ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ

Τμήμα Διδακτικής της Τεχνολογίας και Ψηφιακών Συστημάτων

Πρόγραμμα Μεταπτυχιακών Σπουδών με κατεύθυνση τις Ψηφιακές Επικοινωνίες και Δίκτυα

'VOIP (Voice Over IP) - transportation and signalling of voice communications over ip networks – implementation using Asterisk'

Μάνθα Ματθαίου Παπουτσή

Επιβλέπων: κ. Παναγιώτης Δεμέστιχας

ΠΕΡΙΕΧΟΜΕΝΑ

VOICE OVER INTERNET PROTOCOL OVERVIEW 5

1.1 Εισαγωγή 5
1.2 Ιστορική Αναδρομή 5
1.3 Πλεονεκτήματα της τεχνολογίας VOIP 7
1.4 Μειονεκτήματα της τεχνολογίας VOIP 10
1.5 Η μετεξέλιξη της μεταγωγής κυκλώματος σε μεταγωγή πακέτου (PSTN vs VOIP) 11
1.6 Εφαρμογές ενός VOIP δικτύου 15

2ο ΚΕΦΑΛΑΙΟ 18

ΒΑΣΙΚΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ – ΛΕΙΤΟΥΡΓΙΕΣ ΕΝΟΣ VOIP ΔΙΚΤΥΟΥ 18

2.1Υπηρεσίες βάσεων δεδομένων 18
2.1.1 ENUM SERVER 19
2.1.1.1 ΠΑΡΑΔΕΙΓΜΑΤΑ 21
2.1.1.1.a Μετατροπή ενός E.164 αριθμού σε ένα URI 21
2.1.1.1.b Εγκατάσταση και τερματισμός μιας κλήσης χρησιμοποώντας τον ENUM 21
2.1.2 Αριθμοδότηση E.164 22
2.1.3 Χαρακτηριστικά ποιότητας 25
2.1.3.1 Καθυστέρηση 25
2.1.3.2 Ηχώ 29
2.1.3.3 Απώλεια πακέτων 30
2.1.4 Διαδικασίες Κωδικοποίησης-Αποκωδικοποίησης (CODEC) 30

3ο ΚΕΦΑΛΑΙΟ 32

Session Initiation Protocol (SIP) 32

3.1 Εισαγωγή 32
3.2 Δομή του πρωτοκόλλου 34
3.3 Βασικά συστατικά ενός SIP δικτύου 36
3.3.1 User Agents 36
3.3.2 SIP Proxy Server 37

3.3.2.1 Stateless Proxy Server 38
3.3.2.1 Stateful Proxy Server 38
3.3.3 SIP Registrar Server 39
3.3.4 Location Server 40
3.3.5 Application Server 40
3.3.6 SIP Redirect Server 40
3.4 SIP Messages 42
3.4.1 SIP Addressing 43
3.4.2 SIP Message Structure 44
3.4.2.1 Request Message Format 44
3.4.2.2 Response Message Format 46
3.4.2.3 Πεδία επικεφαλίδας (header fields) 48
3.4.2.4 Σημαντικότεροι headers ενός SIP μηνύματος 49
3.4.3 SIP Requests 52
3.4.4 SIP Responses 55
3.4.4.1 1XX Response Class (Ένα request έχει ληφθεί και επεξεργάζεται) 55
3.4.4.2 2XX Response Class (Το request έχει ληφθεί επιτυχώς από τον UAS) 55
3.4.4.3 Redirection (3xx): Επιπλέον προσπάθεια πρέπει να καταβληθεί (τυπικά από το αποστολέα) για να ολοκληρωθεί το request 56
3.4.4.4 Client Error (4xx): Το request περιέχει συντακτικό λάθος ή δε μπορεί να ολοκληρωθεί στον παρόν Server. 56
3.4.4.5 Server Error (5xx): Ο Server απέτυχε να επεξεργαστεί ένα πιθανότατα valid request. 58
3.4.4.6 Global Failure (6xx): Το request δε μπορεί να επεξεργαστεί από κανένα server. 58
3.5 SIP Transactions 59
3.5.1 INVITE Client Transactions 60
3.5.2 Non - INVITE Client Transactions 61
3.5.3 Αντιστοίχιση ενός Response σε ένα Client Transaction 62
3.5.4 INVITE Server Transactions 63
3.5.5 Non INVITE Server Transactions 64
3.5.6 Αντιστοίχιση ενός Request σε ένα Server Transaction 65

3.6 SIP Dialogs 66
3.6.1 Ta Dialogs διευκολύνουν τη δρομολόγηση των μηνυμάτων (Routing) 68
3.7 Το Πρωτόκολλο SDP 69
3.7.1 Διαδικασία του SDP Negotiation 74
3.8 Πραγματοποίηση Κλήσεων με το SIP πρωτόκολλο 76
3.8.1 SIP Registration Process 76
3.8.2 SIP Call Processing 79
3.9 Συμπληρωματικά Πρωτόκολλα 91
3.9.1 MGCP (media gateway control protocol) 92
3.9.2 Real time transport protocol (RTP) & real time transport control protocol (RTCP) 94

4ο ΚΕΦΑΛΑΙΟ 99

Εργαστηριακή Άσκηση πάνω σε ένα απλό VOIP δίκτυο 99

4.1 Εισαγωγή 99
4.2 Αρχιτεκτονική δομή της άσκησης 100
4.2.1 User Agents 100
4.2.2 Asterisk 101
4.3 Προεργασία εργαστηριακής Άσκησης 101
4.4 Εργαστηριακή Άσκηση 125
4.4.1 Γενικά χαρακτηριστικά του πρωτοκόλλου SIP 125
4.4.2 Call Proccessing 126
4.4.2.1 Registration Proccess 126
4.4.2.2 Call Proccessing 128
4.4.2.3 RTP Stream 134

1ο ΚΕΦΑΛΑΙΟ

VOICE OVER INTERNET PROTOCOL OVERVIEW

1.1 Εισαγωγή

Το Voice over IP, γνωστό και ως VoIP, Τηλεφωνία IP και Τηλεφωνία μέσω διαδικτύου, είναι μια τεχνολογία που καθιστά δυνατή τη δρομολόγηση φωνητικών συνδιαλέξεων μέσω του διαδικτύου. Ουσιαστικά, η τεχνολογιά VOIP μετατρέπει τα αναλογικά σήματα φωνής σε ψηφιακά, τα οποία οργανώνονται σε πακέτα έτοιμα για αποστολή μέσω του IP-based δικτύου. Η αλματώδης ανάπτυξη του internet και η εξάπλωση των δικτύων τα οποία στηρίζονται στο Internet Protocol (IP) έκαναν την ανάπτυξη αυτής της τεχνολογίας βιώσιμη τα τελευταία χρόνια. Πλήθος υπηρεσιών όπως phone to phone επικοινωνίες, PC to phone, phone to PC, IP Phone, toll free number, call center applications, Unified messaging, IN applications using SS7 κτλ είναι μερικά ενδεικτικά παραδείγματα της τεχνολογίας VOIP σήμερα. Ένα από τα κυριότερα πλεονεκτήματα του VOIP (συγκριτικά με το παραδοσιακό δημόσιο τηλεφωνικό δικτύο (PSTN)) είναι ότι δίνει τη δυνατότητα υπεραστικών κλήσεων παρακάμπτοντας την επιβάρυνση επιπλέον φόρου. Επίσης ας σκεφτεί κανείς ότι το IP δίκτυο μπορεί να μεταφέρει 5 έως 10 φορές τον αριθμό φωνητικών κλήσεων πάνω από το ίδιο bandwidth σε σύγκριση με το PSTN. Το VOIP μπορεί να εξασφαλίσει στις εταιρείες παροχής τηλεπικοινωνιών ένα ενιαίο IP δίκτυο, το οποίο θα υποστηρίζει εκτός από υπηρεσίες δεδομένων και υπηρεσίες φωνής. Αυτή η τεχνολογική πρόοδος έχει αρχίσει να επηρεάζει όχι μόνο τη παραδοσιακή βιομηχανία τηλεπικοινωνιών, αλλά θα αλλάξει ριζικά και τη κοστολόγηση της παραδοσιακής τηλεφωνίας, γεγονός που μελλοντικά θα εξασφαλίσει και την βιωσιμότητα της.

1.2 Ιστορική Αναδρομή

Η ιστορία του VOIP ξεκινάει το 1964 όταν δημοσιεύτηκε η πρώτη έρευνα η οποία αφορούσε τη μετάδοση φωνής υπό μορφή πακέτων από τον Paul Baran. Ωστόσο τα πρώτα δείγματα της τεχνολογίας αυτής οριοθετούνται γύρω στο 1995 όταν κάποιοι ερασιτέχνες από το Ισραήλ άρχισαν να δρομολογούν πακέτα φωνής μέσω internet,

εξασφαλίζοντας pc to pc επικοινωνία. Αυτό επέτρεπε σε pc χρήστες να κάνουν υπεραστικές κλήσεις χωρίς επιπλέον χρεώσεις. Αργότερα, την ίδια χρονιά αναπτύχθηκε και το πρώτο λογισμικό για τηλεφωνία μέσω internet από την Vocaltec Inc. Αν και σήμερα η τεχνολογία VOIP χρησιμοιποιεί τυποποιημένες συσκευές τηλεφώνων, οι αρχικοί χρήστες έπερεπε να διαθέτουν υπολογιστές με το ίδιο λογισμικό και το απαιτούμενο hardware (μικρόφωνο, speakers, κάρτα ήχου). Σκοπός του συγκεκριμένου λογισμικού ήταν να συμπιέσει το σήμα φωνής, να το μετατρέψει σε πακέτα και να τα δρομολογήσει μέσω του internet. Αν και η ποιότητα συνδιαλλέξεων των πρώτων IP τηλεφώνων ήταν χαμηλή, έδωσαν τα πρώτα ελπιδοφόρα μηνύματα για την ανάπτυξη μιας νέας τεχολογίας. Συμπτωματικά την ίδια χρονιά η Διεθνής Ένωση τηλεπικοινωνιών (ITU) άρχισε να εργάζετε συστηματικά για την ανάπτυξη του προτύπου H.323.

Το VOIP εξελίχτηκε βαθμιαία τα επόμενα χρόνια και το 1998 εμφανίστηκαν οι πρώτες εταιρείες που παρείχαν PC to phone υπηρεσίες. Οι Phone to phone υπηρεσίες ακολούθησαν, αν και συχνά ήταν απαραίτητη η χρήση ενός υπολογιστή για την εγκατάσταση της σύνδεσης. Όπως οι περισσότερες εφαρμογές internet στα τέλη της δεκατίας του 90, έτσι και οι πρώτες υπηρεσίες VOIP βασίστηκαν κυρίως σε διαφημίσεις (οι οποίες ακούγονταν κάθε φορά πριν την έναρξη της συνδιάλλεξης) για την επιβίωση τους. Η βαθμιαία ανάπτυξη της ευρυζωνικής υπηρεσίας Ethernet βοήθησε την ποιότητα των κλήσεων (μειώνοντας το latency), αν και η δυσκολία τερματισμού των κλήσεων από το IP δίκτυο προς το κλασικό δημόσιο δίκτυο εξακολοθούσε να αποτελεί ανασταλτικό παράγοντα.

Η επανάσταση στην ιστορία του VOIP ήρθε όταν κατασκευαστές hardware, όπως η Cisco Systems και η Nortel, άρχισαν να κατασκευάζουν συσκευές που έκαναν εφικτή τη δυνατότητα switching μεταξύ VOIP και PSTN. Επομένως η μεταγωγή των πακέτων φωνής σε «κάτι» που θα ήταν αναγνωρίσιμο από το PSTN (και αντίστροφα) και που μέχρι τώρα μπορούσε να γίνει μόνο μέσω ενός ηλεκτρονικού υπολογιστή, μπορούσε πλέον να πραγματοποιηθεί από μία άλλη συσκευή κάνοντας το VOIP hardware λιγότερο εξαρτώμενο από ένα pc. Όσο το κόστος των συσκευών αυτών γινόταν όλο και πιο προσιτό, όλο και περισσότερες μεγάλες εταιρείες άρχισαν να ενσωματώνουν το VOIP στο δίκτυο τους.

Το Σεπτέμβριο του 1999 η IETF άρχισε να εργάζετε με το δημοφιλές πλέον πρωτόκολλο SIP (Session Initiation Protocol). Το SIP αναγνωρίστηκε σαν 3GPP (Third Generation Partnership Project) πρωτόκολλο σηματοδοσίας το Νοέμβριο του 2000. Το ενδιαφέρον των εταιρείων για ανάπτυξη τόσο λογισμικού όσο και υλικού για την αγορά VOIP (hard phones – network switches), σε συνδυασμό με την επέκταση του broadband, έφεραν σαν αποτέλεσμα το 2000 το 3% των κλήσεων στην Αμερική να πραγματοποιείτε μέσω της τεχνολογίας VOIP.

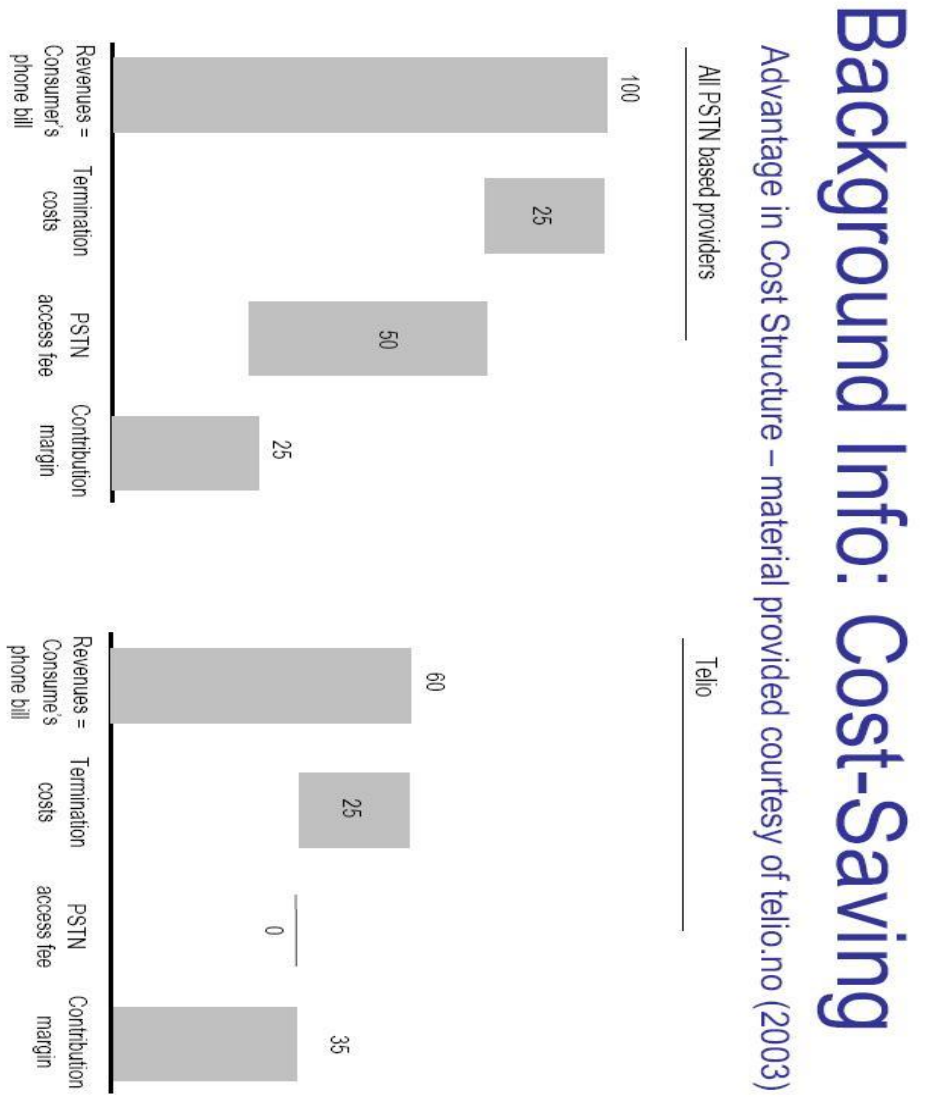
Έτσι το 2001 η Microsoft ενσωμάτωσε την τεχνολογία VOIP στο Windows XP Messenger χρησιμοποιώντας το SIP. Ακολούθησε η ίδρυση της Vonage, η οποία αποτέλεσε ένα από τους κύριους παρόχους στην ευρυζωνική τηλεφωνία και φυσικά το Skype το οποίο έχει σήμερα πάνω από 400 εκατομμύρια χρήστες.

