Για αυτό είναι σημαντικό η URI From να μην περιέχει τις IP διευθύνσεις του οικοδεσπότη στον οποίο ο UA τρέχει, μιας και αυτά τα ονόματα δεν είναι λογικές οντότητες.

Το πεδίο «From» επιτρέπει την χρήση για όνομα επίδειξης. Ένας UAC θα πρέπει να χρησιμοποιεί για όνομα επίδειξης το «Ανώνυμος» μαζί με μια συντακτικά σωστή αλλά χωρίς νόημα URI αν επιθυμούμε η ταυτότητα του φορέα να παραμείνει κρυφή.( για παράδειγμα sip:thisis@Ανώνυμος.invalid)

Συνήθως η τιμή που περιέχεται στο πεδίο «From» στις αιτήσεις που παράγονται από συγκεκριμένο UA, παρέχεται από τον χρήστη ή από τους διαχειριστές του τομέα στον οποίο ανήκει ο χρήστης. Αν ένας συγκεκριμένος UA χρησιμοποιείται από πολλαπλούς χρήστες, μπορεί να έχει εναλλασσόμενα προφίλ τα οποία περιέχουν μια URI το καθένα η οποία αντιστοιχεί στην ταυτότητα κάθε χρήστη. Οι αποδέκτες των αιτήσεων που προέρχονται από τέτοιους UA μπορούν να ξεχωρίσουν το αποστολέα της αίτησης με σκοπό να επιβεβαιώσουν ότι ο αποστολέας είναι αυτός που ισχυρίζεται πως είναι από το πεδίο «From».

Το πεδίο «From» πρέπει να περιέχει μια νέα «tag» παράμετρο η οποία επιλέγεται από τον UAC. Παρακάτω παραθέτω ένα παράδειγμα

#### **8.1.1.4 Πεδίο «Call-ID»**

Το πεδίο Call-ID δρα ως ένας προσδιοριστής για να ομαδοποιήσει μια σειρά μηνυμάτων. Το πεδίο πρέπει να παραμένει το ίδιο σε κάθε αίτηση ή απάντηση η οποία στέλνεται από τον ίδιο UA σε έναν διάλογο. Επίσης πρέπει να είναι το ίδιο για κάθε καταχώρηση από τον UA.

Για μια νέα αίτηση από έναν UAC που βρίσκεται εκτός διαλόγου το κύριο πεδίο Call-ID πρέπει να επιλέγεται από τον UAC ως ένας παγκόσμια μοναδικός προσδιοριστής πέρα από το χώρο και τον χρόνο. Όλοι οι SIP UA πρέπει να έχουν την δυνατότητα να εγγυηθούν ότι τα Call-ID πεδία που παράγουν δεν θα παράγονται και από άλλους UA. Πρέπει να σημειωθεί ότι αιτήσεις που ξαναπροσπαθούν να αποσταλούν εξαιτίας απαντήσεων σφάλματος δεν θεωρούνται νέες αιτήσεις και για αυτό το Call-ID πεδίο παραμένει το ίδιο.

Η χρήση κρυπτογραφημένων τυχαίων προσδιοριστών για την παραγωγή Call-ID πεδίων συνιστάται. Εφαρμογές, μπορούν να χρησιμοποιήσουν την μορφή «localid@host». Τα Call-ID είναι ευαίσθητα πεδία και συγκρίνονται byte προς byte.

Με το να γίνεται χρήση κρυπτογραφημένων τυχαίων προσδιοριστών, παρέχεται μιας μορφής προστασίας από ενάντια στην πειρατεία συνδιαλέξεων και επίσης μειώνεται η πιθανότητα για ανεπιθύμητες συγκρούσεις των Call-ID.

Ένα παράδειγμα ενός Call-ID πεδίου φαίνεται παρακάτω:

Call-ID: f81d4fae-7dec-11d0-a765-00a0c91e6bf6@foo.bar.com

### **8.1.1.5 Πεδίο «CSeq»**

Το Cseq πεδίο δρα ως ένας τρόπος για να αναγνωρίσουμε και να προετοιμάσουμε συναλλαγές. Αποτελείται από έναν αριθμό ακολουθία και μια μέθοδο. Επιβάλλεται η μέθοδος να ταιριάζει με αυτήν της αίτησης. Για αιτήσεις μη-καταχώρησης και εκτός διαλόγου, η τιμή του αριθμού-ακολουθία πρέπει να είναι arbitrary. Η τιμή του αριθμού-ακολουθία πρέπει να εκφράζεται ως ένας 32-bit ακέραιος και πρέπει να είναι μικρότερος από  $2^{*}31$ . Εφόσον ακολουθεί τις παραπάνω οδηγίες, ένας φορέας μπορεί να χρησιμοποιήσει οποιονδήποτε μηχανισμό θέλει για να επιλέξει τις τιμές του CSeq πεδίου.

Παράδειγμα CSeq πεδίου

CSeq: 4711 INVITE

### **8.1.1.6 Πεδίο «Max-Forwards»**

Το πεδίο Max-Forwards δρα ως ένας περιοριστής στον αριθμό των αλμάτων που κάνει μια αίτηση μέχρι να φτάσει στον προορισμό της. Αποτελείται από έναν ακέραιο ο οποίος μειώνεται σε κάθε άλμα. Αν αυτή η τιμή μηδενιστεί προτού η αίτηση φτάσει στον προορισμό της, τότε η αίτηση θα απορριφθεί με μια απάντηση σφάλματος 483 (πολλά άλματα).

Ένας UAC πρέπει να εισάγει ένα πεδίο Max-Forwards σε κάθε αίτηση που παράγει και με τιμή που πρέπει να είναι ίση με 70. Αυτός ο αριθμός επιλέχθηκε ώστε να είναι αρκετά μεγάλος για να είναι σίγουρο πως η

αίτηση δεν θα σταματά σε οποιοδήποτε δίκτυο μόλις τελειώσουν τα άλματα, αλλά επίσης δεν είναι υπερβολικά μεγάλος ώστε να απορροφά όλους τους πόρους του proxy όταν συμβαίνουν άλματα. Μικρότερες τιμές σε αυτόν τον ακέραιο πρέπει να χρησιμοποιούνται με προσοχή και μόνο σε δίκτυα στα οποία η τοπολογία είναι γνωστή στους UA.

#### **8.1.1.7 Πεδίο «Via»**

Το πεδίο Via δείχνει το μέσο μεταφοράς που χρησιμοποιείται για την συναλλαγή και αναγνωρίζει την θέση στην οποία στέλνεται η απάντηση. Η τιμή σε ένα πεδίο Via προστίθεται μόνο όταν το μέσο μεταφοράς για το επόμενο άλμα έχει επιλεγθεί.

Όταν ο UAC δημιουργεί την αίτηση, πρέπει να εισάγει ένα Via πεδίο σε αυτήν. Το όνομα και η έκδοση του πρωτοκόλλου στο πεδίο πρέπει να είναι SIP και 0.2 αντίστοιχα. Η τιμή του πεδίου Via πρέπει να περιέχει μια παράμετρο κλάδου. Αυτή η παράμετρος χρησιμοποιείται για την αναγνώριση της συναλλαγής που δημιουργήθηκε από την αίτηση. Η παράμετρος αυτή χρησιμοποιείται και από τον φορέα και από τον server.

Η τιμή της παραμέτρου πρέπει να είναι μοναδική στον χώρο και στον χρόνο για όλες τις αιτήσεις του UA. Οι εξαιρέσεις σε αυτόν τον κανόνα είναι για τις αιτήσεις CANCEL και ACK για τις μη-2xx τύπου απαντήσεις. Για παράδειγμα η αίτηση CANCEL θα έχει την ίδια παράμετρο με την αίτηση που ακυρώνει. Επίσης αντίστοιχα η ACK έχει την ίδια παράμετρο με την INVITE στην οποία απαντάει.

Αυτή η ID παράμετρος όταν εισάγεται σε ένα στοιχείο που ακολουθεί τους παραπάνω κανόνες ξεκινά πάντα με τους χαρακτήρες “z9hG4bK”. Αυτοί οι 7 χαρακτήρες διασφαλίζουν για τους server που τους δέχονται ότι η ID παράμετρος είναι δομημένη με την μέθοδο που αναφέρεται παραπάνω.

#### **8.1.1.8 Πεδίο «Contact»**

Το πεδίο Contact παρέχει μια SIP ή SIPS URI η οποία μπορεί να χρησιμοποιηθεί για να επικοινωνήσει με τον UA για περαιτέρω αιτήσεις. Το πεδίο Contact πρέπει να εμφανίζεται σε κάθε αίτηση και να περιέχει επακριβώς την SIP ή SIPS URI η οποία καταλήγει στην εγκατάσταση ενός διαλόγου. Το Contact πεδίο περιέχει την URI στην οποία ο UA θα δέχεται αιτήσεις και για αυτόν τον λόγο η URI πρέπει να είναι σωστή ακόμα και όταν στέλνονται περαιτέρω αιτήσεις εκτός διαλόγου.

Αν η Request URI περιέχει SIPS URI τότε και το πεδίο Contact πρέπει να περιέχει SIPS URI.

### **8.1.1.9 Πεδία «Supported» και «Require»**

Αν ένας UAC υποστηρίζει επεκτάσεις του SIP που μπορούν να εφαρμοστούν από τον server σε μια απάντηση, τότε ο UAC πρέπει να συμπεριλάβει ένα πεδίο Supported μέσα στην αίτηση για αυτές τις επεκτάσεις.

Ωστόσο αυτό το πεδίο θα πρέπει να αναφέρεται σε επεκτάσεις που θεωρούνται πιστοποιημένες για το SIP. Και αυτό γίνεται για να αποτρέψει servers στο να παρέχουν υπηρεσίες σε φορείς που δεν χρησιμοποιούν πιστοποιημένες επεκτάσεις.

Όταν ένας UAC επιμένει πως ένας UAS καταλαβαίνει την επέκταση την οποία ο UAC θα εφαρμόσει σε μια αίτηση, τότε πρέπει να εισάγει ένα Request πεδίο στην συγκεκριμένη αίτηση καθορίζοντας τις ρυθμίσεις για την συγκεκριμένη αίτηση. Αν τώρα ο UAC επιθυμεί να εφαρμόσει μια επέκταση σε μια αίτηση και επιμένει πως όλοι οι proxies από τους οποίους θα περάσει η αίτηση θα κατανοήσουν την επέκταση, τότε πρέπει να εισάγει ένα Proxy-Request πεδίο στην αίτηση με τις ρυθμίσεις για την επέκταση. Τα παραπάνω όπως προανέφερα ισχύουν για τις πιστοποιημένες επεκτάσεις.

### **8.1.1.10 Επιπλέον στοιχεία μηνυμάτων**

Αφού μια νέα αίτηση έχει δημιουργηθεί και όλα τα πεδία είναι σωστά δομημένα, τότε προστίθενται όλα τα επιπλέον πεδία. Οι SIP αιτήσεις μπορεί να περιέχουν σώμα μηνύματος με κωδικοποίηση MIME. Ανεξαρτήτως το σώμα που μπορεί να περιέχουν οι αιτήσεις, ορισμένα κύρια πεδία μπορεί να είναι έτσι κατασκευασμένα ώστε να χαρακτηρίζουν τα περιεχόμενα του σώματος.

### **8.1.2 Αποστολή της αίτησης**

Στην συνέχεια εισάγεται ο προορισμός της αίτησης> Εκτός και αν υπάρχει μια συγκεκριμένη πολιτική που υποστηρίζει το αντίθετο, ο προορισμός θα πρέπει να καθορίζεται μέσω των DNS διαδικασιών που περιγράφονται στο τμήμα [4]. Αν το πρώτο στοιχείο του σετ διαδρομής

χαρακτηρίζεται ως ακριβής δρομολογητής, τότε οι διαδικασίες πρέπει να εφαρμόζονται στο Request-URI της αίτησης. Σε διαφορετική περίπτωση πρέπει να εφαρμόζονται στην πρώτη τιμή του Route πεδίου (αν υπάρχει) της αίτησης ή στο Request-URI αν δεν υπάρχει πεδίο Route. Αυτές οι διαδικασίες διατηρούν έναν αριθμό από διευθύνσεις, θύρες και μεταφορές τις οποίες προσπαθούν να πραγματοποιήσουν. Ανεξαρτήτως του πια URI χρησιμοποιείται για να εισαχθούν οι διαδικασίες του [4], αν η Request-URI δείχνει σε έναν SIPS πόρο, τότε ο UAC πρέπει να ακολουθήσει τις διαδικασίες του [4] σαν η είσοδος URI ήταν μια SIPS URI.

Συνήθως η κοινή πολιτική ορίζει ένα διαφορετικό σετ από προορισμούς για αποστολή. Αν η Request URI περιέχει μια SIPS URI, κάθε διαφορετικός προορισμός πρέπει να έρχεται σε επαφή μέσω TLS. Πέρα από αυτό, δεν υπάρχει άλλος περιορισμός στους εναλλακτικούς προορισμούς αν η αίτηση δεν περιέχει Route πεδίο. Αυτό παρέχει μια εναλλακτική λύση για ένα προϋπάρχον σετ διαδρομής, για τον ορισμό ενός proxy εκτός ορίων. Ωστόσο αυτού του τύπου η προσέγγιση για τους proxy εκτός ορίων δεν συνιστάται. Αντίθετα μπορούμε να κάνουμε χρήση ενός προϋπάρχον σετ διαδρομής με μία μόνο URI για τον ίδιο σκοπό. Αν η αίτηση περιέχει ένα Route πεδίο τότε η αίτηση πρέπει να σταλεί στις τοποθεσίες που αναφέρονται στις πρώτες τιμές του πεδίου. Ένας UAC ο οποίος έχει ρυθμιστεί σε έναν proxy εκτός ορίων θα πρέπει να δοκιμάσει να στείλει την αίτηση στην τοποθεσία που δείχνει η πρώτη τιμή του πεδίου Route, αντί να ακολουθήσει την πολιτική του να στείλει όλα τα μηνύματα στον proxy εκτός ορίων.

Αυτό διασφαλίζει ότι οι proxies εκτός ορίων οι οποίοι δεν προσθέτουν τιμές στο Record-Route πεδίο θα βγουν εκτός του δρομολογίου των περαιτέρω αιτήσεων. Αυτό επιτρέπει σε τερματικά τα οποία δεν μπορούν να επιλύσουν το πρώτο Route URI, να αναθέσουν αυτήν την πράξη σε proxies εκτός ορίων.

Ο UAC πρέπει να ακολουθεί τις διαδικασίες που αναλύονται στο [4] για στοιχεία με κατάσταση, δοκιμάζοντας κάθε διεύθυνση μέχρι να εγκατασταθεί επικοινωνία με τον server. Κάθε προσπάθεια αντιστοιχεί σε μια νέα συναλλαγή, και έτσι κάθε μία έχει και διαφορετική πρώτη τιμή στο πεδίο Via με μια νέα παράμετρο.

### **8.1.3 Επεξεργασία απαντήσεων**

Οι απαντήσεις πρώτα επεξεργάζονται από το στρώμα μεταφοράς και μετά προωθούνται στο στρώμα συναλλαγής. Μετά το στρώμα

συναλλαγής αφού κάνει την επεξεργασία προωθεί την απάντηση στον TU. Η σπουδαιότητα της επεξεργασίας της απάντησης από τον TU έχει να κάνει με την μέθοδο που χρησιμοποιείται. Ωστόσο υπάρχει και μια γενική συμπεριφορά ανεξαρτήτου της μεθόδου.

### **8.1.3.1 Σφάλματα Στρώματος Συναλλαγής**

Σε μερικές περιπτώσεις η απάντηση που επιστρέφει από το στρώμα συναλλαγής δεν θα είναι ένα SIP μήνυμα αλλά ένα σφάλμα του στρώματος συναλλαγής. Όταν ένα σφάλμα «τέλος χρόνου» παραλαμβάνεται από το στρώμα συναλλαγής, θα πρέπει να του συμπεριφέρεται όπως και σε ένα σφάλμα 408 (Τέλος χρόνου αίτησης). Αν ένα τερματικό σφάλμα μεταφοράς σημειωθεί από το στρώμα μεταφοράς τότε αυτό θα πρέπει να του συμπεριφερθεί όπως και σε ένα σφάλμα 503 (Υπηρεσία μη διαθέσιμη).