1.3 Πλεονεκτήματα της τεχνολογίας VOIP

Πολλά είναι τα πλεονεκτήματα που θα ωθούσαν μια εταιρεία ή ένα οργανισμό να ενσωματώσει την τεχνολογία VOIP στις καθημερινές του ανάγκες, αν και αδιαμφισβήτητα το κόστος αποτελεί το δημοφιλέστερο. Και σε αυτό το σημείο θα αναρωτηθεί κανείς γιατί το VOIP θεωρείτε οικονομικά πιο προσιτό από το παραδοσιακό δημόσιο τηλεφωνικό δίκτυο. Μερικοί βασικοί λόγοι είναι οι παρακάτω:

- Μειωμένο κόστος τηλεφωνικών κλήσεων:

Το κόστος των τηλεφωνικών κλήσεων μέσω VOIP είναι μηδαμινό συγκριτικά με τις χρεώσεις του κλασικού τηλεφωνικού δικτύου. Αυτό οφείλετε κυρίως στο γεγονός ότι το VOIP χρησιμοποιεί την τεχνολογία internet, η οποία καθιστά εφικτή την επικοινωνία δύο απομακρυσμένων χρηστών (endpoints), με κόστος μιας απλής τηλεφωνικής dial up σύνδεσης στο internet, παρακάμπτοντας με αυτόν τον τρόπο το δημόσιο τηλεφωνικό δίκτυο και τις χρεώσεις του για κλήσεις σε μεγάλες αποστάσεις. Στο παρακάτω σχεδιάγραμμα φαίνετε καθαρά πόσο μπορεί να μειωθεί το κόστος του τελικού χρήστη με τη χρησιμοποίηση της τεχνολογίας VOIP.



Σχήμα 1: Το Cost revenue το οποίο προκύπτει από τη χρήση της τεχνολογίας VOIP

- Μειωμένο κόστος συντήρησης:

Το VOIP είναι μια τεχνολογία η οποία είναι βασισμένη κατά κύριο λόγο στο λογισμικό και κατά δεύτερον στο υλικό, γεγονός που καθιστά τη συντήρηση και τη βελτίωση της ακόμα πιο εύκολη. Παραδείγματος χάριν, η χρησιμοποίηση της τεχνολογίας VOIP από μια επιχείρηση είναι οικονομικά πιο συμφέρουσα από τη δέσμευση ενός Private Branch Exchange (PBX).

- Απλουστευμένη υποδομή:

Η λειτουργία και η διαχείρηση του δικτύου είναι απλουστευμένη μιας και χρησιμοποιείτε η ίδια υποδομή για τη μετάδοση φωνής και δεδομένων, χωρίς όμως να αυξάνεται η πολυπλοκότητα του backbone δικτύου. Επομένως η IP τηλεφωνία επιτρέπει τη χρήση τόσο του Internet όσο και των κλειστών εταιρικών δικτύων δεδομένων (intranet) για φωνητικές συνδιαλέξεις, ενοποιώντας με αυτό το τρόπο ενδοεταιρικές και εξωτερικές κλήσεις.

Όπως ήδη αναφέραμε η χρήση υπηρεσιών IP τηλεφωνίας έχουν ως άμεσο αποτέλεσμα την εξοικονόμηση οικονομικών και διαχειριστικών πόρων. Ωστόσο πλήθος αναλύσεων καταλήγουν στο συμπέρασμα οτί το χαμηλό κόστος δεν είναι το κυριότερο από τα πλεονεκτήματα ενος VOIP δικτύου. Παρακάτω ακολουθούν κάποια από αυτά:

- Μεγαλύτερη αξιοπιστία:

Τα IP δίκτυα προσφέρουν μεγαλύτερη αξιοπιστία σε σχέση με τα δίκτυα μεταγωγής κυκλώματος (PSTN), επειδή έχουν τη δυνατότητα να επαναδρομολογούν πακέτα που δεν έφθασαν στον προορισμό τους, λόγω διαφόρων προβλημάτων (βλάβη στο router ή στις γραμμές μετάδοσης).

- Αυξημένη Ευελιξία δυνατοτήτων και εφαρμογών:

Το VOIP εκτός από τη τις υπηρεσίες τηλεφωνίας παρέχει επιπλέον υπηρεσίες όπως video conferencing, electronic white boarding και multimedia/multiservice εφαρμογές. Η τεχνολογία VOIP διαχειρίζεται τη φωνή σαν οποιοδήποτε άλλο δεδομένο, επομένος ο χρήστης κατά τη διάρκεια της συνδιάλλεξης μπορεί να στένλει συννημένα αρχεία ή να συμμετέχει σε virtual meetings με shared data και videoconferencing. Τα συστήματα VOIP είναι αναπτυξιακά ευέλικτικα για αυτό και πλήθος εφαρμογών δίνουν ακόμα και δυνατότητες “Click to Dial” οι οποίες επιτρέπουν στο χρήστη να ξεκινήσει μια κλήση χτυπώντας ένα URL σε ένα browser.

- Location Independence:

Το VOIP επιτρέπει σε κάθε χρήστη να δέχεται εισερχόμενες κλήσεις οι οποίες θα δρομολογούνται στο γραφείο του και γενικότερα στο προσωπικό του VOIP νούμερο ανεξαρτήτου τοποθεσίας. Αυτό οφείλετε στο γεγονός ότι ένας VOIP χρήστης το μόνο που έχει να κάνει είναι να γίνει register σε ένα συγκεκριμένο VOIP server μέσω του οποίου μπορεί να πραγματοποιεί συνδιαλέξεις με άλλους VOIP χρήστες ή χρήστες άλλων δικτύων.

1.4 Μειονεκτήματα της τεχνολογίας VOIP

Η τεχνολογία VOIP παρουσιάζει και ένα πλήθος μειονεκτημάτων, τα οποία ένας provider ή ένας απλός χρήστης μπορεί να αξιολογήσει ανάλογα πάντα με το ποια χρήση της τεχνολογίας VOIP θα επικαλεστεί. Παρακάτω ακολουθούν κάποια από αυτά:

- **Απώλεια ποιότητας φωνής** : Οι τεχνικοί γνωρίζουν πως τα δίκτυα δεδομένων είναι πολύ διαφορετικά από τα δίκτυα φωνής. Στα δίκτυα δεδομένων, και κυρίως κυρίως στα δίκτυα Ethernet τα οποία κυριαρχούν στα εταιρικά υπολογιστικά περιβάλλοντα, τα πακέτα δρομολογούνται αόριστα και ακαθόριστα. Επομένως μπορούν να συγκρουσθούν και να καταστραφούν ακόμα και να χαθούν. Τόσο οι μηχανισμοί διόρθωσης σφάλματος σε Ethernet εξοπλισμό, όσο και το ίδιο το IP πρωτόκολλο μπορούν εύκολα να αντισταθμίσουν την απώλεια των δεδομένων. Όμως η απώλεια πακέτων μπορεί να έχει σημαντικές επιπτώσεις στις κλήσεις φωνής, που απαιτούν καλή ποιότητα και πραγματικού χρόνου ροή πακέτων από τη μία άκρη του δικτύου στην άλλη. Και ενώ ο ανθρώπινος νους μπορεί να κατανοήσει την ανθρώπινη ομιλία ακόμα και όταν υπάρχει μεγάλη παραμόρφωση, οι χρήστες έχουν εξοικειωθεί σε ένα συγκεκριμένο επίπεδο ποιότητας κλήσεως.
- **Απώλεια αξιοπιστίας** : Τα δίκτυα δεδομένων δεν είναι ακόμα τόσο αξιόπιστα όσο τα δίκτυα φωνής. Όλοι έχουμε ακούσει τις φράσεις : «πάγωσε ο υπολογιστής ή το δίκτυο είναι κάτω». Αυτό όμως σπάνια συμβαίνει στο παραδοσιακό δημόσιο δίκτυο. Η άμεση χρήση των τηλεφωνικών συσκευών του παραδοσιακού δημόσιου δικτύου και η αστάθεια του συνδέσεων internet είναι μερικοί από τους βασικότερους λόγους που λίγοι χρήστες θέλουν να αντικαταστήσουν τη σταθερή με τη VOIP τηλεφωνία παρά τα μεγάλα οικονομικά οφέλη που παρέχει.

- **Ταχεία εξέλιξη της τεχνολογίας** : Ο αλματώδης ρυθμός ανάπτυξης στην τεχνολογία των υπολογιστών, και των επικοινωνιών γενικότερα, κάνει σήμερα τους επιχειρηματίες αρκετά διστακτικούς στην αγορά νέου εξοπλισμού. Δύο είναι οι κύριοι λόγοι που συμβάλλουν στην αναποφασιστικότητα των επιχειρηματιών για την επένδυση σε μια νέα υπηρεσία. Ένας από αυτούς αφορά την πιθανότητα ότι μια άλλη καλύτερη λύση συγκριτικά με το VoIP θα έρθει σύντομα στην αγορά. Εάν η επένδυση σε αυτό το προϊόν είναι σημαντική από οικονομικής απόψεως, είναι άσκοπο να αχρηστευτεί και να επιλεχτεί μια μεταγενέστερη καλύτερη λύση. Μια άλλη ανησυχία για τους υπεύθυνους της εκάστωτε επιχείρησης αποτελεί το γεγονός ότι η επιλογή ενός προϊόντος που οδηγεί στη συνύπαρξη φωνής-δεδομένων μπορεί να οδηγήσει σε μια συμφωνία πέρα από την ίδια τη VoIP λύση, που οδηγεί σε μια μακροπρόθεσμη δέσμευση της εταιρείας σε μια συγκεκριμένη αρχιτεκτονική. Αυτή η ανησυχία επιδεινώνεται από την έλλειψη καθαρών προτύπων στην αγορά του VoIP. Στην απουσία τέτοιων προτύπων οι αρμόδιοι για τον εξοπλισμό μιας εταιρείας στηρίζουν την ανησυχία τους για τη δέσμευση των εταιριών κάτω από ένα συγκεκριμένο αρχιτεκτονικό πρότυπο.
- **Έλλειψη εμπειρίας και πείρας** : Η τεχνολογία VoIP είναι σχετικά καινούρια. Κάθε νέα υπηρεσία πρέπει να περάσει ένα πειραματικό στάδιο και έπειτα να κυριαρχήσει στην αγορά. Αυτό όμως παίρνει αρκετό χρόνο. Χωρίς την ανάλογη εμπειρία και προσεκτικό σχεδιασμό η τεχνολογία μπορεί απλά να μη προχωρήσει στο αγοραστικό κοινό.

1.5 Η μετεξέλιξη της μεταγωγής κυκλώματος σε μεταγωγή πακέτου (PSTN vs VOIP)

Το Δημόσιο Τηλεφωνικό Δίκτυο Μεταγωγής (Public Switched Telephone Network - PSTN), προσφέρει τη δυνατότητα αμφίδρομης μετάδοσης φωνής, με μικρές καθυστερήσεις, αλλά και την εγγύηση της ολοκλήρωσης μιας κλήσης όταν αυτή εγκατασταθεί. Η επικοινωνία επιτυγχάνεται με τη δέσμευση ενός απ' άκρου σ' άκρο φυσικού κυκλώματος ανάμεσα στην πηγή και τον προορισμό της τηλεφωνικής κλήσης, για όσο χρόνο αυτή διαρκεί. Η παραπάνω τεχνική καλείται μεταγωγή κυκλώματος. Τα δεδομένα εισέρχονται στο δίκτυο από κάποια πηγή πληροφορίας (τερματική διάταξη) και μεταφέρονται μέσω ενδιάμεσων κόμβων στον

προκαθορισμένο δέκτη με την ίδια σειρά με την οποία δημιουργούνται. Οι κόμβοι διακινούν τα δεδομένα προς τον προορισμό τους αποφασίζοντας ή όχι για την αποτελεσματική διακίνησή τους. Το κύκλωμα αποδεσμεύετε μετά την ολοκλήρωση της κλήσης.

Οι πιο γνωστές υπηρεσίες που προσφέρει το δημόσιο τηλεφωνικό σύστημα είναι οι εξής:

- **Τηλεφωνικές υπηρεσίες χωρίς χρέωση** (αριθμοί που ξεκινούν με το 800). Για τη δρομολόγηση κλήσεων που αφορούν τέτοιους αριθμούς, θα πρέπει πρώτα να εντοπιστεί η κατάλληλη βάση δεδομένων και μετά να αναζητηθεί ο τελικός τηλεφωνικός αριθμός που θα χρησιμοποιηθεί.
- **Ασύρματη περιαγωγή**. Σ' αυτή την υπηρεσία, το δίκτυο χρησιμοποιεί μία σύνθετη σειρά από μηνύματα σηματοδοσίας για να καταστήσει δυνατές τις εισερχόμενες και εξερχόμενες κλήσεις. Οι βάσεις δεδομένων των συνδρομητών αναζητούνται ώστε να καθοριστούν τα δικαιώματα περιαγωγής σε νέες περιοχές.
- **Τηλεφωνικές κάρτες**. Όταν γίνεται μία κλήση με τηλεφωνική κάρτα, γίνεται προσπέλαση σε βάση δεδομένων για να προσδιοριστεί η εγκυρότητα της κάρτας, ώστε να γίνει η κατάλληλη χρέωση.

Τέτοιου είδους υπηρεσίες προσδίδουν στο PSTN το χαρακτηρισμό Intelligent Network (IN). Οι IN υπηρεσίες του PSTN αρχικοποιούνται, ελέγχονται και τερματίζονται από το πρωτόκολλο σηματοδοσίας SS7 (Signaling System 7).

Μιλώντας για μεταγωγή πακέτου (packet switching) αναφερόμαστε στη φιλοσοφία που διέπει τη μεταφορά δεδομένων σε ένα ή περισσότερα δίκτυα υπολογιστών. Στη μεταγωγή πακέτου δεν υπάρχει εκ των προτέρων σχηματιζόμενο φυσικό κανάλι για την επικοινωνία των δύο συνδρομητών, όπως συμβαίνει στη μεταγωγή κυκλώματος για την περαίωση μιας κλήσης. Εδώ κάθε μήνυμα «τεμαχίζεται» σε πακέτα σταθερού μήκους, που αποτελούν τη βασική μονάδα πληροφορίας. Η επιλογή του μήκους των πακέτων αποτελεί μια βασική σχεδιαστική παράμετρο. Δύο είναι οι βασικές προσεγγίσεις στη διαδικασία προώθησης των πακέτων:

- Η μέθοδος των αυτοδύναμων πακέτων (datagrams) και

- Η μέθοδος των νοητών κυκλωμάτων.

Στην πρώτη περίπτωση το δίκτυο χειρίζεται κάθε πακέτο ανεξάρτητα. Κάθε αυτοδύναμο πακέτο περιέχει όλη την απαραίτητη πληροφορία, σύμφωνα με το χρησιμοποιούμενο αλγόριθμο δρομολόγησης και τα πακέτα μπορούν να ακολουθήσουν διαφορετικές διαδρομές, για να φτάσουν στον κοινό προορισμό τους. Κάθε κόμβος που παραλαμβάνει το πακέτο επιλέγει ποιος θα είναι ο επόμενος έτσι ώστε το δίκτυο να λειτουργεί με τον καλύτερο δυνατό τρόπο. Αντίθετα στην περίπτωση των νοητών κυκλωμάτων, πριν την αποστολή ενός πακέτου, εγκαθίσταται μια νοητή σύνδεση μεταξύ των άκρων που πρόκειται να επικοινωνήσουν. Η εξασφάλιση της ύπαρξης ελεύθερου δρόμου γίνεται με την ανταλλαγή μηνυμάτων μεταξύ των δύο συνδρομητών που πρόκειται να επικοινωνήσουν. Σε αυτή την περίπτωση οι ενδιάμεσοι κόμβοι δεν απαιτείται να έχουν πληροφορίες για την κατάσταση του δικτύου, γιατί δεν αποφασίζουν για τη δρομολόγηση των μηνυμάτων απλά τα προωθούν στον προορισμό τους. Υπάρχουν δύο τύποι νοητών συνδέσεων :

- οι σταθερές νοητές συνδέσεις (PVCs – Permanent Virtual Connections), οι οποίες μοιάζουν με τις μισθωμένες γραμμές και
- οι μεταγώγιμες νοητές συνδέσεις (SVCs – Switched Virtual Connections), οι οποίες μπορεί να θεωρηθούν το ανάλογο των τηλεφωνικών κλήσεων.

Μετά από σύγκριση των δύο μεθόδων καταλήγουμε στα παρακάτω συμπεράσματα:

- Η μεταγωγή κυκλώματος είναι ιδανική για μετάδοση συνεχών σημάτων μεγάλης διάρκειας, π.χ. για μετάδοση φωνής (τηλεφωνικό δίκτυο) και εικόνας.
- Η δέσμευση και η αποκλειστική χρήση υπολογιστικών πόρων του δικτύου ανα κλήση, σε περιπτώσεις μεταφοράς δεδομένων, οδηγεί σε μειωμένη χρήση των πόρων του δικτύου.
- Η μεταγωγή κυκλώματος παρέχει συνδέσεις σταθερού ρυθμού μεταφοράς δεδομένων. Δύο διασυνδεδεμένες διατάξεις, πρέπει να χρησιμοποιούν τους ίδιους ρυθμούς κατά τη λήψη και την εκπομπή, με αποτέλεσμα να περιορίζεται η ικανότητα υποστήριξης τερματικών σημείων με διαφορετικά χαρακτηριστικά.

- Η τεχνική (αυτοδύναμων πακέτων) datagram είναι καλύτερη σε περίπτωση βλάβης (π.χ. καταστροφής ενός κόμβου), επειδή το μήνυμα θα φτάσει στον προορισμό του μέσω άλλων εναλλακτικών δρόμων. Αντίθετα στη μεταγωγή κυκλώματος, καταστροφή του διαθέσιμου καναλιού θα έχει σαν αποτέλεσμα την απώλεια του μηνύματος. Στη μεταγωγή εικονικού κυκλώματος υπάρχει μεγάλη πιθανότητα απώλειας του μηνύματος ή ανάγκη επαναμετάδοσής του, αφού σε περίπτωση που χαλάσει κάποιος κόμβος όλα τα μηνύματα που διέρχονται από τον κόμβο αυτόν θα χαθούν.
- Τα δίκτυα μεταγωγής πακέτου παρουσιάζουν χαμηλή ποιότητα υπηρεσιών σε περιπτώσεις συμφόρησης δρομολογητών και διαμοιραζόμενα κυκλώματα.