### **8.1.3.2 Μη αναγνωρίσιμες απαντήσεις**

Ένας UAC θα πρέπει να συμπεριφερθεί σε κάθε τελική απάντηση την οποία δεν αναγνωρίζει ως ίση με κώδικα απάντησης τύπου X00 και θα πρέπει να είναι σε θέση να επεξεργάζεται τέτοιου τύπου απαντήσεις. Για παράδειγμα αν ένας UAC δεχθεί μια μη αναγνωρισμένη απάντηση με κωδικό 431, μπορεί με σιγουριά να υποθέσει πως υπήρχε κάποιο σφάλμα στην αίτησή του και να συμπεριφερθεί στην απάντηση όπως θα έκανε και σε μια απάντηση τύπου 400 (Σφάλμα αίτησης). Ένας UAC πρέπει να συμπεριφέρεται σε κάθε προσωρινή απάντηση διαφορετικά από ότι σε μία τύπου 100την οποία δεν αναγνωρίζει ως 183 (Κατάσταση Συνδιάλεξης). Ένας UAC θα πρέπει να είναι σε θέση να επεξεργάζεται τις απαντήσεις τύπου 100 και 183.

### **8.1.3.3 Vias**

Αν εμφανίζονται περισσότερες από μία τιμές στο πεδίο Via σε μία απάντηση, τότε ο UAC θα πρέπει να απορρίψει το μήνυμα.

Η παρουσία επιπλέον τιμών στα πεδία Via σημαίνει ότι το μήνυμα ακολούθησε λάθος διαδρομή ή πιθανώς περιέχει σφάλματα.

### **8.1.3.4 Επεξεργασία 3XX απαντήσεων**

Όταν οι φορείς δέχονται απαντήσεις για επανααποστολή σε άλλο προορισμό (για παράδειγμα, μια 301 απάντηση) τότε πρέπει να

χρησιμοποιήσουν το URI στο πεδίο Contact για να δημιουργήσουν μία ή και περισσότερες νέες αιτήσεις για την αίτηση επαναποστολής που δέχθηκαν. Ένας φορέας συνήθως ξεκινά με ένα μόνο σετ στόχων το οποίο περιέχει μόνο μία URI, την Request-URI της αρχικής αίτησης. Αν ένας φορέας επιθυμεί να κατασκευάσει νέες αιτήσεις βασισμένες σε μία 3XX τύπου απάντηση για αυτό το αίτημα, τότε βάζει τις κατάλληλες URIs στο σετ στόχων. Ωστόσο δεν μπορεί να βάλει την ίδια URI παραπάνω από μια φορά στο σετ στόχων. Αν η αρχική αίτηση έχει SIPS URI στο πεδίο Request τότε ο φορέας μπορεί να επιλέξει να οδηγήσει την αίτηση σε μη-SIPS URI αλλά θα πρέπει να ενημερώσει τον χρήστη για την αλλαγή του προορισμού της αίτησης προς μια μη ασφαλή URI.

Κάθε νέα αίτηση μπορεί να δεχθεί απαντήσεις τύπου 3XX οι οποίες περιέχουν την αρχική URI ως σύνδεσμο. Δύο τοποθεσίες μπορούν να είναι ρυθμισμένες να οδηγεί η μία στην άλλη. Η τοποθέτηση μιας δοσμένης URI στο σετ στόχων μόνο μια φορά αποτρέπει το να έχουμε ατελείωτα κλειδωμένα άλματα.

Καθώς μεγαλώνει το σετ στόχων, ο φορέας μπορεί να παράγει νέες αιτήσεις για τα URI με οποιαδήποτε σειρά. Ένας κοινός μηχανισμός είναι Δούμε το σετ στόχων από την τιμή της παραμέτρου “q” του πεδίου Contact. Οι αιτήσεις για τα URIs μπορεί να παράγονται σειριακά ή παράλληλα. Μια προσέγγιση είναι να επεξεργαζόμαστε γκρουπ από φθίνουσες τιμές “q” σειριακά και να επεξεργαζόμαστε τις URIs σε κάθε “q” γκρουπ παράλληλα. Μια άλλη προσέγγιση να κάνουμε μόνο σειριακή επεξεργασία στις παραμέτρους “q” , διαλέγοντας μεταξύ επαφών με ίσες τιμές “q”.

Αν η προσπάθεια επικοινωνίας με μια διεύθυνση που περιέχεται στην λίστα, καταλήξει σε αποτυχία τότε το στοιχείο προχωρά στην επόμενη διεύθυνση στην λίστα μέχρι να εξαντληθούν όλες οι διευθύνσεις της λίστας. Αν όλες οι διευθύνσεις εξαντληθούν χωρίς αποτέλεσμα τότε η αίτηση απέτυχε.

Οι αποτυχίες θα πρέπει να ανιχνεύονται μέσω απαντήσεων αποτυχίας (απαντήσεις με κωδικό μεγαλύτερο από 399). Για σφάλματα δικτύου ο φορέας συναλλαγής θα αναφέρει κάθε τυχόν σφάλματα μεταφοράς στρώματος στον χρήστη συναλλαγής. Σημειώστε ότι κάποιοι κωδικοί απαντήσεων δείχνουν πως κάποιες αιτήσεις δεν μπορούν να σταλούν ξανά.

Όταν λαμβάνεται ένα σφάλμα για μια συγκεκριμένη διεύθυνση επαφής, τότε ο φορέας θα πρέπει να δοκιμάσει την επόμενη διεύθυνση επαφής.

Αυτό συμπεριλαμβάνει την δημιουργία νέου φορέα συναλλαγής για την αποστολή της νέας αίτησης.

Για την δημιουργία μιας νέας αίτησης βασισμένη σε μια διεύθυνση μια απάντησης 3XX, ο UAC πρέπει να αντιγράψει όλη την URI από το σετ στόχων στο Request-URI εκτός από τα σημεία που αναφέρονται στην «method-param» και «header-URI παραμέτρους. Χρησιμοποιεί τις παραμέτρους header για την δημιουργία τιμών για τα κύρια πεδία της αίτησης, διαγράφοντας τις παλαιότερες τιμές των πεδίων που σχετίζονται με την αίτηση προσανατολισμού (redirect).

Σε μερικές περιπτώσεις, τα κύρια πεδία τα οποία τα οποία έχουν επικοινωνήσει με την διεύθυνση επαφής, μπορεί αντίθετα να χρησιμοποιούν τα ήδη υπάρχουσα Request πεδία από την αρχική αίτηση προσανατολισμού. Ως γενικός κανόνας, αν το κύριο πεδίο μπορεί να δεχθεί μια λίστα τιμών διαχωρισμένη από κόμμα, τότε η νέα τιμή του πεδίου μπορεί να επισυνάπτετε στην αρχική αίτηση προσανατολισμού (redirect). Αν το κύριο πεδίο δεν δέχεται πολλαπλές τιμές, τότε η τιμή στην αρχική αίτηση προσανατολισμού θα παραγραφεί και στην θέση της θα προστεθεί αυτή του κύριου πεδίου από την διεύθυνση επαφής. Για παράδειγμα αν μια διεύθυνση επαφής επιστρέφει με την παρακάτω τιμή.:

sip: user@host?Subject=foo&Call-Info=<http://www.foo.com>

Τότε κάθε Subject κύριο πεδίο στην αρχική αίτηση προσανατολισμένη αίτηση μπορεί να διαγραφεί, αλλά η HTTP URL μερικώς επισυνάπτετε σε κάθε υπάρχουσα τιμή Call-Info πεδίου.

Συνιστάται, κάθε UAC να χρησιμοποιεί το ίδιο «To», «From», και «Call-ID» πεδίο που χρησιμοποιείται και στην αρχική αίτηση προσανατολισμού, αλλά ο UAC θα πρέπει έχει μια νέα Branch ID στην κορυφή του πεδίου Via.

Με άλλα λόγια, οι αιτήσεις που αποστέλλονται μετά από παραλαβή μιας απάντησης προσανατολισμού θα πρέπει να χρησιμοποιούν τα κύρια πεδία της αρχικής αίτησης απaráλλακτα.

### **8.1.3.5 Επεξεργασία 4XX απαντήσεων**

Κάποιες απαντήσεις τύπου 4XX απαιτούν συγκεκριμένο τρόπο επεξεργασίας από τους UA, ανεξαρτήτου μεθόδου.