Στο παρακάτω σχεδιάγραμμα φαίνονται αναλυτικά τα αποτελέσματα της σύγκρισης των δύο μεθόδων:

Μεταγωγή κυκλώματος	**Datagram μεταγωγή πακέτου**	**Virtual-circuit μεταγωγή πακέτου**
Προκαθορισμένη διαδρομή μετάδοσης	Μη προκαθορισμένη διαδρομή μετάδοσης	Μη προκαθορισμένη διαδρομή μετάδοσης
Συνεχής μετάδοση δεδομένων	Μετάδοση πακέτων	Μετάδοση πακέτων
Γρήγορη απόκριση	Γρήγορη απόκριση	Γρήγορη απόκριση
Πακέτα δεν αποθηκεύονται	Πακέτα μπορεί να αποθηκεύονται μέχρι να διανεμηθούν	Πακέτα αποθηκεύονται μέχρι να διανεμηθούν
Ίδια διαδρομή για όλη τη σύνδεση	Η διαδρομή ορίζεται για κάθε πακέτο	Η διαδρομή ορίζεται για όλη τη σύνδεση
Καθυστέρηση δημιουργίας κλήσης, αμελητέα καθυστέρηση μετάδοσης	Καθυστέρηση μετάδοσης πακέτων	Καθυστέρηση δημιουργίας κλήσης, καθυστέρηση μετάδοσης πακέτων
Σήμα απασχόλησης εάν το καλούμενο τμήμα είναι απασχολημένο	Ο πομπός μπορεί να ειδοποιηθεί εάν το πακέτο δεν διανεμηθεί	Ο πομπός ειδοποιείται εάν η σύνδεση δε πραγματοποιηθεί
Υπερφόρτωση μπορεί να μπλοκάρει τη δημιουργία κλίσης, δεν έχει καθυστέρηση για καθιέρωση κλήσης	Υπερφόρτωση αυξάνει την καθυστέρηση πακέτων	Υπερφόρτωση μπορεί να μπλοκάρει τη δημιουργία κλίσης, αυξάνει την καθυστέρηση πακέτων

Ηλεκτρομηχανικοί ή ηλεκτρονικοί επιλογείς κόμβων	Μικροί επιλογείς κόμβων	Μικροί επιλογείς κόμβων
Χρήστης υπεύθυνος για την προστασία των μηνυμάτων που χάνονται	Δίκτυο μπορεί να είναι υπεύθυνο για τα ανεξάρτητα πακέτα	Δίκτυο μπορεί να είναι υπεύθυνο για την ακολουθία πακέτων
Συνήθως όχι αλλαγή ταχύτητας και κωδικών	Αλλαγή ταχύτητας και κωδικών	Αλλαγή ταχύτητας και κωδικών
Σταθερό εύρος ζώνης μετάδοσης	Δυναμική χρήση του εύρους ζώνης	Δυναμική χρήση του εύρους ζώνης
Όχι επιπλέον bits μετά τη δημιουργία κλήσης	Επιπλέον bits σε κάθε μήνυμα	Επιπλέον bits σε κάθε μήνυμα
Χρέωση ανά λεπτό	Χρέωση ανά πακέτο	Χρέωση ανά πακέτο

Πίνακας 1: Συγκριση μεταγωγής κυκλώματος με μεταγωγή πακέτου.

1.6 Εφαρμογές ενός VOIP δικτύου

Όπως έχουμε ήδη αναφέρει, μιλώντας για VOIP αναφερόμαστε στην ικανότητα εκείνη του δικτύου να πραγματοποιεί τηλεφωνικές κλήσεις (όπως ακριβώς και ένα PSTN δίκτυο) και να στέλνει μηνύματα τηλεομοιοτυπίας μέσω άλλων δικτύων δεδομένων που βασίζονταιι στο πρωτόκολλο IP, παρέχοντας την κατάλληλη ποιότητα φωνής και χαμηλό κόστος. Η τεχνολογία VOIP μπορεί να καλύψει μια μεγάλη γκάμα υπηρεσιών, ξεκινώντας από ένα απλό σύστημα ενδοεπικοινωνίας και καλύπτωντας ακόμα και multipoint συστήματα τηλεσυνεδριάσεων. Όσον αφορά την ποιότητα της φωνής, θα μπορούσε να τη διαχειριστεί διαφορετικά ανάλογα πάντα με το είδος της εφαρμογής και τις απαιτήσεις της. Επομένως ο VoIP εξοπλισμός πρέπει να έχει την ευελιξία να τροφοδοτεί σε μια μεγάλη κλίμακα διαμορφώσεων και διαφορετικών περιβαλλόντων και να επιτρέπει την επαναχρησιμοποίηση των παλιών τερματικών συσκευών.

Ακολοθούν μερικά παραδείγματα VOIP υπηρεσιών:

- **Πύλες PSTN:** Η αλληλοσύνδεση του Internet με το PSTN μπορεί να ολοκληρωθεί χρησιμοποιώντας μια πύλη που ενσωματώνεται σε ένα PBX ή που παρέχεται σα μια χωριστή συσκευή. Ένα PC-based τηλέφωνο, για παράδειγμα,

μπορεί να έχει πρόσβαση στο δημόσιο δίκτυο καλώντας μια πύλη σε ένα σημείο κοντά στον προορισμό (ελαχιστοποιώντας τις δαπάνες μεγάλων αποστάσεων).

- **Internet-aware τηλέφωνα:** Τα συνηθισμένα τηλέφωνα (ενσύρματα ή ασύρματα) μπορούν να εξοπλιστούν κατάλληλα ώστε να λειτουργούν σαν μια Internet συσκευή πρόσβασης και ταυτόχρονα να παρέχουν τις υπηρεσίες ενός κανονικού τηλεφώνου. Οι υπηρεσίες καταλόγου θα μπορούσαν να προσεγγιστούν μέσω του Internet με την υποβολή του ονόματος και να λαμβάνετε μια φωνητική (ή γραπτή) απάντηση.
- **Εσωτερική ζεύξη πάνω από εταιρικά δίκτυα:** Η δικτυακή σύνδεση ανάμεσα στην εταιρεία και στο κύριο PBX παρέχει μεγάλα οικονομικά οφέλη και βοηθά να παγιωθούν οι δυνατότητες των δικτύων.
- **Μακρινή πρόσβαση από ένα περιφερειακό γραφείο:** Ένα μικρό γραφείο θα μπορούσε να αποκτήσει πρόσβαση σε εταιρική φωνή, δεδομένα και άλλες υπηρεσίες χρησιμοποιώντας το δίκτυο της εταιρείας.
- **Φωνητικές κλήσεις από ένα φορητό υπολογιστή μέσω του Internet:** Οι κλήσεις εντός γραφείου μπορούν να πραγματοποιούνται χρησιμοποιώντας ένα PC που συνδέεται στο internet.
- **Κεντρική πρόσβαση στο internet:** Η εφαρμογή του ηλεκτρονικού εμπορίου διευκολύνετε πολύ από την τεχνολογία VOIP. Η κύρια πρόσβαση στο διαδίκτυο δίνει τη δυνατότητα σε έναν administrator να έχει online πρόσβαση στις υπηρεσίες των πελατών του. Η φτωχή online επικοινωνία είναι ένα από τα μεγαλύτερα εμπόδια που αντιμετωπίζουν οι εταιρείες, από τη στιγμή που οι επισκέπτες των ιστοσελίδων τους δύσκολα μετατρέπονται σε αγοραστές. Σε ένα κατάστημα οι πελάτες μπορούν να ζητήσουν περισσότερες πληροφορίες από τους υπαλλήλους. Σε μια ιστοσελίδα αυτού του είδους η επικοινωνία είναι προβληματική. Χρησιμοποιώντας όμως την τεχνολογία VoIP οι επισκέπτες των ιστοσελίδων θα μπορούν πατώντας ένα κουμπί να ανοίξουν μια φωνητική συνομιλία με ένα κέντρο που μπορεί να απαντήσει οποιαδήποτε ερώτηση τους ή να εξετάσει κάποιο πρόβλημα που έχουν.
- Οι υπηρεσίες πραγματικού χρόνου είναι άλλη μία καινοτομία των VOIP δικτύων. Φωνητικά μηνύματα μπορούν να δημιουργηθούν χρησιμοποιώντας τις κλασικές τηλεφωνικές συσκευές και να αποσταλούν σε Voice mail boxes, χρησιμοποιώντας υπηρεσίες τόσο του τοπικού δικτύου, όσο και του διαδικτύου.

- Τέλος, αξίζει να αναφέρουμε ότι η IP τηλεφωνία χρησιμοποιείτε από εταιρείες με ιδιωτικά δίκτυα, από ευρείας περιοχής δίκτυα, από τα εκτεταμένα δίκτυα (extranets) και από συνδιασμούς δικτύων. Χαρακτηριστικά αναφέρουμε: μεταφορείς συναλλαγών (InterExchange Carriers, IXCs), μεγάλης αποστάσεως μεταφορείς, εταιρείες παροχής υπηρεσιών Internet (Internet Service Providers, ISPs), εταιρείες παροχής τηλεφωνίας μέσω Internet (Internet Telephony Service Providers, ITSPs), τηλεφωνικά κέντρα και ευρύτερες υπηρεσίες γραφείων.

2ο ΚΕΦΑΛΑΙΟ

ΒΑΣΙΚΑ ΧΑΡΑΚΤΗΡΙΣΤΙΚΑ – ΛΕΙΤΟΥΡΓΙΕΣ ΕΝΟΣ VOIP ΔΙΚΤΥΟΥ

Σε αυτό το κεφάλαιο θα αναπτύξουμε τις βασικές δικτυακές λειτουργίες που καθιστούν δυνατή την επίτευξη της επικοινωνίας μεταξύ δύο ή περισσοτέρων χρηστών σε ένα VOIP δίκτυο . Πιο συγκεκριμένα θα μιλήσουμε για:

- Υπηρεσίες βάσεων δεδομένων για τον εντοπισμό των χρηστών καθώς και την μετάφραση διαφορετικών τρόπων διευθυνσιοδότησης, στην περίπτωση που οι τελικοί χρήστες ανήκουν σε ετερογενή δίκτυα.
- Υπηρεσίες σηματοδότησης για την εγκατάσταση, τον έλεγχο και τον τερματισμό μιας κλήσης.
- Μηχανισμούς μεταφοράς δεδομένων φωνής
- Διαδικασίες κωδικοποίησης-αποκωδικοποίησης (CODEC) για την μετατροπή του αναλογικού σήματος φωνής σε ψηφιακό και αντίστροφα, για να είναι δυνατή η μετάδοση μέσω του επικοινωνιακού διαύλου και η μετατροπή του σήματος φωνής σε ακουστικό σήμα.

2.1Υπηρεσίες βάσεων δεδομένων

Μία από τις πλέον τεχνικές προκλήσεις που προέκυψε από τη συνύπαρξη των δικτύων μεταγωγής κυκλώματος με τα δίκτυα μεταγωγής πακέτων, είναι η δρομολόγηση των κλήσεων από το ένα δίκτυο στο άλλο. Γενικά, η πιο επιθυμητή προσέγγιση ήταν η ύπαρξη ενός κοινού σχεδίου δρομολόγησης κλήσεων ανεξαρτήτως δικτύου. Μιλώντας για “telephone number mapping” αναφερόμαστε στην διαδικασία αντιστοίχησης του τηλεφωνικού συστήματος αριθμοδότησης του δημόσιου τηλεφωνικού δικτύου (PSTN), με την διευθυνσιοδότηση που χρησιμοποιείτε στο internet. Τα τηλεφωνικά νούμερα είναι οργανωμένα με βάση το

E.164 standard, ενώ το internet χρησιμοποιεί τη τεχνολογία DNS για την αντιστοίχηση ενός domain name με μια IP address. Τα συστήματα "telephone number mapping" είναι υπεύθυνα για τον καθορισμό των κατάλληλων Internet communications servers, οι οποίοι αναλαμβάνουν να εξυπηρετήσουν έναν συνδρομητή με ένα απλό lookup στο DNS (Domain Name System).

2.1.1 ENUM SERVER

Το E.164 Number Mapping (ENUM) standard είναι ο πιο διακεκριμένος τρόπος αντιστοίχηση της διεθνούς ITU-T E.164 στάνταρ αριθμοδότησης με το Internet Domain Name System ("DNS"). Το ENUM χρησιμοποιεί ειδικούς τύπους DNS εγγραφών για τη μετάφραση ενός τηλεφωνικού αριθμού σε ένα URI (Uniform Resource Identifier) ή μια IP address, οι οποίες είναι γνωστές ως Name Authority Pointers (NAPTR). Τα Universal Resource Names (URNs) είναι ένα υποσύνολο των Universal Resource Identifiers (URIs) τα οποία χησιμοποιούνται για τα αφηρημένα προσδιοριστικά, όπως είναι το όνομα και το νούμερο ενός συνδρομητή. Οι NAPRT εγγραφές οι οποίες μπορεί να ειναι URNs, URLs ή και domain names, καθοδηγούν τους εκάστωτε clients να χρησιμοποιήσουν το κατάλληλο πρωτόκολλο για τη πορεία της κλήσης. Οι "Naming Authority Pointer Resource Records" όπως ορίζονται στο RFC 3403, μπορεί να είναι e-mail addresses, αριθμοί fax, ένα προσωπικό website, ένας αριθμός VoIP, ένας αριθμός κινητού τηλεφώνου, voice mail systems, IP-telephony addresses, web pages, GPS coordinates, call diversions or instant messaging.

Για να μπορέσει ένας ENUM συνδρομητής να ενεργοποιήσει και να χρησιμοποιήσει την υπηρεσία ENUM, θα πρέπει να έχει αποσπάσει τα παρακάτω χαρακτηριστκά από έναν Registrar server:

- Ένα προσωπικό URI (Uniform Resource Identifier) το οποίο θα χρησιμοποιήσει στο IP κομμάτι του δικτύου.
- Ένα προσωπικό E.164 τηλεφωνικό μούμερο, συσχετιζόμενο με το προσωπικό του URI.

- Τη δικαιωδοσία να καταγράφει τις προσωπικές του προτιμήσεις για την προώθηση και τον τερματισμό μιας κλήσης σε μια NAPTR εγγραφή η οποία είναι προσβάσιμη μόνο από το προσωπικό του URI.

Αναλυτικότερα η διαδικασία που ακολουθείτε είναι η εξής:

- Ο εκάστωτε Registrar server επιστρέφει το συνδρομητή με ένα συγκεκριμένο domain name, το οποίο όπως ήδη αναφέραμε αποκαλείτε URI και το οποίο θα χρησιμοποιηθεί από τον DNS για την αντιστοίχηση σε μια NAPRT εγγραφή.
- Το URI domain name αντιστοιχίζεται κατ' αποκλειστικότητα με το E.164 αριθμό του συνδρομητή (ENUM number).
- Τελικά, με βάση την NAPTR εγγραφή η οποία αντιστοιχεί στο URI του συγκεκριμένου συνδρομητή, οι κλήσεις θα προωθηθούν και θα καταλήξουν εκεί που επιθυμεί ο εκάστωτε συνδρομητής.

Επομένως, εαν μια κλήση αρχικοποιηθεί από το PSTN δίκτυο πληκτρολογώντας ένα E.164 ENUM αριθμό, ο αριθμός αυτός θα μεταφραστεί από τον ENUM gateway στο αντίστοιχο URI. Στη συνέχεια αυτό το URI θα μας παραπέμψει σε μια NAPTR εγγραφή η οποία περιέχει όλη την πληροφορία για τη πορεία της κλήσης. Από την άλλη πάλι όταν το calling party ανήκει στο IP δίκτυο, ο User Agent αναλαμβάνει να μετατρέψει το πληκτρολογούμενο E.164 αριθμό του called party στο αντίστοιχο URI, το οποίο βέβαια θα αντιστοιχηθεί με τη σειρά του από τον ENUM DNS gateway σε μια NAPTR εγγραφή.

Η ITU ENUM διαθέτει μια συγκεκριμένη ζώνη για χρήση των ENUM E.164 αριθμών στο IP δίκτυο, η οποία ονομάζεται e164.arpa. Το RFC 6116 καθορίζει πώς ένας E.164 αριθμός, μπορεί να μετατραπεί σε ένα URI, αντιστρέφωντας τα νούμερα, χωρίζοντας τα με τελείες και προσθέτοντας ως κατάληξη το e164.arpa. Το URI στη συνέχεια μπορεί να χρησιμοποιηθεί για τη λήψη IP διευθύνσεων για υπηρεσίες όπως η SIP VOIP τηλεφωνία που μελετάμε εδώ.