Αν μια απάντηση 401 (Μη επικυρωμένο) ή 407 (Απαιτείται επικύρωση από Proxy) ληφθεί, τότε ο UAC θα πρέπει να ακολουθήσει συγκεκριμένες διαδικασίες επικύρωσης που αναφέρονται σε παρακάτω τμήματα.

Αν μια απάντηση 413 (Μεγάλο μέγεθος Αίτησης) ληφθεί τότε το μέγεθος της αίτησης ήταν μεγαλύτερο από αυτό που επιθυμούσε ο UAC να λάβει. Αν είναι δυνατόν ο UAC θα πρέπει να ξαναδοκιμάσει να αποστείλει την αίτηση αυτή τη φορά όμως με μικρότερο μέγεθος στο σώμα της αίτησης.

Αν μια απάντηση τύπου 415 (Μη υποστηριζόμενο μέσο) ληφθεί, τότε είναι προφανές ότι η αίτηση περιείχε μέσα τα οποία δεν υποστηρίζει ο UAS. Τότε ο UAC πρέπει να αποστείλει ξανά την αίτηση, αυτή τη φορά με περιεχόμενο του οποίου τον τύπο να αναγνωρίζει και να υποστηρίζει ο UAS. Αυτό γίνεται χρησιμοποιώντας τα στοιχεία των του πεδίου Accept, αναγνωρίζοντας ποιοι τύποι γίνονται αποδεκτοί από τον UAS.

Αν μια απάντηση τύπου 416 (Μη υποστηριζόμενη μορφή URI) ληφθεί, τότε το Request-URI πεδίο της αίτησης χρησιμοποίησε μια μορφή URI η οποία δεν υποστηρίζεται από τον server. Ο φορέας θα πρέπει να αποστείλει την αίτηση ξανά, αυτή την φορά χρησιμοποιώντας μια SIP URI.

Αν μια απάντηση τύπου 420 (Λάθος επέκταση) ληφθεί, τότε η αίτηση στο Request ή Proxy Request πεδίο της περιέχει μια παράμετρο στα options για μια δυνατότητα την οποία δεν υποστηρίζει ο Proxy ή ο UAS. Στην συγκεκριμένη περίπτωση η αίτηση αποστέλλεται ξανά χωρίς αυτές τις παραμέτρους.

Σε όλες τις παραπάνω περιπτώσεις η αίτηση αποστέλλεται ξανά, μέσω μιας νέας αίτησης αφού έχει υποστεί τις απαραίτητες τροποποιήσεις. Αυτή η νέα αίτηση ουσιαστικά αποτελεί νέα συναλλαγή και ενώ θα πρέπει να έχει τις ίδιες τιμές στα πεδία «To», «From», και «Call-ID» με την προηγούμενη αίτηση, το πεδίο CSeq θα πρέπει να έχει νέα τιμή η οποία να είναι μεγαλύτερη από την προηγούμενη.

Για τις υπόλοιπες απαντήσεις τύπου 4XX το δικαίωμα αποστολής ξανά της αίτησης δεν είναι πάντα διαθέσιμο, αλλά εξαρτάται από τον τύπο της μεθόδου που χρησιμοποιείται.

## **8.2 Συμπεριφορά UAS**

Όταν μια αίτηση εκτός ενός διαλόγου, επεξεργάζεται από έναν UAS ακολουθείται ένα σετ κανονισμών, ανεξαρτήτου της μεθόδου.

Σημειώστε πως η επεξεργασία της αίτησης είναι ατομική διαδικασία. Αν μια αίτηση αποδεχθεί, όλες οι αλλαγές καταστάσεως που την αφορούν πραγματοποιούνται. Αν απορριφθεί καμία αλλαγή δεν πραγματοποιείται.

Οι UAS πρέπει να επεξεργάζονται τις αιτήσεις με την σειρά την οποία εμφανίζεται σε αυτό το τμήμα (επικύρωση, επιθεώρηση μεθόδου,.....)

### **8.2.1 Επιθεώρηση Μεθόδου**

Από την στιγμή που η μέθοδος επικυρώνεται, ο UAS πρέπει να επιθεωρήσει την μέθοδο της αίτησης. Αν ο UAS αναγνωρίσει την μέθοδο αλλά δεν την υποστηρίζει, τότε πρέπει να δημιουργήσει μια απάντηση 405 (Μέθοδος μη επιτρεπόμενη). Επίσης ο UAS πρέπει να δημιουργήσει ένα πεδίο Allow στο μήνυμα 405 με την λίστα των επιτρεπόμενων μεθόδων που υποστηρίζει.

### **8.2.1 Επιθεώρηση Header**

Αν ένας UAS δεν κατανοεί το κύριο πεδίο σε μια αίτηση, τότε ο server πρέπει να αγνοήσει το συγκεκριμένο πεδίο και να συνεχίσει να επεξεργάζεται το μήνυμα. Ένας UAS θα πρέπει να αγνοήσει μη αναγνωρισμένα κύρια πεδία τα οποία δεν είναι απαραίτητα στην επεξεργασία αιτήσεων.

#### **8.2.2.1 «To» και «Request» URI**

Το κύριο πεδίο «To» αναγνωρίζει τον τελικό αποδέκτη της αίτησης η οποία στέλνεται από τον χρήστη που φαίνεται στο πεδίο «From». Ο τελικός αποδέκτης μπορεί να μην είναι ο UAS που επεξεργάζεται την αίτηση. Ο UAS μπορεί να εφαρμόσει συγκεκριμένη πολιτική στο αν θα δέχεται αιτήσεις των οποίων το πεδίο «To» δεν έχει την ταυτότητα του UAS. Ωστόσο συνιστάται ο UAS να δέχεται αιτήσεις ακόμα και όταν δεν αναγνωρίζουν την φόρμα της URI στο πεδίο «To», ή όταν το πεδίο «To» δεν υποδυκνίει έναν γνωστό ή τον εκάστοτε χρήστη αυτού του UAS. Στην περίπτωση που ο UAS αποφασίσει να απορρίψει την αίτηση, θα

πρέπει να δημιουργήσει μια απάντηση τύπου 403 (Απαγορευμένη) την οποία θα προωθήσει στον server για εκπομπή.

Ωστόσο το «Request» URI αναγνωρίζει τον UAS ο οποίος πρόκειται να επεξεργαστεί την αίτηση. Αν η «Request» URI χρησιμοποιεί μια φόρμα η οποία δεν υποστηρίζεται από τον UAS, θα πρέπει να απορρίψει την αίτηση με μια απάντηση τύπου 416 (Μη υποστηριζόμενη φόρμα URI). Αν η «Request» URI δεν αναγνωρίσει την διεύθυνση στην οποία ο UAS δέχεται αιτήσεις, θα πρέπει να απορρίψει την αίτηση με μια απάντηση τύπου 404 (Δεν εντοπίστηκε). Συνήθως ο UA ο οποίος χρησιμοποιεί την μέθοδο «REGISTER» (ΚΑΤΑΧΩΡΗΣΗ) βλέπει αιτήσεις των οποίων η «Request» URI είναι η διεύθυνση επαφής. Άλλες πιθανές πηγές αποδεχτών «Request» URIs είναι τα πεδία «Contact» των αιτήσεων και απαντήσεων οι οποίες στέλνονται από τον UA για την εγκατάσταση ή ανανέωση διαλόγων.

#### **8.2.2.2 Συγγωνευμένες αιτήσεις**

Αν η αίτηση δεν έχει ετικέτα στο πεδίο «To», τότε ο UAS πυρήνας πρέπει να ελέγξει την αίτηση για καθοδόν συναλλαγές. Αν η ετικέτα στο πεδίο «From», «Call-ID» και «CSeq» ταιριάζει απόλυτα με αυτήν που σχετίζεται με την καθοδόν συναλλαγή, αλλά η αίτηση δεν ταιριάζει με την συναλλαγή, τότε ο UAS πρέπει να δημιουργήσει μια απάντηση 482 (Ανιχνεύτηκε ατέρμονο άλμα) και να την προωθήσει στον server.

#### **8.2.2.3 Require**

Υποθέτοντας ότι ο UAS αποφασίζει πως είναι το κατάλληλο στοιχείο για την επεξεργασία μιας αίτησης, τότε εξετάζει το πεδίο Require αν υπάρχει.

Το πεδίο Require χρησιμοποιείται από τον UAC για να δείξει στον UAS τις SIP επεκτάσεις τις οποίες ο UAC περιμένει πως θα υποστηρίξει ο UAS ώστε να επεξεργαστεί σωστά την αίτηση. Αν ο UAS δεν καταλάβει την ετικέτα-επιλογές «options» στο πεδίο Require, θα πρέπει να απαντήσει στέλνοντας μια απάντηση τύπου 420 (Λάθος Επέκταση). Ο UAS θα πρέπει να εισάγει μια λίστα με τις μη-υποστηριζόμενες επεκτάσεις και με τις επιλογές τις οποίες δεν καταλαβαίνει από το Require πεδίο της αίτησης.

Να σημειώσουμε πως τα πεδία «Require» και «Proxy-Require» δεν πρέπει να χρησιμοποιούνται στις αιτήσεις CANCEL και ACK (αν η

τελευταία στέλνεται για μια απάντηση τύπου 2XX). Αυτά τα πεδία θα πρέπει να αγνοούνται αν περιέχονται σε τέτοιες αιτήσεις. Μια αίτηση ACK η οποία στέλνεται για απάντηση τύπου 2XX πρέπει να περιέχει τα πεδία Require ή Proxy-Require τα οποία υπήρχαν και στην αρχική αίτηση.

Παράδειγμα:

```
UAC→UAS  INVITE sip: Watson@bell-telephone.com SIP/2.0
          Require: 100rel
```

```
UAS→UAC  SIP/2.0 420 Bad Extension
          Unsupported: 100rel
```

Αυτή η συμπεριφορά διασφαλίζει ότι η αλληλεπίδραση του φορέα με τον server δεν θα καθυστερήσει μόλις όλες οι επιλογές κατανοηθούν και από τις δύο πλευρές, ενώ θα καθυστερήσει αν δεν γίνουν κατανοητές (όπως το παραπάνω παράδειγμα). Για ένα ταιριαστό ζευγάρι φορέα-server, η αλληλεπίδραση προχωρά γρήγορα. Επιπλέον αναιρεί την ασάφεια όταν ο φορέας απαιτεί στοιχεία τα οποία ο server δεν κατανοεί. Εξάλλου κάποια στοιχεία ενδιαφέρουν μόνο τα τερματικά συστήματα.

### **8.2.3 Επεξεργασία Περιεχομένου**

Υποθέτοντας πως ο UAS έχει αντιληφθεί όλες τις επεκτάσεις που απαιτούνται από φορέα, ο UAS εξετάζει το σώμα του μηνύματος και τα κύρια πεδία που το περιγράφουν. Αν υπάρχουν σώματα των οποίων ο τύπος, η γλώσσα και η κωδικοποίηση δεν γίνονται κατανοητά και των οποίων το σώμα δεν είναι προαιρετικό, τότε ο UAS πρέπει να απορρίψει την αίτηση με μια απάντηση τύπου 415 (Μη-υποστηριζόμενος τύπος Μέσου). Η απάντηση θα πρέπει να περιέχει ένα πεδίο Accept με τη λίστα όλων των σωμάτων που γίνονται αποδεκτά. Αν η αίτηση περιείχε κωδικοποίηση περιεχομένου η οποία δεν αναγνωριζόταν από τον UAS, τότε η απάντηση θα πρέπει να περιείχε Accept-Encoding πεδίο με την λίστα των αποδεκτών κωδικοποιήσεων. Αν το περιεχόμενο της αίτησης αποτελούνταν από γλώσσα μη κατανοητή από τον UAS, τότε η απάντηση πρέπει να περιέχει ένα πεδίο Accept-Languages με λίστα όλων των αναγνωρίσιμων γλωσσών από τον UAS. Εκτός από τους παραπάνω ελέγχους, η επεξεργασία του σώματος εξαρτάται από την μέθοδο και τον τύπο.

### **8.2.4 Εφαρμόζοντας επεκτάσεις**

Ένας UAS που επιθυμεί να εφαρμόσει επεκτάσεις όταν παράγει την απάντηση, δεν θα πρέπει να το κάνει εκτός και αν υποστηρίζεται, δηλαδή περιέχεται στην λίστα στο πεδίο Supported της αίτησης. Αν η επιθυμητή επέκταση δεν υποστηρίζεται, ο server θα πρέπει να βασιστεί μόνο σε βασικές αρχές του SIP και σε όποιες επεκτάσεις υποστηρίζονται από τον φορέα. Σε σπάνιες περιπτώσεις όπου ο server δεν θα μπορεί να επεξεργαστεί την αίτηση χωρίς την επέκταση, ο server μπορεί να στείλει μια απάντηση τύπου 421 (Απαιτείται Επέκταση). Αυτή η απάντηση δείχνει πως η σωστή απάντηση δεν μπορεί να παραχθεί δίχως την κατάλληλη επέκταση. Οι απαιτούμενες επεκτάσεις θα πρέπει να περιέχονται στο πεδίο Require της απάντησης.

Κάθε επέκταση που εφαρμόζεται σε μια μη-421 τύπου απάντηση θα πρέπει να περιέχεται στην λίστα των επιτρεπτών επεκτάσεων του πεδίου Require της απάντησης. Φυσικά ο server δεν θα πρέπει να συμπεριλαμβάνει επεκτάσεις που δεν περιέχονται στη λίστα του πεδίου Supported της αίτησης.

### **8.2.5 Επεξεργασία της αίτησης**

Υποθέτοντας πως όλοι οι έλεγχοι των παραπάνω τμημάτων έχουν ολοκληρωθεί επιτυχώς, ο UA επεξεργάζεται την αίτηση σύμφωνα με την μέθοδο που χρησιμοποιεί. Σε επόμενα τμήματα θα καλύψουμε την επεξεργασία της αίτησης ανάλογα με την μέθοδο αυτής.

### **8.2.6 Δημιουργώντας την απάντηση**

Όταν ο UAS επιθυμεί να δημιουργήσει μια απάντηση για μια αίτηση, ακολουθεί τις διαδικασίες που περιγράφονται στα παρακάτω τμήματα. Όταν όλες οι διαδικασίες που αφορούν την δημιουργία της αίτησης ολοκληρωθούν, τότε ο UAS δίνει την αίτηση πίσω στον server από τον οποίο πήρε την αίτηση.

#### **8.2.6.1 Στέλνοντας μια προσωρινή απάντηση**

Μια γενική μεθοδολογία παραγωγής απαντήσεων των UASs για αιτήματα εκτός από του τύπου INVITE είναι η παραγωγή τελικών απαντήσεων σε αυτού του τύπου τα αιτήματα όσο το δυνατόν γρηγορότερα.

Όταν μια απάντηση τύπου 100 (Προσπάθεια) παράγεται, κάθε πεδίο Timestamp στην αίτηση θα πρέπει να αντιγραφεί σε αυτήν την 100 απάντηση. Αν υπάρχει καθυστέρηση στην παραγωγή της απάντησης< ο UAS θα πρέπει να προσθέσει μια τιμή καθυστέρησης στο πεδίο Timestamp της απάντησης. Η τιμή πρέπει να περιέχει την διαφορά μεταξύ του χρόνου αποστολής της απάντησης και την παράδοση της αίτησης, σε δευτερόλεπτα.

### **8.2.6.2 Πεδία και Παράμετροι**

Το πεδίο «From» της απάντησης πρέπει να ταιριάζει με το πεδίο «From» της αίτησης. Το πεδίο Call-ID της απάντησης πρέπει να ταιριάζει με το πεδίο Call-ID της αίτησης. Το ίδιο ισχύει και για τα πεδία Cseq και Via.

Αν η αίτηση περιείχε μια παράμετρο στο πεδίο «To», τότε η απάντηση πρέπει να περιέχει την ίδια παράμετρο στο δικό της πεδίο «To». Όμως ακόμα και αν η αίτηση δεν είχε παράμετρο στο πεδίο «To», ο UAS πρέπει να προσθέσει παράμετρο στο πεδίο «To» της απάντησης. Αυτό συμβαίνει διότι αυτήν την παράμετρο χρησιμοποιεί ο UAS ως ταυτότητα σε αυτόν τον διάλογο. Άρα όπως είναι προφανές θα χρησιμοποιεί την ίδια παράμετρο και σε όλες τις απαντήσεις για αυτόν τον διάλογο. Να σημειωθεί ότι για τις απαντήσεις τύπου 100 η προσθήκη παραμέτρου είναι προαιρετική.