2.1.1.1 ΠΑΡΑΔΕΙΓΜΑΤΑ

2.1.1.1.a Μετατροπή ενός E.164 αριθμού σε ένα URI

Ας υποθέσουμε ότι έχουμε έναν E.164 αριθμό, για παράδειγμα τον +1 555 42 42. Για να βρούμε σε πιο URI αντιστοιχεί αρκεί να αντιστρέψουμε τα νούμερα, να τα χωρίσουμε με τελείες και να προσθέουμε την κατάληξη e164.arpa. Επομένως το ζητούμενο URI είναι το 2.4.2.4.5.5.5.1.e164.arpa.

Ας υποθέσουμε ότι έχουμε την παρακάτω NAPTR εγγραφή:

```
$ORIGIN 2.4.2.4.5.5.5.1.e164.arpa.

IN NAPTR 100 10 "u" "E2U+sip"  "!^.*$!sip:phoneme@example.net!" .

IN NAPTR 102 10 "u" "E2U+mailto" "!^.*$!mailto:myemail@example.com!" .
```

Η εγγραφή έχει τιμή σειράς (order value) 100, το οποίο είναι μικρότερο από το 102, οπότε θα επιλεγεί πρώτη. Η τιμή προτίμησης (preference value) είναι 10, αλλά στο συγκεκριμένο παράδειγμα δε παίζει κανένα ρόλο μιας και δε υπάρχει άλλη εγγραφή με τη ίδια τιμή σειράς. Το συγκεκριμένο παράδειγμα υποδηλώνει ότι αν θέλουμε να χρησιμοποιήσουμε την υπηρεσία "E2U+sip", θα πρέπει να χρησιμοποιήσουμε σαν διεύθυνση την sip:phoneme@example.net, η οποία θα αντικαταστήσει το URN tel: +1 555 42 42. Τέλος, το "U" flag υποδηλώνει ότι το string το οποίο θα αντικατασταθεί είναι ένα SIP URN, και ότι κανένας περαιτέρω κανόνας δεν πρέπει να εφαρμοστεί.

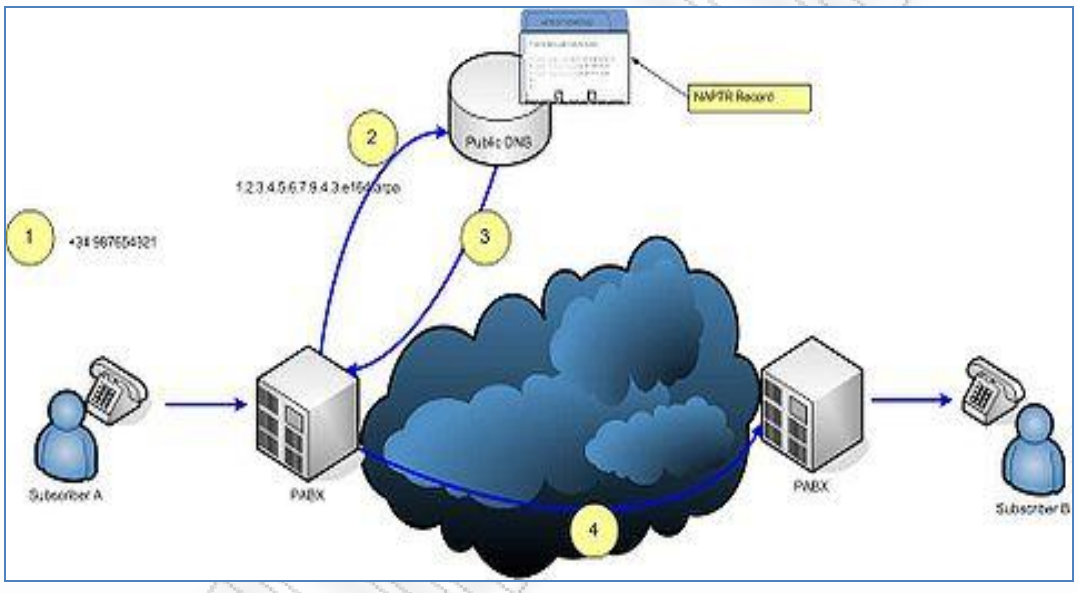
2.1.1.1.b Εγκατάσταση και τερματισμός μιας κλήσης χρησιμοποώντας τον ENUM

Ας υποθέσουμε ότι ένας συνδρομητής A αρχικοποιεί μια κλήση προς ένα συνδρομητή B (Σχήμα 2).

- Ο User Agent μιας τερματικής συσκευής, ή κάποιο PBX, ή ένα gateway, αναλαμβάνουν να μεταφράσουν τον ENUM-συνδρομητή με αριθμό +34

98 765 4321 στο ENUM domain 1.2.3.4.5.6.7.8.9.4.3.e164.arpa, σύμφωνα με τη διαδικασία που περιγράφτηκε παραπάνω.

- Το συγκεκριμένο request θα σταλεί στη συνέχεια στο Domain Name System (DNS), ζητώντας του να αναζητήσει τις εγγραφές για το συγκεκριμένο ENUM domain 1.2.3.4.5.6.7.8.9.4.3.e164.arpa.
- Το συγκεκριμένο query θα επιστρέψει μια NAPTR εγγραφή, σύμφωνα με το RFC 3403. Στη συγκεκριμένη περίπτωση, η απάντηση είναι μια SIP διεύθυνση.
- Το τερματικό μπορεί τώρα να εγκαταστήσει μια σύνδεση επικοινωνίας και να δρομολογήσει την κλήση μέσω του διαδικτύου.

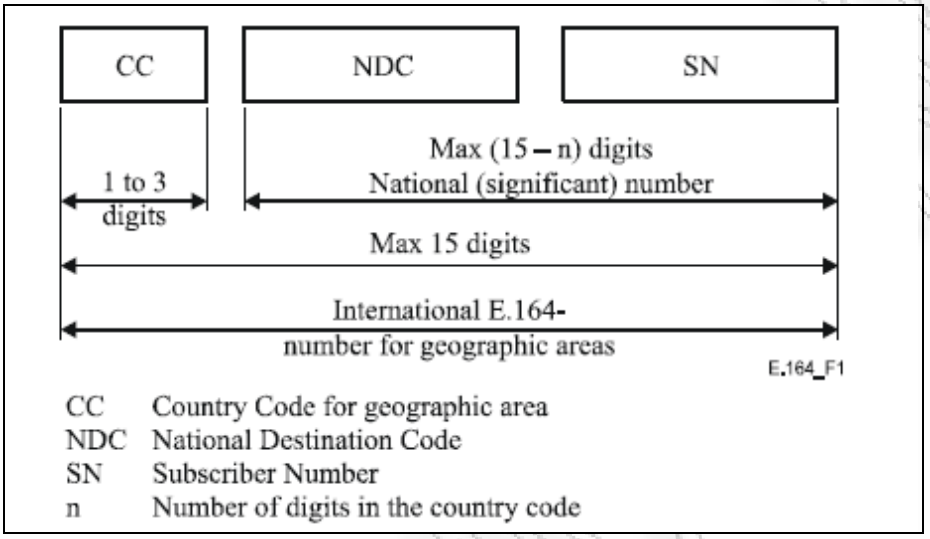


Σχήμα 2: Δρομολόγηση μιας SIP κλήσης με τη χρήση ENUM server.

2.1.2 Αριθμοδότηση E.164

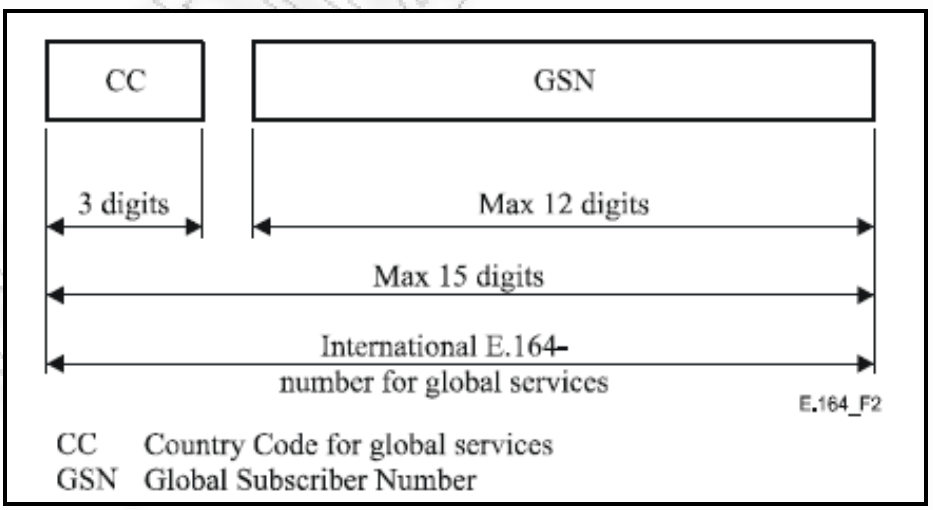
Το E.164 είναι μια σύσταση της ITU η οποία καθορίζει το διεθνή τρόπο αριθμοδότησης τόσο για το δημόσιο τηλεφωνικό δίκτυο, όσο και για άλλα δίκτυα δεδομένων. Τα E.164 νούμερα μπορεί να έχουν το πολύ 15 ψηφία και συνήθως είναι γραμμένα με ένα πρόθεμα + στο ξεκίνημα τους. Αυτή η σύσταση καθορίζει ουσιαστικά τη δομή και τη λειτουργία διαφορετικών κατηγοριών αριθμών που χρησιμοποιούνται στις διεθνείς δημόσιες τηλεπικοινωνίες. Οι πιο βασικές κατηγορίες αριθμών είναι οι εξής:

- **Δομή αριθμών για γεωγραφικές περιοχές**. Η E.164 αριθμοδότηση για τις γεωγραφικές περιοχές αποτελείται από έναν μεταβλητό αριθμό δεκαδικών ψηφίων οι οποίοι διαχωρίζονται σε συγκεκριμένες περιοχές όπως φαίνετε στο Σχήμα 3.



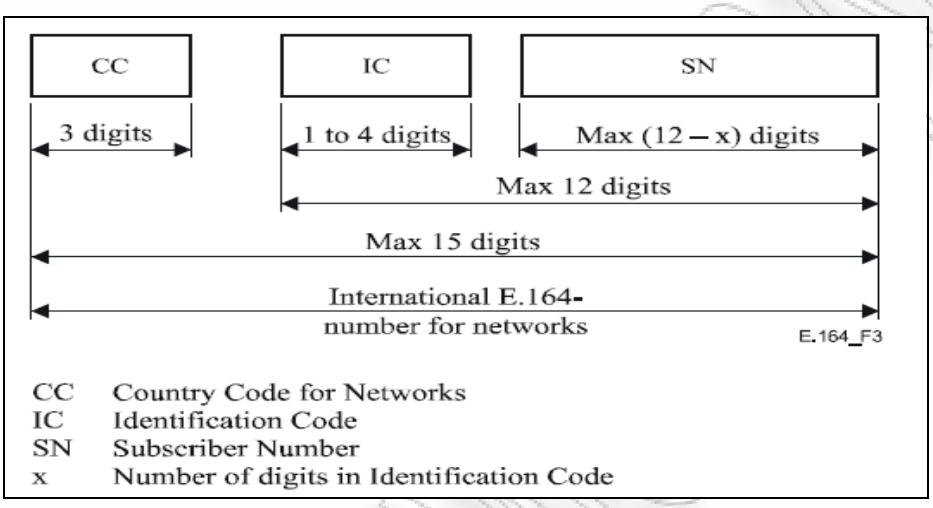
Σχήμα 3: Δομή των International E.164-αριθμών για γεωγραφικές περιοχές.

- **Δομή αριθμών για παγκόσμιες υπηρεσίες**. Η E.164 αριθμοδότηση για τις παγκόσμιες υπηρεσίες αποτελείται από δεκαδικά ψηφία που ποικίλλουν ανάλογα με τη συγκεκριμένη υπηρεσία. Ένας τέτοιου είδους αριθμός αποτελείτε από ένα τριψήφιο country code(CC) για τις υπηρεσίες και τον Global Subscriber Number (GSN).



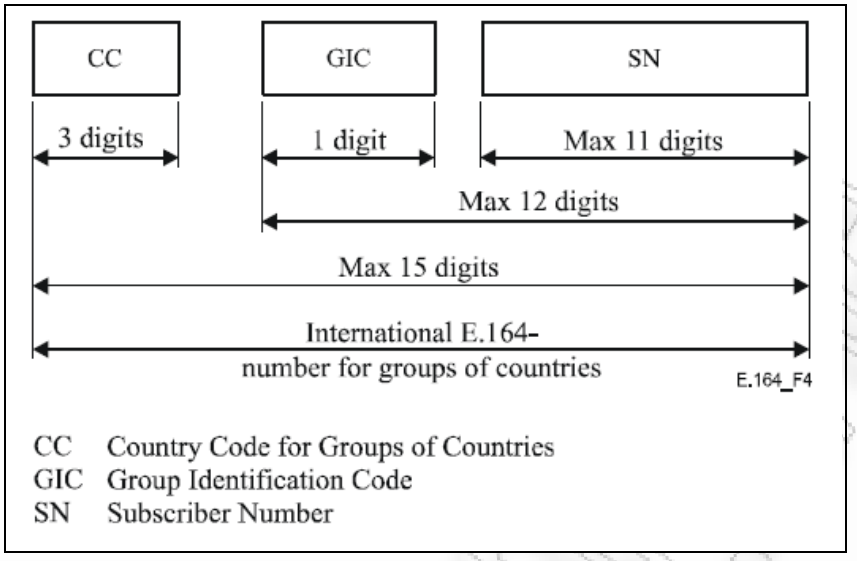
Σχήμα 4: Δομή των International E.164-αριθμών για παγκόσμιες υπηρεσίες.

- **Δομή αριθμών για δίκτυα.** Η E.164 αριθμοδότηση για τα δίκτυα αποτελείται από δεκαδικά ψηφία τα οποία είναι χωρισμένα σε τρία πεδία. Τα τρία ψηφία του country code (CC) αποτελούν το κομμάτι του δικτύου, ακολουθούν το IC κομμάτι, το οποίο ποικίλλει σε μέγεθος από ένα εως τέσσερα ψηφία, και ο αριθμός του συνδρομητή (SN).



Σχήμα 5: Δομή των International E.164-αριθμών για δίκτυα.

- **Δομή αριθμών για ομάδες χωρών.** Η E.164 αριθμοδότηση για ομάδες χωρών αποτελείτε από δεκαδικά ψηφία χωρισμένα σε τρία πεδία. Το τριψήφιο country code (CC) πεδίο για την ομάδα χωρών, το Group Identification Code (GIC) πεδίο το οποίο έχει πάντα το μέγεθος ενός ψηφίου, και τον αριθμό του συνδρομητή (SN) ο οποίος μπορεί να έχει μέγεθος μέχρι 11 ψηφία.



Σχήμα 6: Δομή των International E.164-αριθμών για ομάδες χωρών

2.1.3 Χαρακτηριστικά ποιότητας

2.1.3.1 Καθυστέρηση

Η μετατροπή των αναλογικών δεδομένων σε ψηφιακά γίνεται με τη βοήθεια ενός κωδικοποιητή – αποκωδικοποιητή (Codec). Σύμφωνα με το θεώρημα Nyquist, ο ρυθμός δειγματοληψίας πρέπει να είναι τουλάχιστον διπλάσιος από την υψηλότερη συχνότητα που παρουσιάζεται στο αναλογικό σήμα. Στο δημόσιο τηλεφωνικό δίκτυο (PSTN) το αναλογικό σήμα δειγματοληπτείται στα 8 kHz για παλμοκωδική διαμόρφωση (Pulse Code Modulation, PCM). Το G.711 είναι το διεθνές πρότυπο για την κωδικοποίηση του τηλεφωνικού ήχου σε ένα κανάλι 64 kbps. Είναι μια PCM διάταξη που λειτουργεί με ρυθμό 8 bits ανά δείγμα. Υπάρχουν δύο παραλλαγές του G.711, ο κανόνας A (A-law) και ο μ-κανόνας (μ-law). Ο A-law υιοθετείται στην Ευρώπη ενώ ο μ-law χρησιμοποιείται στην Αμερική και την Ιαπωνία. Το όφελος από αυτές τις τεχνικές είναι πως ενώ διατηρούν υψηλή ποιότητα, το εύρος ζώνης συχνοτήτων μειώνεται σημαντικά. Πιο προηγμένοι αλγόριθμοι συμπίεσης μπορούν να χρησιμοποιηθούν σε περιπτώσεις έλλειψης εύρου ζώνης. Αυτό όμως μειώνει την ποιότητα ομιλίας.

Η παροχή ενός επιπέδου ποιότητας φωνής τουλάχιστον ισοδύναμο με αυτό του PSTN είναι βασική απαίτηση σε ένα VOIP δίκτυο. Αν και η ποιότητα των υπηρεσιών

αναφέρεται συνήθως στην πιστότητα της μεταδιδόμενης φωνής, μπορεί να απευθυνθεί επίσης στη διαθεσιμότητα των δικτύων (ικανότητα κλήσης, βαθμός μπλοκαρίσματος κλήσεως), στη διαθεσιμότητα των τηλεφωνικών χαρακτηριστικών (εμφάνιση αριθμού κλήσεως) και στην εξέλιξη.