### **8.2.7 Συμπεριφορά UAS δίχως κατάσταση**

Ο UAS δίχως κατάσταση είναι ένας UAS οποίος δεν διατηρεί την κατάσταση συναλλαγής. Απαντά κανονικά σε κάθε αίτηση αλλά δεν διατηρεί τις αλλαγές στην κατάστασή του όπως κανονικά θα έπρεπε μετά την αποστολή της απάντησης. Αν ένας UAS δίχως κατάσταση λάβει ξανά μια ήδη αποδεχτή αίτηση αναπαράγει την απάντηση και την ξαναστέλνει σαν να την έστειλε πρώτη φορά. Ένας UAC δεν μπορεί να είναι δίχως κατάσταση, εκτός και αν ο τύπος επεξεργασίας της μεθόδου είχε ως αποτέλεσμα συνεχώς την ίδια απάντηση (για πανομοιότυπες αιτήσεις). UASs δίχως κατάσταση δεν χρησιμοποιούν στρώμα συναλλαγής αλλά δέχονται απευθείας τις αιτήσεις από το στρώμα μεταφοράς στο οποίο επίσης στέλνουν απευθείας τις απαντήσεις

Ο βασικότερος ρόλος των UAS δίχως κατάσταση είναι να διαχειρίζονται μη πιστοποιημένες αιτήσεις

Οι βασικότερες αρχές των UAS δίχως κατάσταση είναι οι παρακάτω:

1. Ένας UAS δίχως κατάσταση δεν πρέπει να στέλνει απαντήσεις τύπου 1XX
2. Ένας UAS δίχως κατάσταση δεν πρέπει να επαναστέλνει απαντήσεις
3. Ένας UAS δίχως κατάσταση πρέπει να αγνοεί ACK αιτήσεις
4. Ένας UAS δίχως κατάσταση πρέπει να αγνοεί αιτήσεις CANCEL

Σε όλες τις άλλες περιπτώσεις ένας UAS δίχως κατάσταση μπορεί να συμπεριφέρεται όπως ένας UAS που διατηρεί την κατάσταση του.

### **8.3 Redirect Servers (Servers επαναπροσανατολισμού)**

Σε κάποιες αρχιτεκτονικές μπορεί να είναι επιθυμητή η μείωση του επεξεργαζόμενου φορτίου στους Proxy Servers οι οποίοι είναι υπεύθυνοι για την διαδρομή των αιτήσεων βασισμένοι στον επαναπροσανατολισμό.

Ο επαναπροσανατολισμός επιτρέπει στους Servers να διοχετεύουν πληροφορίες διαδρομής μιας αίτησης μέσω προηγούμενων απαντήσεων που αφορούν μια αίτηση. Όταν ο αποστολέας μιας αίτησης λάβει τον επαναπροσανατολισμό, θα στείλει μια νέα αίτηση βασισμένη στην URI που έλαβε. Με την διάδοση των URI από τον πυρήνα του δικτύου, στις άκρες του, ο επαναπροσανατολισμός επιτρέπει την άμεση προσπέλαση του δικτύου.

Ένας server επαναπροσανατολισμού είναι λογικά δομημένος από ένα στρώμα συναλλαγής server και από ένα χρήστη συναλλαγής που έχει πρόσβαση στην υπηρεσία εντοπισμού. Αυτή η υπηρεσία εντοπισμού είναι μια βάση δεδομένων που περιέχει διαδρομές από μία URI σε μια άλλη ανάλογα με την θέση αυτής της URI.

Ένας server επαναπροσανατολισμού δεν μπορεί να παράγει SIP αιτήσεις από μόνος του. Όταν δέχεται αιτήσεις πέρα από αυτές του τύπου CANCEL, ο server είτε απορρίπτει την αίτηση ή συγκεντρώνει όλες τις πιθανές τοποθεσίες από την υπηρεσία εντοπισμού και επιστρέφει μια τελική απάντηση τύπου 3XX. Για αιτήσεις τύπου CANCEL ο server παράγει απαντήσεις τύπου 2XX. Αυτού του τύπου η απάντηση ουσιαστικά τελειώνει την SIP συναλλαγή. Ο server προσανατολισμού διατηρεί την κατάσταση συναλλαγής για όλη την διάρκεια της SIP συναλλαγής.

Όταν ένας server επαναπροσανατολισμού επιστρέφει μια απάντηση τύπου 3XX σε μία αίτηση, τότε παράγει μια λίστα με όλες τις πιθανές τοποθεσίες την οποία εισάγει στο πεδίο Contact

Το πεδίο Contact περιέχει URIs που προσδιορίζουν νέες τοποθεσίες, ονόματα χρήστη και επιπρόσθετες παραμέτρους μεταφοράς. Απαντήσεις τύπου 301 (μόνιμη μεταφορά) ή 302 (προσωρινή μεταφορά) μπορεί να περιέχουν τις αρχικές πληροφορίες τοποθεσίας που είχε και η αρχική αίτηση αλλά έχουν και επιπλέον πληροφορίες για πιθανές τοποθεσίες και παραμέτρους μεταφοράς.

Επίσης οι Servers επαναπροσανατολισμού δεν πρέπει να ξαναστέλνουν μία αίτηση σε URI η οποία είναι ίδια με αυτήν που περιέχεται στο πεδίο Request-URI.

Για παράδειγμα αν ένας χρήστης χρησιμοποιεί έναν proxy εκτός ορίων και αυτός ο proxy αναπροσανατολίζει αιτήσεις, τότε υπάρχει πιθανότητα να δημιουργηθούν ατελείωτες λούπες.

## **9 Ακύρωση Αίτησης**

Στο προηγούμενο κεφάλαιο μελετήσαμε την γενική συμπεριφορά των UA για την παραγωγή αιτήσεων και την επεξεργασία απαντήσεων για τις αιτήσεις κάθε τύπου για κάθε μέθοδο. Σε αυτό το κεφάλαιο θα μελετήσουμε μια γενικού τύπου μέθοδο που ονομάζεται Cancel (ακύρωση).

Η αίτηση Cancel (ακύρωση) χρησιμοποιείται για να ακυρώσει μια προηγούμενη αίτηση που στάλθηκε από έναν φορέα. Συγκεκριμένα ζητά από έναν UAS να σταματήσει την επεξεργασία της αίτησης και να παράγει μια απάντηση λάθους για αυτήν την αίτηση. Το Cancel (ακύρωση) δεν έχει καμία επίπτωση σε αιτήσεις για τις οποίες ο UAS έχει ήδη παράγει τελικές απαντήσεις. Για τον λόγο αυτό η αίτηση Cancel (ακύρωση) πρέπει να χρησιμοποιείται για αιτήσεις που ο UAS χρειάζεται αρκετό χρόνο να τις επεξεργαστεί. Έτσι συνήθως η αίτηση Cancel (ακύρωση) την χρησιμοποιούμε μόνο για να ακυρώσουμε αιτήσεις INVITE για τις οποίες ο Server χρειάζεται αρκετό χρόνο για να παράγει απάντηση. Για παράδειγμα αν ο κουδουνισμός είχε αρχίσει προτού ο UAS λάβει την αίτηση Cancel (ακύρωση) τότε μόλις την λάβει ο κουδουνισμός θα σταματήσει και ο server θα παράγει ένα μήνυμα σφάλματος

Οι αιτήσεις Cancel (ακύρωση) μπορούν να παραχθούν και αποσταλούν και από proxies και από φορείς χρηστών.



## **9.1 Συμπεριφορά φορέα**

Όπως ανάφερα και νωρίτερα τις αιτήσεις Cancel (ακύρωση) τις χρησιμοποιούμε μόνο για να διακόψουμε τις αιτήσεις INVITE διότι σε όλες τις υπόλοιπες αιτήσεις ο server σχεδόν απαντά άμεσα.

Οι παρακάτω διαδικασίες χρησιμοποιούνται για την κατασκευή μιας αίτησης Cancel (ακύρωση). Τα πεδία Request-URI, Call-ID, To, CSeq και From της αίτησης ακύρωσης πρέπει να είναι πανομοιότυπα με αυτά της αίτησης την οποία έχει σκοπό να ακυρώσει. Η αίτηση Cancel (ακύρωση) που δημιουργείται από κάποιον φορέα πρέπει να έχει μόνο μία τιμή του πεδίου Via της ολόιδια με αυτήν της πρώτης τιμής του πεδίου Via της αίτησης την οποία θέλει να ακυρώσει. Αυτό επιτρέπει στην αίτηση ακύρωσης να ταιριάζει με την αίτηση που προσπαθεί να ακυρώσει.

Επίσης αν η αίτηση που ακυρώνεται έχει τιμή στο πεδίο Route τότε και η αίτηση Cancel (ακύρωση) πρέπει να περιέχει ακριβώς το ίδιο πεδίο. Αυτό συμβαίνει ώστε οι proxies δίχως κατάσταση να μπορούν να στέλνουν εντολές Cancel κανονικά.

Η αίτηση Cancel δεν πρέπει να περιέχει τα πεδία Require και Proxy-Require. Από την στιγμή που η αίτηση δημιουργηθεί ο φορέας θα πρέπει να ελέγξει αν έχει δεχθεί κάποια απάντηση που να αφορά την αίτηση την οποία έχουμε ως σκοπό να ακυρώσουμε. Αν δεν έχει δεχθεί ήδη μια προσωρινή απάντηση που να αφορά την συγκεκριμένη αίτηση τότε η αίτηση Cancel (ακύρωση) δεν στέλνεται διότι ο φορέας πρώτα πρέπει να δεχτεί μια προσωρινή απάντηση για να στείλει την αίτηση Cancel (ακύρωση). Αν όμως ο φορέας έχει δεχτεί ήδη την τελική απάντηση που να αφορά την αρχική αίτηση τότε δεν πρέπει να στείλει καν την αίτηση Cancel (ακύρωση) μιας και αυτή δεν θα έχει κανένα αποτέλεσμα στην συγκεκριμένη περίπτωση. Όταν ο φορέας αποφασίσει να στείλει την αίτηση Cancel (ακύρωση) δημιουργεί πρώτα μια συναλλαγή φορέα για την αίτηση Cancel (ακύρωση) και την στέλνει μαζί με τις υπόλοιπες πληροφορίες συναλλαγής.

## **9.2 Συμπεριφορά Server**

Ουσιαστικά η αίτηση Cancel (ακύρωση) ζητά από ένα φορέα συναλλαγής να ακυρώσει μια συναλλαγή. Ο φορέας συναλλαγής καθορίζει το ποια συναλλαγή θα ακυρωθεί μέσω των πληροφοριών που του παρέχονται από την αίτηση Cancel (ακύρωση).

Η επεξεργασία της αίτησης Cancel (ακύρωση) από ένα server εξαρτάται από τον ίδιο τον server. Ένας proxy δίχως κατάσταση απλά θα την μεταβιβάσει, ένας proxy με κατάσταση μπορεί να απαντήσει ο ίδιος στην αίτηση και να δημιουργήσει μερικές δικές του αιτήσεις Cancel και ένας UAS απλά να απαντήσει σε αυτό το αίτημα.

Ο UAS πρώτα επεξεργάζεται την αίτηση CANCEL σύμφωνα με τον τύπο επεξεργασίας που περιγράψαμε στον τομέα 8.2. Αν ο UAS δεν εντοπίσει κατάλληλη συναλλαγή για την αίτηση Cancel (ακύρωση) τότε θα πρέπει να απαντήσει στην αίτηση Cancel (ακύρωση) με μια απάντηση 481 (η συναλλαγή δεν υφίσταται). Ωστόσο αν η συναλλαγή υφίσταται η συμπεριφορά του UAS στην αίτηση Cancel (ακύρωση) εξαρτάται από το αν έχει δεχτεί τελική απάντηση για την αίτηση που θέλουμε να ακυρώσουμε. Αν η τελική απάντηση έχει γίνει δεκτή από τον UAS τότε η αίτηση Cancel (ακύρωση) δεν έχει καμία επίδραση στην επεξεργασία της αρχικής αίτησης την οποία σκοπεύαμε να ακυρώσουμε. Αν ο UAS δεν έχει δεχτεί ακόμα τελική απάντηση τότε η συμπεριφορά του απέναντι στην αίτηση Cancel (ακύρωση) εξαρτάται από την μέθοδο στην οποία ανήκει η αίτηση προς ακύρωση. Αν η αρχική αίτηση ήταν η INVITE τότε ο UAS πρέπει να απαντήσει αμέσως στην αίτηση με ένα μήνυμα 487 (Τερματισμός αίτησης). Όπως προανέφερα η αίτηση Cancel (ακύρωση) δεν έχει καμία επίπτωση σε αιτήσεις που ανήκουν σε άλλη μέθοδο εκτός από την INVITE.

## **10. Καταγραφή**

### **10.1 Εισαγωγή στην καταγραφή**

Το τελικό κεφάλαιο με το οποίο θα ασχοληθούμε είναι η καταγραφή. Το SIP προσφέρει δυνατότητες εντοπισμού χρήστη. Αυτό σημαίνει ότι όταν ένας χρήστης επιθυμεί να εγκαταστήσει επικοινωνία με κάποιον άλλο το SIP είναι σε θέση να εντοπίσει τον προορισμό στον οποίο ο άλλος χρήστης είναι προσβάσιμος. Αυτό επιτυγχάνεται με την υπηρεσία εντοπισμού η οποία περιέχει καταγεγραμμένο κάθε χρήστη όπως ανέφερα και σε προηγούμενο κεφάλαιο.

Η καταγραφή δεσμεύει μια περιοχή της υπηρεσίας εντοπισμού για ένα συγκεκριμένο τομέα και συσχετίζει μια διεύθυνση εγγραφής URI με μία ή περισσότερες διευθύνσεις επαφής. Έτσι όταν ένας proxy αυτού του τομέα λάβει μία αίτηση της οποίας η request-URI ταιριάζει με την

διεύθυνση εγγραφής, τότε ο proxy θα στείλει την αίτηση σε κάποια διεύθυνση επαφής που είναι εγγεγραμμένη σε αυτή την διεύθυνση εγγραφής.

Η καταγραφή γίνεται με την αποστολή μιας αίτησης REGISTER σε έναν ειδικό UAS γνωστό ως registrar. Αυτός διαχειρίζεται τις αιτήσεις REGISTER για την υπηρεσία εντοπισμού. Αυτήν την υπηρεσία εντοπισμού συμβουλεύονται οι proxies που είναι υπεύθυνοι για την προώθηση αιτήσεων στον συγκεκριμένο τομέα.

## **10.2 Δομή της αίτησης Καταγραφής**

Η αίτηση REGISTER (καταγραφή) χρησιμοποιείται για την διαμόρφωση της συσχέτισης μεταξύ της διεύθυνσης εγγραφής URI και των διευθύνσεων επαφής. Μπορεί να διαγραφεί να προθέτει και να αλλάζει διευθύνσεις επαφής που σχετίζονται με μία διεύθυνση εγγραφής ή και το αντίστροφο.

Τα απαραίτητα πεδία μιας αίτησης REGISTER (καταγραφή) είναι τα παρακάτω:

1.Request-URI: Το πεδίο αυτό κατονομάζει τον τομέα της υπηρεσίας εντοπισμού για την οποία γίνεται η καταγραφή. Για παράδειγμα "sip:Athens.com". Τα στοιχεία «userinfo» και «@» σεν πρέπει να υπάρχουν.

2.To: Το πεδίο αυτό περιέχει την διεύθυνση εγγραφής για την οποία δημιουργείται, σβήνεται ή μεταβάλλεται η καταγραφή. Η διεύθυνση εγγραφής πρέπει να είναι SIP URI ή SIPS URI.

3.From: Το πεδίο αυτό περιέχει την διεύθυνση εγγραφής για το πρόσωπο για το οποίο γίνεται η καταγραφή. Η τιμή του είναι ίδια με αυτή του πεδίου To εκτός και αν την καταγραφή την κάνει τρίτο άτομο.