Το πρόβλημα στις μεταδόσεις φωνής μέσα από δίκτυα δεδομένων όπως IP (VoIP) είναι η καθυστέρηση (delay) στα πακέτα δεδομένων η οποία εισέρχεται λόγο της αρχιτεκτονικής των δικτύων αυτών. Τα επιτρεπτά επίπεδα καθυστέρησης σύμφωνα με τις συστάσεις της ITU είναι ανάμεσα στα 150-400 msec. Η καθυστέρηση χωρίζεται σε δύο βασικές κατηγορίες:

- Την καθυστέρηση διάδοσης (**propagation delay**) που αποδίδεται στα χαρακτηριστικά του μέσου, και
- Την καθυστέρηση επεξεργασίας (**handling delay**) που αποδίδεται στις διάφορες ενεργές συσκευές οι οποίες παρεμβαίνουν στη διαδρομή από την εκκίνηση στον προορισμό.

Η καθυστέρηση επεξεργασίας (την οποία σε κάποιο σημαντικό βαθμό μπορούμε να την ελέγξουμε) χωρίζεται σε τρεις υποκατηγορίες ανάλογα με το σημείο του δικτύου που παρουσιάζεται:

α) **Καθυστέρηση επεξεργασίας στο σημείο εκκίνησης**. Στο σημείο αυτό παρατηρούνται δύο είδη καθυστερήσεων. Το πρώτο είδος καθυστέρησης οφείλετε στο codec (**codec delay).** Η κωδικοποίηση της φωνής, η συμπίεση και η τοποθέτηση των δεδομένων φωνής σε πακέτα, είναι λειτουργίες που εκτελεί ο codec και είναι υπεύθυνες για πολλές καθυστερήσεις. Συνολικά ο (codec) μπορεί να προκαλέσει καθυστέρηση ως και 35 msec (ανάλογα το είδος του codec). Ο παρακάτω πίνακας περιέχει ενδεικτικά κάποια είδη codec και την αντίστοιχη καθυστέρηση που εισάγουν.

CODEC	MOS Score	Delay(msec)
PCM (G.711)	4.4	0.75
32K ADPCM (G.726)	4.2	1
16K LD-CELP (G.728)	4.2	3-5
8K CS-ACELP (G.729)	4.2	10
8K CS-ACELP (G.729a)	4.2	10
6.3 MPMLG (G.723.1)	3.98	30
5.3 ACLEP (G723.1)	3.5	30

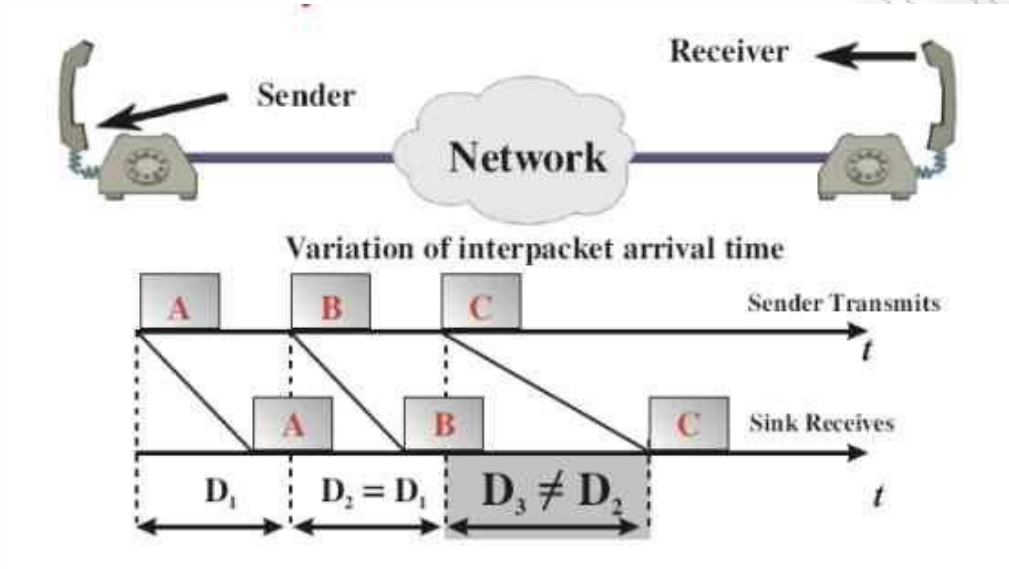
Πίνακας 2: Codec Delays

Το δεύτερο είδος καθυστέρησης που παρατηρούμε στο σημείο εκκίνησης προέρχεται από την καθυστέρηση των (IP πια) πακέτων στην ουρά εξόδου (Output Queuing Delay) του δρομολογητή μας (για παράδειγμα, στην ουρά εξόδου μιας σειριακής σύνδεσης). Αυτό το είδος καθυστέρησης είναι μεταβλητό και πολύ πιο δύσκολο να προσδιοριστεί. Εξαρτάται από πολλούς παράγοντες μεταξύ των οποίων και τα ίδια τα χαρακτηριστικά του δρομολογητή. Πρέπει γενικά να διατηρείται σε επίπεδα κάτω των 10 msec.

β)**Καθυστέρηση επεξεργασίας μέσα στο δίκτυο**. Σαν καθυστέρηση επεξεργασίας μέσα στο δίκτυο ορίζουμε το χρόνο που θα χρειαστεί το πακέτο να φτάσει από τον δρομολογητή εκκίνησης (αφού έχει γίνει η κατάλληλη επεξεργασία και το πακέτο είναι έτοιμο για εκπομπή) στον δρομολογητή προορισμού (μόλις το πακέτο εισαχθεί στην ουρά εισόδου). Είναι φανερό ότι τις καθυστερήσεις αυτές, οι οποίες είναι εσωτερικές στο δίκτυο κορμού, δεν μπορεί να τις ελέγξει ο απλός χρήστης. Είναι ευθύνη του παροχέα υπηρεσιών αν πρόκειται για δημόσιο δίκτυο, ή του administrator αν πρόκειται για κάποιο μεγάλο ιδιωτικό δίκτυο. Ο ρυθμός μετάδοσης των δεδομένων μας από το σημείο εκκίνησης προς το δίκτυο (serialization Up-link delay), ο ρυθμός μετάδοσης από το δίκτυο προς το σημείο προορισμού (serialization Down-link delay) και οι εσωτερικές καθυστερήσεις του ίδιου του δικτύου κορμού (General Network Delay) αποτελούν τους κυριότερους παράγοντες οι οποίοι ευθύνονται για αυτού του είδους καθυστέρηση.

γ) **Καθυστέρηση επεξεργασίας στο σημείο προορισμού.** Στο σημείο προορισμού η καθυστέρηση είναι συνδυασμός των καθυστερήσεων από τον codec **(codec delay),** την καθυστέρηση στην ουρά εισόδου του δρομολογητή **(Input Queuing Delay)** και

την καθυστέρηση λόγο του **jitter buffer.** Το jitter ορίζεται ως η διαφορά μεταξύ του αναμενόμενου (θεωρητικού) χρόνου άφιξης ενός πακέτου στο σημείο προορισμού και του πραγματικού χρόνου άφιξης.



Σχήμα 7: Καθυστέρηση λόγω jitter.

Για παράδειγμα, αν στο σημείο εκκίνησης παράγονται πακέτα φωνής ανα χρονικά διαστήματα ισοδύναμα με D1, λόγω των καθυστερήσεων στο σημείο προορισμού θα παρατηρήσουμε άφιξη πακέτων σε χρονικά διαστήματα διαφορετικά από D1. Η διαφορά αυτή αποτελεί το jitter. Η απάντηση στο πρόβλημα του jitter δίνεται με την χρήση buffering στο σημείο προορισμού, έτσι ώστε ο δρομολογητής προορισμού να παράγει με σταθερό ρυθμό τα πακέτα φωνής στον τελικό χρήστη ‘κρύβοντας’ την μεταβλητή αυτή καθυστέρηση.

Συνυπολογίζοντας κανείς όλες τις επιμέρους καθυστερήσεις του δικτύου μπορεί να προϋπολογίσει ην καθυστέρηση που ένα πακέτο φωνής μπορεί να συναντήσει κατά τη διάρκεια του ταξιδιού του. Το παρακάτω σχήμα αποτελεί ένα χαρακτηριστικό παράδειγμα των καθυστερήσεων σε μια end to end κλήση.

	Fixed Delay	Variable Delay
Κάθυστέρηση μετάδοσης Propagation Delay (Private Lines)	20 msec	
Σημείο εκκίνησης CODEC Delay G. 729 (5 msec look ahead)	5 msec	
Σημείο εκκίνησης CODEC Delay G.729 (10msec per frame)	20 msec	
Σημείο εκκίνησης - Output Queuing delay		6 msec
Serialization Delay uplink 64 Kbps Trunk - 64 bytes πακέτο	3msec	
Network Delay (e.g. Public Frame Relay)		?
Serialization Delay Downlink 64 Kbps Trunk - 64 bytes πακέτο	3msec	
Σημείο προορισμού - Imput Queuing delay		5msec
Σημείο προορισμού CODEC delay G.729	8msec	
Σημείο προορισμού Dejitter Buffer		4-200ms
Total	120msec	

50ms used in this example

Σχήμα 8: Υπολογισμός συνολικών καθυστερήσεων

2.1.3.2 Ηχώ

Η ηχώ (echo) και η επικάλυψη του ομιλητή (talker overlap) είναι κυρίως τα προβλήματα που προκαλούνται από τις καθυστερήσεις σε μια end to end κλήση. Η ηχώ αποτελεί πρόβλημα όταν η καθυστέρηση επιστροφής είναι πάνω από 50 msec. Στα παραδοσιακά δημόσια τηλεφωνικά δίκτυα, η ηχώ είναι αποτέλεσμα των ανακλάσεων των σημάτων που παράγονται από την υβριδική σύνδεση που μετατρέπει ένα κύκλωμα 4-καλωδίων (2 χωριστά ζευγάρια για μετάδοση και λήψη) σε ένα κύκλωμα 2-καλωδίων (1 ζευγάρι για μετάδοση και λήψη). Όταν το σήμα περάσει από το κύκλωμα 4-καλωδίων στο κύκλωμα 2-καλωδίων, μερική από την ενέργεια του 4-καλωδίων αντανακλάται και επιστρέφει στον ομιλητή. Καθώς η καθυστέρηση ανάμεσα στη φωνή και το ανακλώμενο σήμα αυξάνει, η ηχώ γίνεται ενοχλητική επειδή γίνεται αντιληπτή από τους χρήστες του δικτύου. Τα 50 msec είναι το μέγιστο ποσοστό που δε γίνεται αντιληπτό από τον ομιλητή. Σύμφωνα με τα πρότυπα της ITU όλα τα δίκτυα μεταγωγής κυκλώματος διαμορφώνονται έτσι ώστε να απαλείφουν κάθε αντήχηση πάνω από 45 ή 50 msec ανάλογα με το δίκτυο.

Στα δίκτυα μεταγωγής κυκλώματος για τη μείωση της ηχούς στα τηλεφωνικά κυκλώματα χρησιμοποιούμε συσκευές όπως οι **echo suppressors (** ή acoustic echo suppressor). Οι echo suppressors ανιχνεύουν εάν υπάρχει ένα σήμα φωνής σε μια από τις κατεύθυνσεις του κύκλωματος και προσθέτουν ένα μέγεθος απωλειών στην άλλη κατεύθυνση, εμποδίζοντας τον ομιλητή να ακούει τη φωνή του.

2.1.3.3 Απώλεια πακέτων

Στα VOIP δίκτυα τα IP πακέτα παρέχουν μια εγγύηση ότι τα πακέτα θα διανεμηθούν παντού, αν και κάποια από αυτά θα απορριφθούν σε περιπτώσεις που έχουμε μέγιστο φορτίο και κατά τη διάρκεια περιόδων συμφόρησης. Ωστόσο χρησιμοποιούνται μέθοδοι όπως η παρεμβολή της ομιλίας με την επανάληψη του τελευταίου πακέτου, και η αποστολή επιπλέον πληροφορίας, οι οποίες αντισταθμίζουν εως ένα βαθμό τα πακέτα που χάνονται.

2.1.4 Διαδικασίες Κωδικοποίησης-Αποκωδικοποίησης (CODEC)

Η διαδικασία μετατροπής των αναλογικών ακουστικών σημάτων σε ψηφιακά, ώστε να είναι δυνατή η μετάδοση του σήματος μέσω του δικτύου, καθώς και η αντίστροφη διαδικασία για την παραγωγή ακουστικού σήματος για τον τελικό παραλήπτη, γίνεται με τη χρήση του κωδικοποιητή – αποκωδικοποιητή. Η μεταφορά της πληροφορίας υπό μορφή παλμών εξασφαλίζει την ευκολότερη μεταφορά τους με τις μικρότερες καταστροφές. Συνεπώς η αναλογική πληροφορία, όπως είναι η ανθρώπινη ομιλία, πρέπει να μετατρέπεται σε ψηφιακή μορφή. Καποιοί βασικοί παράγοντες που οδήγησαν στην ψηφιοποίηση του αναλογικού σήματος είναι οι εξής:

1. Χρήση της τεχνολογίας TDM (Time Diversion Multiplex). Η τεχνική αυτή είναι πολύ πιο φθηνή συγκριτικά με την πολυπλεξία στη συχνότητα, εφόσον μειώνεται το κόστος των γραμμών μεταφοράς.
2. Παρέχει δευκολύνσεις στον ελέγχο και τη διαχείριση του δικτύου, καθώς τα μηνύματα ελέγχου είναι από τη φύση τους ψηφιακά.
3. Δίνει τη δυνατότητα χρήσης της φθηνής ψηφιακής τεχνολογίας των ολοκληρωμένων κυκλωμάτων.
4. Δυνατότητα αναγέννησης των ψηφιακών σημάτων και μείωση των επιδράσεων θορύβου.

Αξίζει μόνο να αναφερθεί ότι υπάρχουν πολλοί αλγόριθμοι για το μετασχηματισμό ενός αναλογικού σήματος φωνής σε ψηφιακό. Κάθε αλγόριθμος έχει διαφορετικές ανάγκες εύρους ζώνης και διαφορετικές ικανότητες συμπίεσης. Στο παρακάτω πίνακα παρουσιάζονται μερικοί από αυτούς.

Name	Standard by	Description	bit rate (kb/s)	sampling rate (kHz)	Raw Bandwidth Usage	Remarks
(ADPCM) DVI	Intel, IMA	ADPCM	32	8	var	
G.711	ITU-T	Pulse code modulation (PCM)	64	8	87.2 Kbps	Also known as ulaw/alaw, mu-law (US, Japan) and A-law (Europe)
G.722	ITU-T	7 kHz audio-coding within 64 kbit/s	64	16	* 120 Kbps +	Subband-codec that divides 16 kHz band into two subbands, each coded using ADPCM
G.722.1	ITU-T	Coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss	24/32	16	* 60 Kbps +	Variable Frame Size
G.723.1	ITU-T	Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s	5.3/6.4	8	20.8/21.9 Kbps	Part of H.324 video conferencing. DSP Group.
G.726	ITU-T	40, 32, 24, 16 kbit/s adaptive differential pulse code modulation (ADPCM)	16/24/32 /40	8	31.5/47.2/55.2/6 3.4 Kbps	ADPCM; replaces G.721 and G.723.
G.727	ITU-T	5-, 4-, 3- and 2-bit/sample embedded adaptive differential pulse code modulation (ADPCM)	var.	?	var	ADPCM. Related to G.726.
G.728	ITU-T	Coding of speech at 16 kbit/s using low-delay code excited linear prediction	16	8	31.5 Kbps	CELP. Annex J offers variable-bit rate operation for DCME.
G.729	ITU-T	Coding of speech at 8 kbit/s using conjugate-structure algebraic-code-excited linear-prediction (CS-ACELP)	8	8	31.2 Kbps	Low delay (15 ms)
GSM 06.10	ETSI	Regular Pulse Excitation Long-Term Predictor (RPE-LTP)	13	8	30.3 Kbps	Used for GSM cellular telephony.
LPC10e	US Govt.	Linear-predictive codec	2.4	8	7.8 Kbps	10 coefficients. Also known as FIPS 1015
iLBC	IETF	iLBC (internet Low Bitrate Codec) designed for narrow band speech.	13.3	8	27.7 Kbps	Frames are encoded completely independently.
Speex	N/A	Speex is based on CELP and is designed to compress voice at bitrates ranging from 2 to 44 kbps.	2.15-44.2	8/16/32	7.4 Kbps +	open-source, multirate codec

Πίνακας 3: Codecs και βασικά χαρακτηριστικά

3ο ΚΕΦΑΛΑΙΟ

Session Initiation Protocol (SIP)

3.1 Εισαγωγή

Το Session Initiation Protocol (SIP), είναι ένα πρωτόκολλο σηματοδοσίας υπεύθυνο για την αρχικοποίηση, τη διαχείρηση και το τερματισμό συνόδων (sessions) με ένα ή περισσότερους συμμετέχοντες. Ως σύνοδο (session), σύμφωνα με το RFC 2327 (SDP - Session Description Protocol), έχει οριστεί ένα σύνολων ροών δεδομένων (data streams) που περιέχουν διαφορετικούς τύπους media δεδομένων μεταξύ αποστολέα και δέκτη. Ένα session μπορεί να αποτελέσει μια απλή τηλεφωνική κλήση μέσω internet, η διανομή πολυμέσων, οι τηλεδιασκέψεις κτλ.

Το SIP σχεδιάστηκε αρχικά από τους Henning Schulzrinne και and Mark Handley το 1996. Η τελική μορφή του πρωτοκόλλου διαμορφώθηκε από την IETF το Νοέμβριο του 2000 και κατοχυρώθηκε στο RFC 3261. Σε αυτό το RFC το SIP έγινε αποδεκτό ως ένα 3GPP πρωτόκολλο σηματοδοσίας και ως βασικό κομμάτι της αρχιτεκτονικής των IMS συστημάτων (IP Multimedia Subsystems).

Ως client – server πρωτόκολλο το SIP καθορίζει πέντε διαφορετικές παραμέτρους απαραίτητες για την εγκατάσταση και το τερματισμό μιας κλήσης:

1. Τη θέση του χρήστη (User Location): προσδιορισμός των συντεταγμένων του τελικού χρήστη (IP address – port number) του συστήματος για την περαίωση της επικοινωνίας.

2. Τη διαθεσιμότητα του χρήστη (User availability): προσδιορισμός της ικανότητας του καλούντος (called party) να συμμετέχει σε μια κλήση.

3. Τις δυνατότητες του χρήστη (User capabilities): καθορισμός των μέσων που θα χρησιμοποιηθούν και των παραμέτρων τους.

4. Την εγκατάσταση ενός session (session establishment): «κουδούνισμα» και εγκατάσταση των παραμέτρων κλήσης και στις δύο πλευρές, τόσο από τη μεριά του καλούντος όσο και από τη μεριά του καλούμενου.

5. Τη δαιχείρηση ενός session (Session management): το οποίο μπορεί να περιλαμβάνει τη μεταφορά και το τερματισμό ενός session, ή ακόμα την αλλαγή των παραμέτρων ενός session.

Το SIP είναι ένα πρωτόκολλο επιπέδου εφαρμογής (Application layer protocol) το οποίο έχει σχεδιαστεί έτσι ώστε να είναι ανεξάρτητο από τα πρωτόκολλα που χρησιμοποιούνται στο επίδεδου δικτύου. Επομένως μπορεί να τρέχει πάνω από TCP (Transmission Control Protocol), UDP (User Datagram Protocol) ή SCTP (Stream Control Transmission Protocol). Ως text-based πρωτόκολλο ενσωματώνει πολλά στοιχεία από τα πρωτόκολλα HTTP και SMTP. Το SIP χρησιμοποιείτε σε συνδυασμό με τα άλλα IETF πρωτόκολλα για να παρέχει ολοκληρωμένες υπηρεσίες στους τελικούς χρήστες. Επομένως μια ολοκληρωμένη αρχιτεκτονική θα συμπεριλαμβάνει πρωτόλλα όπως το Real Time Transport Protocol (RTP)(RFC 1889) για τη μεταφορά δεδομένων πραγματικού χρόνου και τον έλεγχο του QoS της κλήσης, το Real-Time streaming protocol (RTSP)(RFC 2326) για τον έλεγχο της παράδοσης των ροών δεδομένων, το Media Gateway Control Protocol (MEGACO) (RFC 3015) για την επικοινωνία με τα gateways του δημόσιου τηλεφωνικού δικτύου, και το Session Description Protocol (SDP) για τον καθορισμό των παραμέτρων ενός session.

Το SIP δε παρέχει υπηρεσίες, αντιθέτως είναι σχεδιασμένο έτσι ώστε να παρέχει τις αρχές που χρειάζονται για το σχεδιασμό νέων υπηρεσιών. Το SIP δε παρέχει υπηρεσίες ελέγχου συνδυαλέξεων, ούτε καθορίζει τη διαχείρηση μιας συνδιάσκεψης, αλλά μπορεί να χρησιμοποιηθεί για τη αρχικοποίηση ενός session το οποίο με τη σειρά του θα χρησιμοποήσει άλλα πρωτόκολλα ελέγχου συνδιασκέψεων. Επομένως, εφόσον τα μηνύματα SIP και τα sessions που δημιουργούν μπορούν να σταλθούν μέσω τελείως διαφορετικών δικτύων, το SIP δε μπορεί και δε παρέχει δέσμευση πόρων του δικτύου.

Η φύση των υπηρεσιών που υποστηρίζονται από το SIP καθιστούν την ασφάλεια ιδαίτερα σημαντική. Για αυτό το λόγο, το SIP παρέχει ένα σύνολο υπηρεσιών ασφαλείας, όπως είναι οι υπηρεσίες παρεμπόδισης της μη πραγματοποίησης μιας

εφαρμογής (denial of service), ταυτοποίηση του χρήστη, προστασία της ακεραιότητας του και κρυπτογράφηση, και υπηρεσίες ιδιωτικότητας.

3.2 Δομή του πρωτοκόλλου

Το SIP είναι ένα πρωτόκολλο δομημένο σε επίπεδα, που σημαίνει ότι η συμπεριφορά του δομείτε από διαφορετικά στάδια επεξεργασίας τα οποία είναι αλληλένδετα μεταξύ τους. Η συμπεριφορά του πρωτοκόλλου έχει διαχωριστεί σε διαφορετικά επίπεδα επιτρέποντας την περιγραφή των κοινών λειτουργιών διαφορετικών στοιχείων σε μια κοινή βάση. Όταν λέμε ότι ένα στοιχείο (element) «ανήκει» σε ένα επίπεδο, εννούμε ότι ακολουθεί τους κανόνες που συνάδουν σε αυτό το επίπεδο. Φυσικά κάθε στοιχείο που περιγράφεται από το SIP δεν περιέχει κάθε επίπεδο. Σε αυτό το σημείο θα πρέπει να διευκρινήσουμε ότι όταν μιλάμε για διαφορετικά elements στο SIP αναφερόμαστε σε λογικά και όχι σε φυσικά elements.

Το χαμηλότερο επίπεδου του SIP είναι η σύνταξη και η κωδικοποίηση του. Η κωδικοποίηση του περιγράφετε από την γραμματική Backus-Naur Form (BNF).

Το δεύτερο επίπεδο του SIP είναι το επίπεδο μεταφοράς (transport layer). Σε αυτό το επίπεδο καθορίζεται πως ένας client στέλνει αιτήματα και λαμβάνει απαντήσεις και πως ένας server δέχεται αιτήσεις και στέλνει απαντήσεις πάνω από το IP δίκτυο. Όλα τα SIP elements περιέχουν το επίπεδο μεταφοράς στη δομή τους.

Το τρίτο επίπεδο είναι το επίπεδο συναλλαγής (transaction layer). Οι συναλλαγές είναι θεμελιώδες συστατικό του SIP. Μια συναλλαγή περιλαμβάνει όλα τα αίτηματα που θα στείλει ένας client (χρησιμοποιώνατς το επίπεδο μεταφοράς) σε ένα server και όλες τις απαντήσεις που θα στείλει ο server πίσω στον client. Το επίπεδο συναλλαγής διαχειρίζεται όλες τις αναμεταδόσεις και τα time-outs του επίπεδου εφαρμογής, ενώ συνδυάζει συγκεκριμένες απαντήσεις με συγκεκριμένα αιτήματα. Οι user agents, στους οποίους θα αναφερθούμε παρακάτω, περιέχουν επίπεδο συναλλαγής όπως και οι stateful proxies, σε αντίθεση με τους stateless proxies οι οποίοι δε περιλαμβάνουν αυτό το επίπεδο στη δομή τους. Το επίπεδο συναλλαγής έχει ένα client κομμάτι (το οποίο αποκαλείτε client transaction) και ένα server κομμάτι (το οποίο αποκαλείτε server transaction). Κάθε κομμάτι της συναλλαγής

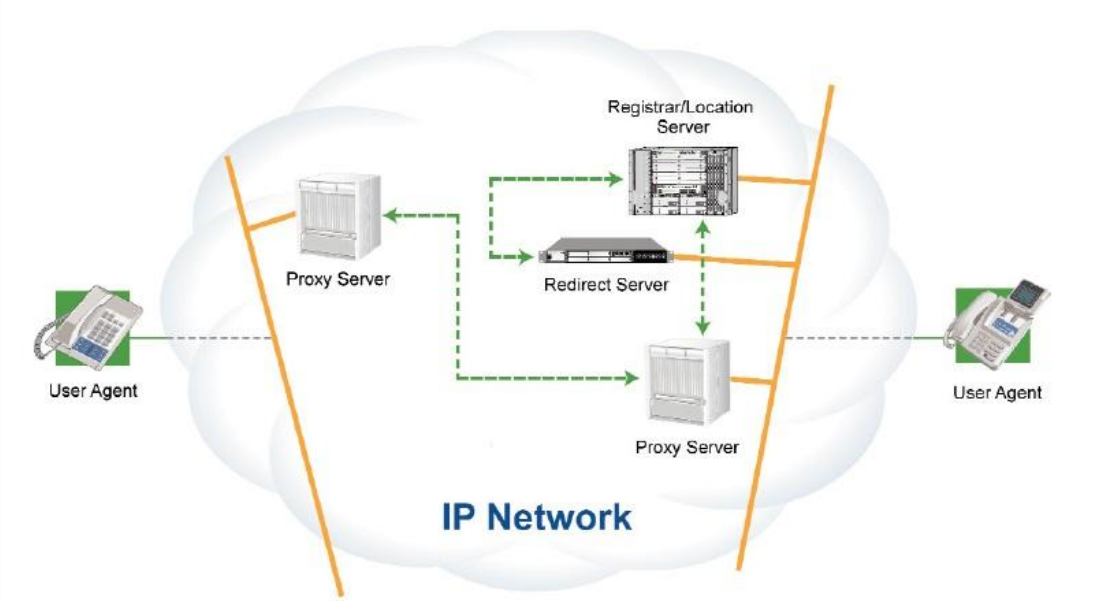
μεταξύ ενός client και ενός server αντιπροσωπεύετε από μια finite state machine (FSM) η οποία σχεδιάστηκε για τη περαίωση του κάθε αιτήματος.

Το επίπεδο πάνω από το επίπεδο συναλλαγής ονομάζεται χρήστης συναλλαγής (**TU – transaction user**). Κάθε SIP οντότητα, εκτός από τους stateless proxy, είναι χρήστης συναλλαγής. Όταν ένας χρήστης συναλλαγής θέλει να στείλει ένα αίτημα, δημιουργεί μια καινούρια συναλλαγή (**transaction**) και προωθεί το αίτημα στο τελικό προορισμό του, με βάση την IP address προορισμού, την πόρτα της εφαρμογής και το πρωτόκολλο μεταφοράς που θα χρησιμοποιηθεί. Ένας χρήστης συναλλαγής, όπως αναφέραμε, μπορεί να δημιουργήσει μια συναλλαγή (transaction) αλλά μπορεί και να την ακυρώσει. Όταν ένας client ακυρώσει μια συναλλαγή (transaction), απαιτεί από το server να σταματήσει τη περαιτέρω επεξεργασία, να επανέλθει στην κατάσταση που ήταν πριν ξεκίνησει η συναλλαγή (transaction) και να στείλει μια απάντηση σφάλματος για την εν λόγω συναλλαγή. Αυτό ακριβώς κάνει ένα CANCEL μήνυμα, το οποίο δημιουργεί ένα δικό του transaction, αλλά παραπέμπει στο transaction που πρέπει να ακυρωθεί.

Τα βασικά στοιχεία του SIP (οι user agent clients και servers, οι stateless, stateful proxies και οι registrars) περιέχουν έναν πυρήνα (core) που τα διακρίνει το ένα από το άλλο. Οι πυρήνες είναι χρήστες συναλλαγής, εκτός από τους stateless proxy. Αν και η συμπεριφορά των πυρήνων ενός user agent client ή ενός user agent server εξαρτάται από τη μέθοδο, υπάρχουν κάποιοι κοινοί κανόνες για όλες τις μεθόδους. Για έναν UAC, αυτοί οι κανόνες καθορίζουν τη κατασκευή ενός αιτήματος; για ένα UAS, οι κανόνες διέπουν την επεξεργασία ενός αιτήματος και τη δημιουργία μιας απάντησης. Από τη στιγμή που οι καταχωρήσεις (registrations) παίζουν σημαντικό ρόλο στο SIP, οι UAS που διαχειρίζονται τα REGISTER αιτήματα, ονομάζονται registrar. Διάφορα αιτήματα μπορούν να σταλούν μέσα σε ένα παράθυρο διαλόγου (dialog). Ένα παράθυρο διαλόγου είναι μια από άκρο σε άκρο SIP συσχέτιση μεταξύ δύο user agents που διαρκεί για κάποιο χρονικό διάστημα. Ένα παράθυρο διαλόγου (dialog) διευκολύνει την αλληλουχία των μηνυμάτων και την ορθή δρομολόγηση των αιτημάτων μεταξύ δύο user agents. Τα INVITE αιτήματα είναι εκείνα που θεσπίζουν ένα dialog. Όταν ένας UAC στέλνει ένα αίτημα που εντάσσεται στο πλαίσιο ενός dialog, ακολουθεί τους απλούς κανόνες του UAC, αλλά και τους κανόνες για τα αιτήματα ενδιάμεσου dialog. Στα dialogs θα αναφερθούμε πιο αναλυτικά παρακάτω.

3.3 Βασικά συστατικά ενός SIP δικτύου

Αν και το πιο απλό configuration ενός SIP δικτύου αποτελείτε από δύο User agents που ανταλλάσουν SIP μηνύματα, ένα τυπικό SIP δίκτυο περιλαμβάνει περισσότερα από ένα SIP elements. Τα βασικότερα SIP elements είναι οι user agents, οι proxies, οι registrars, και οι redirect servers τα οποία είναι υπεύθυνα για την εγκατάσταση, τη δρομολόγηση, τη διευθυνσιοδότηση, τη μετάφραση των τηλεφωνικών αριθμών κτλ. . Στο παρακάτω σχήμα απεικονίζετε ένα τυπικό SIP δίκτυο.



Σχήμα 9: Απεικόνιση ένος τυπικού VOIP δικτύου

3.3.1 User Agents

Ένας SIP User Agent είναι μια λογική οντότητα σε ένα SIP endpoint η οποία μπορεί να δημιουργήσει και να λάβει SIP μηνύματα και επομένως μπορεί να διαχειριστεί ένα SIP session. Επίσης, SIP User Agent αποτελεί το λογισμικό που τρέχει στις συσκευές των τελικών χρηστών, όπως τα cell phones, οι μικροσυσκευές πολυμέσων, τα PCs και τα PDAs, κατά την διάρκεια εγκατάστασης μιας VoIP κλήσης. Ένας User Agent αποτελείτε από ένα User Agent Client ο οποίος στέλνει ένα request και ένα User Agent Server ο οποίος δέχεται το request και απαντά με ένα response. Όταν ένας UAC στέλνει ένα request, το request διαπερνά έναν αριθμό από proxy servers, οι οποίοι το προωθούν στον UAS. Και από την άλλη πάλι όταν ένας UAS παράγει ένα

response αυτό προωθείτε στον UAC. Η συμπεριφορά τόσο των UAC όσο και των UAS εξαρτάται από δύο κυρίως παράγοντες: από το αν το request ή το response είναι εντός ή εκτός ενός dialog, και από τη μέθοδο του request.

Ανάλογα με τις ικανότητες που παρέχουν οι SIP User Agents χωρίζονται σε επιμέρους κατηγορίες:

- **Basic**: Μια βασική εφαρμογή η οποία δίνει τη δυνατότητα σε ένα BYE να ολοκληρώσει μια κλήση που εκρεμεί.
- **Redirection**: Για την υποστήριξη της προώθησης μιας κλήσης σε ένα άλλο τερματικό (Call Forwarding). Ένας client πρέπει να είναι σε θέση να αναγνωρίζει έναν contact header, αλλά μόνο το SIP-URL κομμάτι και όχι τις λοιπές παραμέτρους.
- **Firewall-Friendly**: Ένας firewall-friendly client αντιλαμβάνεται τους Route και Record-Route headers και μπορεί να σεταριστεί με τέτοιο τρόπο ώστε να χρησιμοποιεί ένα τοπικό proxy για όλα τα outgoing requests.
- **Negotiation**: Ένας client πρέπει να είναι σε θέση να καταλαβαίνει και να αναγνωρίζει το 380 Status code (alternative service) και τις παραμέτρους του contact header, έτσι ώστε να μπορεί να συμμετέχει σε ένα terminal και media negotiation. Πρέπει επίσης να είναι σε θέση να σκανάρει ένα Warning response header, ώστε να δίνει χρήσιμη πληροφορία στον καλούντα.
- **Authentication**: Εάν ένας client επιθυμεί να καλέσει ένα συνδρομητή ο οποίος προαπαιτεί call authentication, πρέπει να μπορεί να αναγνωρίζει το 401 Unauthorized status code, να μπορεί να δημιουργήσει ένα Authorization request header και να καταλαβαίνει έναν www-Authenticate response header.

3.**3.2 SIP Proxy Server**

Οι SIP Proxies είναι τα τμήματα εκείνα ενός SIP δικτύου τα οποία δρομολογούν τα requests προς τους user agent servers και τα SIP responses στους User agent clients. Ένα request μπορεί να διαπεράσει πολλούς proxies μέχρι να καταλήξει σε ένα UAS. Έτσι όταν ένας SIP Proxy Server λάβει ένα τέτοιο SIP request από ένα χρήστη, αρχικά επικοινωνεί με τον SIP Registrar Server για να πάρει πληροφορία για την τρέχουσα θέση του χρήστη που καλείται. Στην περίπτωση που ο χρήστης αυτός εντοπιστεί τότε το SIP request προωθείται στον UAS του χρήστη, διαφορετικά

προωθείται στον επόμενο SIP Proxy Server. Κάθε SIP proxy θα πάρει αποφάσεις σχετικά με το πώς θα δρομολογηθεί ένα request και θα τροποποιήσει το request πριν το στείλει στον επόμενο παραλήπτη, είτε αυτός είναι ένας άλλος proxy είτε ένα UA. Τα responses για το συγκεκριμένο request θα δρομολογηθούν μέσω των ίδιων proxies προς τον UAC. Ένας proxy μπορεί να λειτουργεί σαν stateful ή stateless proxy.

3.3.2.1 Stateless Proxy Server

Η δουλειά ενός stateless proxy περιορίζεται στο να προωθεί τα SIP μηνύματα στο επόμενο Sip element, δρομολογώντας τα με βάση την πληροφορία που περιέχετε στους headers του request. Ένας stateless proxy δε αποθηκεύει πληροφορία σχετική με το request που έχει λάβει, όταν αυτό προωθηθεί στο επόμενο SIP element. Αν και τα SIP μηνύματα ομαδοποιούνται σε transactions, οι stateless proxies δε μπορούν να διαχειριστούν ένα transaction. Ένα από τα κύρια μειονεκτήματα των stateless proxies είναι ότι αδυνατούν να απορροφήσουν τις αναμεταδόσεις των μηνυμάτων και να εκτελέσουν πιο προηγμένη δρομολόγηση, για παράδειγμα forking ή επαναλαμβανόμενη αποστολή. Οι stateless proxies είναι γενικά πιο απλοί στη λειτουργία τους, αλλά πολύ πιο γρήγοροι από τους stateful proxies και χρησιμοποιούνται κυρίως σαν απλοί load balancers, μεταφραστές μηνυμάτων και routers.

3.3.2.1 Stateful Proxy Server

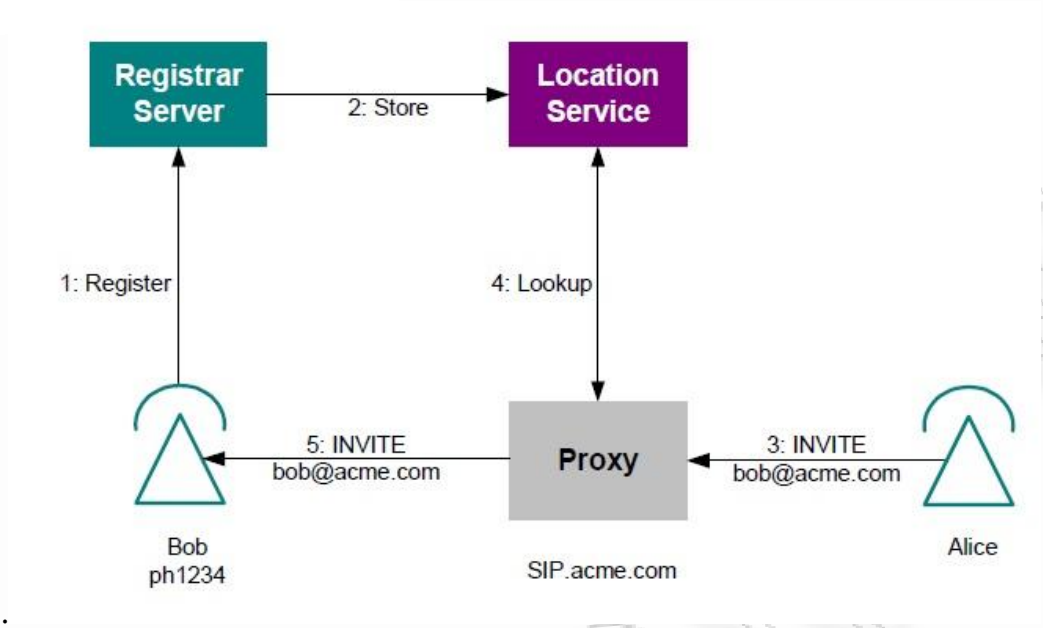
Οι stateful proxies είναι πολύ πιο πολύπλοκοι από τους stateless proxies. Από τη στιγμή που θα λάβουν ένα request, δημιουργούν ένα state το οποίο διατηρούν μέχρι να ολοκληρωθεί ένα transaction. Μερικά transactions, και κυρίως όσα δημιουργούνται από INVITE requests, μπορεί να έχουν μεγάλη διάρκεια (μέχρι ο καλούμενος να απαντήσει ή να απορρίψει την κλήση). Ακριβώς επειδή πρέπει να διατηρήσουν το συγκεκριμένο state για όλη τη διάρκεια του transaction, η απόδοση τους είναι περιορισμένη. Από την άλλη πάλι η ικανότητα τους να συσχετίζουν SIP μηνύματα τα οποία ανήκουν στο ίδιο transaction τους δίνει πολλά ενδιαφέροντα χαρακτηριστικά.

- Οι stateful proxies μπορούν να υποστηρίξουν τη δυντατότητα του forking, που σημαίνει ότι κατά τη λήψη ενός μηνύματος δύο ή περισσότερα μηνύματα θα σταλθούν σε πολλαπλούς αποδέκτες.

- Οι stateful proxies μπορούν να απορρίψουν αναμεταδόσεις μηνυμάτων, επειδή μπορούν από το transaction state να αναγνωρίσουν αν έχουν ξαναλάβει ένα μήνυμα ή όχι. (Οι stateless proxies λόγω της αδυναμίας τους να αποθηκεύουν τα μηνύματα στη μνήμη τους δε μπορούν να ξεχωρίσουν αν ένα μήνυμα είναι αναμετάδοση κάποιου άλλου ή όχι).
- Οι stateful proxies μπορούν να εκτελέσουν πιο περίπλοκες μεθόδους για την εύρεση ενός χρήστη. Για παράδειγμα, είναι πιθανό να προσπαθούμε να καλέσουμε κάποιον στο τηλέφωνο του γραφείου του και επειδή δε απαντάει η κλήση ανακατευθύνεται στο κινητό του. Οι stateless proxies δε μπορούν να κάνουν κάτι αντίστοιχο γιατί δε μπορούν να ξέρουν αν ολοκληρώθηκε και πότε το transaction προς το τηλέφωνο του γραφείου.
- Υποστήριξη πολύπλοκων δικτυακών λειτουργιών, όπως NAT traversal.

3.3.3 SIP Registrar Server

Ένας SIP registrar είναι ένας server σε ένα SIP δίκτυο, ο οποίος δέχετε και επεξεργάζετε SIP Register requests. Από τα requests που δέχετε ένας registrar, εξάγει πληροφορίες για την τρέχουσα θέση ενός συνδρομητή (IP διεύθυνση, port number και όνομα χρήστη) και δημιουργεί εγγραφές που τις αποθηκεύει σε μια βάση δεδομένων. Έτσι για παράδειγμα, η λογική διεύθυνση sip:user@sipserver.com του χρήστη user, αντιστοιχεί στη φυσική διεύθυνση sip:user@10.85.150.152, η οποία περιέχει πληροφορία για την τρέχουσα θέση του. Αυτό ονομάζεται “location service” και είναι ουσιαστικά μια βάση δεδομένων την οποία χρησιμοποιεί ο registrar έτσι ώστε να αποθηκεύει και να ανακτά πληροφορία για τη θέση ενός χρήστη. Το location service μπορεί να τρέχει σε ένα άλλο μηχάνημα με το οποίο δύναται να επικοινωνεί μέσω ενός πρωτοκόλλου, όπως το LDAP. Το SIP αφήνει την επιλογή για το αν το location service και ο registrar θα τρέχουν στο ίδιο μηχάνημα στην εκάστωτε εφαρμογή. Επομένως, εάν ένας συνδρομητής επιθυμεί να ξεκινήσει ένα session με έναν άλλο συνδρομητή, το SIP πρέπει να ανακαλύψει τους τρέχοντες hosts στους οποίους είναι προσβάσιμος ο τερματικός χρήστης. Τέλος, ένας registrar server μπορεί να επικυρώνει ένα REGISTER request χρησιμοποιώντας ένα 401 (Unauthorized) response.



.

Σχήμα 10: Registrar server functionality.

3.3.4 Location Server

Ένας location server χρησιμοποιείτε για να αποθηκεύει διευθύνσεις, τις οποίες λαμβάνει από τα registration requests. Πρέπει να σημειώσουμε ότι όλοι οι servers στους οποίους αναφερόμαστε μπορεί να αποτελούν ανεξάρτητες οντότητες ή να συνδυάζονται σε μια κοινή φυσική οντότητα.

3.3.5 Application Server

Οι Application Servers, είναι servers οι οποίοι παρέχουν βελτιωμένες υπηρεσίες σε ένα SIP περιβάλλον.

3.3.6 SIP Redirect Server

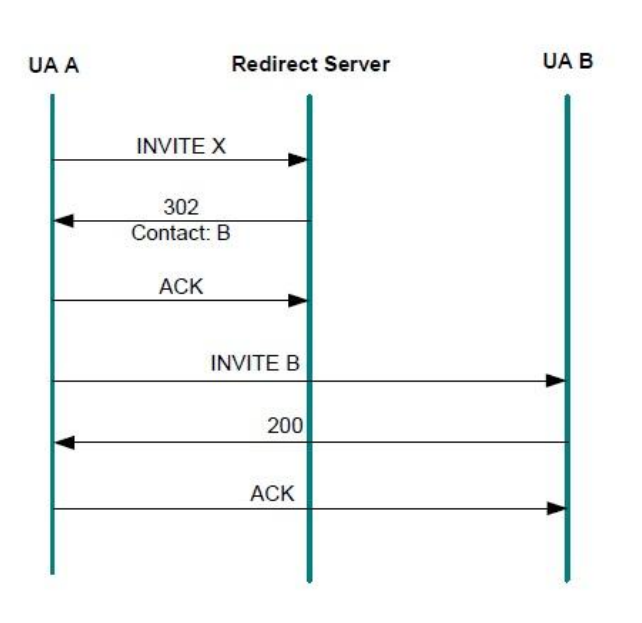
Η λειτουργία ενός Redirect server είναι αρκετά απλή. Ένας redirect server δέχεται SIP requests και απαντά με 3XX redirect responses, κατευθύνοντας τον client να επικοινωνήσει με ένα αλλό σετ από SIP addresses. Οι εναλλακτικές addresses επιστρέφονται σαν Contact headers στο response message.

3XX RESPONSES	Επεξήγηση
300 Multiple Choices	Η διεύθυνση στο request αναλύεται σε πολλαπλές επιλογές, καθεμία με το δικό της location. Ο UA μπορεί να επιλέξει το location που θέλει και να αναδρομολογήσει το request του προς αυτό.
301 Moved Permanently	Ο χρήστης δε είναι πλεόν διαθέσιμος στη διεύθυνση που δηλώνεται στο Request-URI και ο client θα πρέπει να ξαναπροσπαθήσει στέλνοντας ένα request στην καινούρια διεύθυνση η οποία υπάρχει στο contact header.
302 Moved Temporarily	Ο χρήστης είναι προσωρινά διαθέσιμος σε διαφορετική διεύθυνση(ή διευθύνσεις). Η διάρκεια ισχύος των διευθύνσεων αυτών μπορεί να δηλωθεί στον Contact header και ο client μπορεί να ξαναπροσπαθήσει να στείλει το request του στην νέα διεύθυνση.
305 Use Proxy	Η ζητούμενη διεύθυνση προορισμού θα είναι προσβάσιμη μέσω ενός proxy, ο οποίος καθορίζεται στον Contact header.
380 Alternative Service	Αν και η κλήση ήταν ανεπιτυχής, εναλλακτικές υπηρεσίες είναι διαθέσιμες. Οι υπηρεσίες αυτές περιγράφονται στο message body του response.

Πίνακας 4: 3xx responses τα οποία ορίζονται στο πρωτόκολλο SIP.

Το παρακάτω σενάριο παρουσιάζει ένα σενάριο ανακατεύθυνσης ενός request. Πρέπει να σημειώσουμε ότι το δεύτερο INVITE request δημιουργείτε με τους ίδιους

dialog identifiers, Call-ID, To και From headers όπως η αρχική INVITE, αλλά με διαφορετικό CSeq. Η ανακατεύθυνση ενός request, επιτρέπει στους servers να στέλνουν πληροφορία δρομολόγησης για το request που έλαβαν, αποφεύγοντας την επανάληψη αποστολής περαιτέρω μηνυμάτων για το συγκεκριμένο transaction. Οι redirect servers συνήθως δε έχουν επίγνωση της κάταστασης ενός dialog (κλήσεις, συνδρομές), παρα μόνο της κατάστασης του συγκεκρμένου transaction που διαχειρίζονται. Για αυτό ακριβώς το λόγο χαρακτηρίζονται και transaction-stateful elements. Η αναδρομολόγηση των requests είναι μια απλή και γρήγορη διαδικασία, η οποία επιτρέπει στους redirect servers να είναι ιδιαίτερα ευέλικτοι και με υψηλή απόδοση. Τέλος μπορούν να χρησιμοποιηθούν σαν load balancers, ενώ μπορεί να απαιτούν αυθεντικοποίηση του χρήστη χρησιμοποιώντας το 401 UnAuthorised response (Proxy Authentication Required).



Σχήμα 11: Redirect server scenario

3.4 SIP Messages

Από τη στιγμή που το πρωτόκολλο SIP σχεδιάστηκε με βάση το μοντέλο request/response, υπάρχουν δύο τύποι SIP μηνυμάτων, τα requests (ή methods) και τα SIP responses. Έτσι στην ενότητα αυτή θα ασχοληθούμε με τα είδη των SIP μηνυμάτων, τη σύνταξή τους, καθώς και τα πεδία (header fields) που τα αποτελούν.

Methods:

Όπως ορίζεται στο RFC 3261, ένα request (ή μια μέθοδο) είναι ένα SIP μήνυμα το οποίο στέλνει ένας client σε ένα sever, με απότερο σκοπό την πραγματοποίηση μιας λειτουργίας. Υπάρχουν έξι μέθοδοι οι οποίοι καθορίζονται στο RFC 3261 – INVITE, REGISTER, BYE, ACK, CANCEL και OPTIONS. Άλλα RFC's έχουν καθορίσει άλλες μεθόδους όπως: Refer, Subscribe, Notify, Publish, Message Update, Info and PRACK.

Responses:

Κατά RFC 3261 ένα response είναι ένα SIP μήνυμα που στέλνει ένας server σε έναν client για να καθορίσει το status του request που έστειλε ο client στο Server. Τα responses διαχωρίζονται σε έξι κλάσεις, οι οποίες καθορίζουν τη γενική κατηγορία του response. Στο παρακάτω πίνακα απεικονίζονται οι έξι διαφορετικές response κλάσεις.

Response Class	Description
1XX	Informational
2XX	Success
3XX	Redirection
4XX	Client Error
5XX	Server Error
6XX	Global Error

Πίνακας 5: Response classes

3.4.1 SIP Addressing

Οι SIP οντότητες διαχωρίζονται από τον SIP Uniform Resource Identifier (URI) όπως έχουμε αναφέρει και σε προηγούμενο κεφάλαιο. Ένα SIP URI μπορεί να καθοριστεί ως το τηλεφωνικό νούμερο ενός χρήστη, ή η IP διεύθυνσή του μέσα στο SIP δίκτυο. Αυτά τα URI's έχουν παρόμοιο φορμάτ με τις email διευθύνσεις όπως παραδείγματος χάριν:

Mail to: j.smith@nowhere.com.

SIP Addressing:

- sip:jane.doe@192.168.1.102
- sip:support@help.me.com
- sip:22444032@mynumber.is.com

Ένα SIP URI, ξεκινά με ένα scheme, το οποίο όπως βλέπουμε και στα προηγούμενα παραδείγματα έναι το 'sip'. Αυτό που ακολουθεί είναι μια έγκυρη SIP διεύθυνση, η οποία μπορεί να είναι ένα host name ή μια IP διεύθυνση. Ωστόσο το γενικό φορμάτ ενός URI μπορεί να περιέχει και άλλες παραμέτρους οι οποίες είναι προεραιτικές. Γενικά ένα URI μπορεί να έχει την παρακάτω δομή:

sip:user:password@host:port;uri-parameters?headers

3.4.2 SIP Message Structure

3.4.2.1 Request Message Format

Όπως έχουμε ήδη αναφέρει το SIP είναι ένα text-based πρωτόκολλο το οποίο αναπτύχθηκε από την IETF. Στο αρχικό design του πρωτοκόλλου χρησιμοποιήθηκαν τεχνικές άλλων text-based πρωτοκόλλων όπως το HyperText Transport Protocol (HTTP) και το Simple Mail Transport Protocol (SMTP). Το SIP χρησιμοποιεί το ίδιο σχήμα διευθυνσιοδότησης με το HTTP και την ίδια λογική χρήσης των headers με το SMTP. Ένα SIP μήνυμα αποτελείτε από τρία βασικά τμήματα, το Request line, το Header section και το Message body.

- **Request Line**

  Το Request Line αποτελείτε από το request type, το SIP URI του προορισμού ή του next hop και το version του SIP που χρησιμοποιείτε. Από το παράδειγμα που ακολουθεί παρατηρούμε ότι το request είναι ένα INVITE message, με SIP Uri sip:9103682854@192.168.16.140, ενώ το SIP version που χρησιμοποιείτε είναι το 2.0.

- **Header Section**

Το header section αποτελείτε από πολλαπλούς headers, καθένας από τους οποίους περιέχει τη δική του καλώς ορισμένη πληροφορία. Κάθε header διαχωρίζεται από ένα χαρακτήρα <cr><lf> στο τέλος του header. Θα αναφερθούμε αναλυτικά στους headers ενός SIP μηνύματος παρακάτω.

- **Message body**

Το τελικό τμήμα ενός request, το message body, είναι προαιρετικό και εξαρτάται από τον τύπο του μηνύματος και από το που εμπίπτει στη διαδίκασια εγκατάστασης μιας κλήσης. Τα όρια μεταξύ του header section και του message body διαχωρίζονται από μια κενή γραμμή. Το παρακάτω μήνυμα περιέχει και SDP (Session Description Protocol). Το SDP κομμάτι του μηνύματος έχει μήκος 322 χαρακτήρες. Αυτή η πληροφορία περιέχεται στον Content-Type header (application/sdp) και στον Content Length header (322). Το SDP που περιέχεται σε μια INVITE καθορίζει πληροφορία σχετική με τα media που είναι απαραίτητη για το συγκεκριμένο session. Πιο αναλυτικά θα αναφερθούμε στο SDP σε παρακάτω ενότητα.

Request Line

INVITE sip:9103682854@192.168.16.140 SIP/2.0

Headers

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z

Instant Messaging and Presence ProtocolMax-Forwards: 70

Contact: <sip:9103683957@192.168.16.105:44646

To: "Jane Doe"<sip:9103682854@192.168.16.140

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg

CSeq: 2 INVITE

Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO

Record-Route: <sip:192.168.16.140;lr>

Content-Type: application/sdp

Content-Length: 322

**************************Empty Line**************************

Message Body

v=0

o=- 3 2 IN IP4 192.168.16.105

s=CounterPath X-Lite 3.0

c=IN IP4 192.168.16.105

t=0 0

m=audio 32854 RTP/AVP 107 0 8 101

a=alt:1 2 : oND7cBBb qhQukSDp 10.100.100.250 32854

a=alt:2 1 : napYhQOC H731+lWt 192.168.16.105 32854

a=fmtp:101 0-15

a=rtpmap:107 BV32/16000

a=rtpmap:101 telephone-event/8000

a=sendrecv

3.4.2.2 Response Message Format

Όπως ένα SIP request έτσι και ένα SIP response αποτελείτε από τρία βασικά κομμάτια: το Status line, το Header section και το Message body. Σε μερικές περιπτώσεις ένα Response είναι λίγο μεγαλύτερο από ένα Request γιατί περιέχει ένα reason header που δείχνει γιατί ένα Response έχει αποσταλεί.

- **Status line**

  Το Status line αποτελείτε από τρία elements: το protocol version, το status code και το reason phrase. Μόνο τα δύο πρώτα elements επεξεργάζονται από οποιοδήποτε SIP network element. Το reason phrase κάνει απλά ένα response κατανοητό σε ένα user. Στο παράδειγμα που ακολουθεί το protocol version είναι 2.0. Το status code είναι 100 και δηλώνει ότι το response είναι μέρος μιας

1XX response κλάσης και πιο συγκεκριμένα ότι πρόκειτε για ένα 100 response. Ένα 100 response χρησιμοποείτε για να ενημερώσει τον αποδέκτη ότι το request έχει ληφθεί από ένα άλλο device, και ότι μπορεί να χρειαστεί κάποιο χρονικό διάστημα μέχρι να εγκατασταθεί αυτό το session. Το τελικό κομμάτι του Status line είναι το reason phrase “Trying.”

- **Header Section**

  Το header section περιλαμβάνει πολλαπλούς headers, καθένας από τους οποίους κουβαλάει τη δική του καλώς ορισμένη πληροφορία. Κάθε header διαχωρίζεται από τον επόμενο από ένα carriage return (CRLF), στο τέλος κάθε header.

- **Message Body**

  Η κατάληξη ενός response, το message body, είναι προαιρετικό και συνήθως εξαρτάται από το είδος του μηνύματος. Τα όρια μεταξύ του header section και του message body, διαχωρίζονται από μια κενή γραμμή. Το μήνυμα που ακολουθεί δε περιέχει message body, εφόσον η τιμή του Content-Length Header είναι 0.

Status Line
SIP/2.0 100 Trying
Headers
Via: SIP/2.0/UDP 192.168.16.140:5060;branch=z9hG4bK-85z25d58d7461b9
To: "Jane Doe"<sip:9103682854@192.168.16.140>
From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d
Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg
CSeq: 2 INVITE
Content-Length: 0
**********Empty Line**********
Message Body

3.4.2.3 Πεδία επικεφαλίδας (header fields)

Τα πεδία της επικεφαλίδας περιέχουν πληροφορίες για τους συμμετέχοντες στη συνομιλία, τη διαδρομή του μηνύματος κ.α. Τα πεδία αυτά μοιάζουν με τα αντίστοιχα του HTTP τόσο σε σύνταξη όσο και σε σημασιολογία. Η σειρά των πεδίων γενικά δεν έχει σημασία με εξαίρεση τα πεδία που αλλάζουν από βήμα σε βήμα (hop-to-hop). Αυτά πρέπει να εμφανίζονται πριν από όλα τα αλλά πεδία που έχουν σταθερές τιμές σε όλη τη διαδρομή (end-to-end). Κάποια πεδία χρησιμοποιούνται σε όλα τα μηνύματα ενώ άλλα μόνο όταν είναι απαραίτητα. Μια εφαρμογή SIP δε χρειάζεται να μπορεί να ερμηνεύσει όλα τα πεδία της επικεφαλίδας, αν και κάτι τέτοιο θα ήταν επιθυμητό. Αν η εφαρμογή δεν μπορεί να καταλάβει ένα πεδίο το αγνοεί. Συνολικά υπάρχουν 37 πεδία που χωρίζονται σε τέσσερις κατηγόριες.

1) πεδία που χρησιμοποιούνται και στους δυο τύπους μηνυμάτων.
2) πεδία που ορίζουν το περιεχόμενο του μηνύματος (entity header fields).
3) πεδία που μεταφέρουν επιπλέον πληροφορίες για το αίτημα και τον αποστολέα του (request header fields).
4) πεδία που δίνουν πληροφορίες στον πελάτη για το διακομιστή και για την παραπέρα πρόσβαση στον πόρο που περιγράφεται στο request-uri.

Στο παρακάτω πίνακα βλέπουμε όλους τους headers που συναντάμαι σε ένα SIP μήνυμα:

Request-headers	**General-headers**	**Entity-headers**	**Response-headers**
Accept	Authorization	Content-Encoding	Allow
Accept-Encoding	Contact	Content-Length	Proxy-Authenticate
Accept-Language	Hide	Content-Type	Retry-After
Call-Id	Max-Forwards	Server	
Contact	Organization	Unsupported	
Case	Priority	Warning	

Date	Proxy-Authorization	WWW-Authenticate
Encryption	Proxy-Require	
Expires	Route	
From	Require	
Record-Route	Response-Key	
Timestamp	Subject	
To	User-Agent	
Via		

Πίνακας 6: Header Fields

3.4.2.4 Σημαντικότεροι headers ενός SIP μηνύματος

Επιλεκτικά θα αναφερθούμε σε κάποιους από τους πιο σημαντικούς headers ενός SIP μηνύματος. Πιο συγκεκριμένα θα μιλήσουμε για τους εξής:

- **Via Header**

  Ένας Via header μεταφέρει πληροφορία σχετική με το πρωτόκολλο μεταφοράς ενός request και την τρέχουσα διεύθυνση του originator. Ένας Via Header αποτελείται από τα εξής στοιχεία:

  - Την έκδοση του πρωτοκόλλου SIP (SIP/2.0).
  - Το πρωτόκολλο μεταφοράς των μηνυμάτων (UDP/TCP).
  - Την IP διεύθυνση ή το Domain Name του αποστολέα του SIP request.
  - Την παράμετρο branch, η οποία χρησιμοποιείται για την αναγνώριση του SIP Transaction που σχετίζεται το SIP request.

Η παράμετρος branch πρέπει να είναι μοναδική για κάθε request που στέλνει ένας UAC. Εξέραιση αποτελούν οι CANCEL και ACK requests. Ένα CANCEL request πρέπει να έχει την ίδια branch παράμετρο με την αντίστοιχη παράμετρο του request που «ακυρώνει», ενώ η παράμετρος branch ενός ACK request για ένα 2XX response είναι ίδια με την branch παράμετρο του αρχικού INVITE request στο οποίο αντιστοιχεί. Για να υπάρχει συμβατότητα μεταξύ του RFC 3261 και παλαιότερων RFCs, η παράμετρος branch πρέπει να ξεκινά με τους χαρακτήρες z9hG4bK. Στην πιο συνηθισμένη περίπτωση, όπου μια SIP κλήση εγκαθίσταται έμμεσα μέσω ενός ή περισσότερων SIP Proxy Servers, κάθε server προσθέτει μία Via επικεφαλίδα στο SIP request που λαμβάνει πριν το προωθήσει σε επόμενο server ή στον τελικό παραλήπτη.

- **Max Forward header**

Ο Max Forward header είναι ένας υποχρεωτικός header που χρησιμοποιείται για την αποφυγή του φαινομένου routing loops, όταν η εγκατάσταση μιας σύνδεσης πραγματοποιείται μέσω SIP Proxy Servers. Έτσι όταν λέμε ότι ο Max Forward header έχει τιμή 70 εννοούμε ότι το μήνυμα μπορεί να διέλθει το πολύ από 70 SIP Proxy Servers πριν παραδοθεί στον τελικό παραλήπτη. Κάθε SIP Proxy server, που λαμβάνει και προωθεί ένα SIP μήνυμα μειώνει κατά ένα την τιμή του Max Forwards header.

- **From Header**

Ο From header είναι ένας από τους πιο βασικούς header ενός SIP μηνύματος και προσδιορίζει τη λογική οντότητα, αλλά όχι την τρέχουσα φυσική διεύθυνση του χρήστη που αποστέλλει το SIP request. Στην περίπτωση ενός SIP response, το From Header προσδιορίζει και πάλι τη λογική οντότητα του χρήστη που έστειλε το request, μιας και ο From Header ενός SIP response αντιγράφεται από το αντίστοιχο request. Ένας From Header αποτελείται από τα εξής στοιχεία:

 Το Display Name, το οποίο μπορεί να εμφανίζεται μεταξύ double quotes και είναι προαιρετικό.

 Το SIP URI το οποίο προσδιορίζει τη λογική οντότητα του αποστολέα του request. Το URI περικλείεται από τους χαρακτήρες < και >, εκτός αν δεν χρησιμοποιείται το προαιρετικό Display Name.

 Την παράμετρο tag η οποία είναι υποχρεωτική και μοναδική για κάθε SIP μήνυμα που δημιουργείται στα πλαίσια ενός SIP dialog. Η παράμετρος tag είναι πολύ σημαντική παράμετρος διότι χρησιμοποιείται για την αντιστοίχηση των SIP μηνυμάτων με τα existing dialogs, στα οποία θα αναφερθούμε σε επόμενη ενότητα.

- **To Header**

  Ο To header προσδιορίζει την λογική οντότητα του αποδέκτη του request καιι χρησιμοποιεί την ίδια σύνταξη με τον From header. Η μοναδική διαφορά έγκειτε στο γεγονός ότι η παράμετρος tag μπορεί να λείπει από την αρχική INVITE μιας κλήσης, και θα προστεθεί από τον αποδέκτη στο αρχικό provisional response που θα αποστείλει.

- **Call-ID Header**

  Ο Call-ID Header είναι υποχρεωτικός header για όλα τα requests και responses και χρησιμοποιείτε για να διαχωρίζει τα μηνύματα που σχετίζονται με μια συγκεκριμένη κλήση μεταξύ δύο τελικών χρηστών. Ο Call ID header είναι χαρακτηριστικός για κάθε κλήση και αποτελείται από ένα μοναδικό αριθμό και την IP διεύθυνση ή το πλήρες DNS host name του UA που δημιουργεί το request. Ο συνδυασμός του Call-ID header και των παραμέτρων tag των From και To επικεφαλίδων καθορίζουν πλήρως ένα SIP dialog.

- **CSeq Header**

  Ο Cseq header είναι ένας υποχρεωτικός header που χρησιμεύει ως προσδιοριστικό για την αντιστοίχιση requests και responses που ανήκουν σε ένα μοναδικό SIP Transaction. Ο CSeq header αποτελείται από ένα τυχαίο αριθμό και το όνομα της μεθόδου για το συγκεκριμένο request. Το αριθμητικό μέρος του header πρέπει να αυξάνεται κατά ένα, κάθε φορά που αποστέλλεται ένα request που ανήκει σε ένα υπάρχον dialog. Εξαίρεση αποτελούν τα ACK

και CANCEL requests, όπου το αριθμητικό μέρος παραμένει ίδιο με αυτό του αντίστοιχου request.

- **Contact Header**

  Ο Contact Header αν και δε είναι ένας από τους υποχρεωτικούς headers ενός SIP μηνύματος, περιέχει πληροφορία σχετική με τη SIP address του χρήστη ο οποίος έστειλε το μήνυμα. Ο Contact Header έχει τη μορφή URI και έχει την ίδια σύνταξη με τον FROM header με τη διαφορά ότι δεν περιέχει την παράμετρο tag. Ωστόσο άλλες παράμετροι μπορεί να συμπληρώνουν το συγκεκριμένο header, όπως η παράμετρος expires που δηλώνει το χρονικό διάστημα (σε seconds) που η Contact διεύθυνση είναι valid.

- **Συμπληρωματικοί headers**

  Εκτός από τους παραπάνω headers υπάρχουν και άλλοι οι οποίοι ολοκληρώνουν ένα SIP μήνυμα. Για παράδειγμα ένας ALLOW header προσδιορίζει όλες τις μεθόδους που υποστηρίζονται από τον αποστολέα του μηνύματος. Επίσης ένας RECORD-ROUTE header χρησιμοποιείτε από έναν proxy για να προωθήσει τη δρομολόγηση των μηνυμάτων που ανήκουν σε ένα session μέσω ενός συγκεκριμένου proxy. Άλλοι σημαντικοί headers είναι οι Content-Type και Content-Length οι οποίοι προσδιορίζουν τον τύπο της πληροφορίας που μεταφέρεται στο message body και το μέγεθος του σε octets. Ωστόσο υπάρχουν πολύ ακόμη headers σε ένα SIP μήνυμα, οι οποίοι έχουν συγκεκριμένο ρόλο και περιγράφονται αναλυτικά στα αντίστοιχα RFC's.

3.4.3 SIP Requests

Όπως έχουμε ήδη αναφέρει το RFC 3261 καθορίζει έξι SIP μεθόδους, ένω άλλες συμπληρωματικές μέθοδοι περιγράφονται σε άλλα RFCs. Στο παρακάτω πίνακα περιγράφονται τα requests που υποστηρίζονται από το πρωτόκολλο SIP:

Method	**Use**

ACK (RFC 3261)	Η μέθοδος ACK χρησιμοποιείτε ως acknowledgment σε ένα final response ενός INVITE request. Final responses είναι τα responses κλάσης 2xx, 3xx, 4xx, and 5xx
Bye (RFC 3261)	Μια μέθοδος Bye χρησιμοποιείτε για τερματίσει ένα established media session.
Cancel (RFC 3261)	Η μέθοδος CANCEL χρησιμοποιείτε για να τερματίσει ένα session πριν αυτό εγκατασταθεί.
Info (RFC 3261)	Μια μέθοδος INFO χρησιμοποιείτε για να μεταφέρει πληροφορία σχετική με τη σηματοδοσία μιας κλήσης, από έναν user agent σε έναν άλλο user agent, με τον οποίο έχει εγκαταστήσει ένα media session.
Invite (RFC 3261)	Μια INVITE χρησιμοποιείτε από ένα user agent για την εγκατάσταση ενός session με έναν άλλο user agent
Message (RFC 3428)	Η μέθοδος Message χρησιμοποιείτε για να μεταφέρει πληροφορία Instant messaging στο πρωτόκολλο SIP.
Notify (RFC 3265)	Η μέθοδος Notify χρησιμοποιείτε για να περέχει updated events και status information που απαιτούνται από ένα Subscribe request.

Options (RFC 3261)	Η μέθοδος OPTION χρησιμοποιείτε για την αναζήτηση πληροφορίας από ένα user agent ή έναν server σχετικά με τις τρέχουσες ικανότητες του και τη διαθεσιμότητα του.
PRACK (RFC 3262)	Η μέθοδος PRACK χρησιμοποιείτε για να επιβεβαιώσει αξιόπιστα μεταδιδόμενα provisional responses (1XX Class responses).
Publish (RFC 3903)	Μια Publish μέθοδο χρησιμοποιείτε από έναν user agent για να στείλει/εκδόσει πληροφορία σχετική με την κατάσταση ενός event, σε έναν SIP Server (Event State