4.Call-ID: Όλες οι καταγραφές από έναν UAC θα πρέπει να χρησιμοποιούν το ίδιο πεδίο Call-ID για καταγραφές που προορίζονται για ένα συγκεκριμένο καταγραφέα.

5.Cseq: Η τιμή αυτού του πεδίου καθορίζει την σωστή σειρά των αιτήσεων καταγραφής. Ο UA θα πρέπει να αυξάνει κατά ένα την τιμή αυτού του πεδίου για κάθε καταγραφή με το ίδιο Call-ID πεδίο.

6.Contact: Προαιρετικό πεδίο

### **10.2.1 Εισαγωγή διευθύνσεων επαφής**

Μια αίτηση REGISTER (καταγραφή) μπορεί να χρησιμοποιείται για πρόσθεση διευθύνσεων επαφής στις οποίες θα στέλνονται αιτήσεις που σχετίζονται με την διεύθυνση εγγραφής. Σε αυτήν την περίπτωση το πεδίο Contact της αίτησης REGISTER (καταγραφή) περιέχει SIP ή SIPS URIs που αναφέρονται σε τερματικά σημεία και είναι οι διευθύνσεις επαφής (για παράδειγμα : “sip:Μαρία@cube2214a.athens.com” ενώ η διεύθυνση εγγραφής της Μαρίας είναι “sip:Μαρία@athens.com”).

Υπάρχει η δυνατότητα να ορίσουμε μέσω της αίτησης REGISTER (καταγραφή) το χρονικό διάστημα για το οποίο αυτή η καταγραφή θα ισχύει. Αυτό γίνεται με 2 τρόπους. Είτε μέσω του πεδίου EXPIRE (Λήξης) το οποίο προσθέτουμε στην αίτηση είτε προσθέτοντας μια παράμετρο λήξης στο πεδίο Contact. Και στις 2 περιπτώσεις χρησιμοποιούνται παράμετροι λήξης οι οποίοι μόλις γίνουν «0» η καταγραφή παύει να ισχύει.

### **10.2.2 Διαγραφή διευθύνσεων επαφής**

Όπως ανέφερα και παραπάνω υπάρχει η δυνατότητα να διαγραφούν κάποιες διευθύνσεις επαφής. Αυτό γίνεται με το να επεμβαίνουμε στην παράμετρο λήξης μιας καταγραφής. Συνήθως με την επέμβαση κάποιου UA γίνεται δυνατό να θέσουμε την τιμή μιας παραμέτρου λήξης στο «0» με αποτέλεσμα να διαγράφεται η καταγραφή στην οποία αντιστοιχεί και ταυτόχρονα να διαγράφεται και η διεύθυνση επαφής την οποία εισήγαμε με την συγκεκριμένη καταγραφή.

## **10.3 Επεξεργασία της αίτησης REGISTER**

Σε αυτό το τμήμα το οποίο είναι και το τελευταίο αυτής της μελέτης είναι το πώς γίνεται η επεξεργασία των αιτήσεων καταγραφής από τους registrars.

Ο registrar διαχειρίζεται αιτήσεις όπως περιγράψαμε στο κεφάλαιο 8 με την διαφορά ότι αυτός επεξεργάζεται μόνο αιτήσεις τύπου REGISTER. Για τον λόγο αυτό δεν μπορεί να παράγει απαντήσεις τύπου 6XX.

Σημαντικό επίσης είναι ο registrar να γνωρίζει τους τομείς (domain) για τα οποία διαχειρίζεται τέτοιες αιτήσεις και καταγράφει διευθύνσεις, καθώς και να επεξεργάζεται τις αιτήσεις REGISTER με την σειρά που τις λαμβάνει. Επίσης θα πρέπει να τις επεξεργάζεται κάθε μία ξεχωριστά και ατομικά χωρίς να συνδυάζει στοιχεία από άλλες καταγραφές.

Όταν ένας registrar δέχεται μία αίτηση καταγραφής ακολουθεί τα παρακάτω βήματα.

1) Πρώτα εξετάζει την αίτηση για να δει αν έχει πρόσβαση σε καταγραφές που αφορούν τον τομέα που φαίνεται στο πεδίο Request-URI της αίτησης. Αν όχι τότε ο registrar λειτουργεί σαν proxy και στέλνει την αίτηση στον τομέα που την αφορά.

2) Ο registrar εξετάζει την αίτηση για ενδεχόμενες μη αναγνωρίσιμες επεκτάσεις όπως είχαμε αναφέρει και στο κεφάλαιο 8.

3) Ο registrar μετά αναγνωρίζει τον UAC. Αν δεν υπάρχει μηχανισμός αναγνώρισης από το δίκτυο τότε ο registrar χρησιμοποιεί την διεύθυνση από το πεδίο From ως την ταυτότητα του χρήστη.

4) Μετά ο registrar ανιχνεύει για το αν ο συγκεκριμένος χρήστης έχει εξουσιοδότηση για να μεταβάλλει καταγραφές που αφορούν την συγκεκριμένη διεύθυνση εγγραφής. Για αυτό το λόγο συμβουλευεται μια βάση δεδομένων που αντιστοιχεί ονόματα με διευθύνσεις εγγραφής. Αν ο χρήστης δεν έχει εξουσιοδότηση τότε ο registrar του στέλνει το μήνυμα 403 (Απαγορευμένη ενέργεια)

5) Μετά ο registrar εξάγει την διεύθυνση εγγραφής από το πεδίο "To" της αίτησης. Αν δεν βρεθεί η διεύθυνση τότε στέλνεται το μήνυμα 404 (Δεν βρέθηκε) και τα επόμενα βήματα παραλείπονται. Η URI θα πρέπει να μετατραπεί στην κανονική της μορφή και για να γίνει αυτό παραλείπονται όλοι οι URI παράμετροι.

6) Ο registrar ελέγχει για το αν η αίτηση περιέχει το πεδίο Contact. Αν όχι πηγαίνει στο τελικό στάδιο. Αν ναι κοιτά το αν το πεδίο έχει παράμετρο λήξης και ποια είναι αυτή.

7) Μετά επεξεργάζεται κάθε διεύθυνση επαφής στο πεδίο Contact. Για κάθε διεύθυνση καθορίζει και χρόνο λήξης σύμφωνα με την

παράμετρο λήξης για κάθε διεύθυνση. Αν κάποια από αυτές δεν έχει παράμετρο λήξης τότε ορίζεται μια προεπιλεγμένη.

8) Ο registrar επιστρέφει το μήνυμα 200 (OK).

## **ΕΠΙΛΟΓΟΣ**

Στην παραπάνω μελέτη είδαμε τις σημαντικότερες λειτουργίες και αρχές δομής του πρωτοκόλλου SIP. Αναλύσαμε τις αρχές δομής του και παρουσιάσαμε τον τρόπο με τον οποίο το πρωτόκολλο χρησιμοποιείται για την εγκατάσταση και διαχείριση μιας συνδιάλεξης. Επίσης αναλύσαμε τα πλεονεκτήματα που καθιστούν το SIP ως ένα συνεχώς αναπτυσσόμενο πρωτόκολλο που αποκτά συνεχώς και περισσότερο έδαφος απέναντι στα νέα πρωτόκολλα. Ταυτόχρονα οι δυνατότητες που παρέχει καθώς και η υποστήριξη που του παρέχεται, το καθιστούν μαζί με το H.323 ένα από τα ισχυρότερα πρωτόκολλα και εργαλεία στις σύγχρονες VoIP εφαρμογές. Σαφώς και η μεγάλη υποστήριξη που έχει από την σύγχρονη βιομηχανία βοήθησε το SIP να γίνει ένα συνεχώς εξελισσόμενο πρωτόκολλο που ολοένα αυξάνει την πληθώρα των δυνατοτήτων του και την ευχρηστία του. Το σημαντικότερο από τα πλεονεκτήματά του είναι η συμβατότητα του με άλλα πρωτόκολλα καθώς και η ευελιξία του στην επεξεργασία και στην εγκατάσταση συνδιαλέξεων.

### **Βιβλιογραφία**

<http://www.packetizer.com/>

[http://www.sipforum.org/component/option,com\\_docman/task,doc\\_view/gid,16/Itemid,75](http://www.sipforum.org/component/option,com_docman/task,doc_view/gid,16/Itemid,75)

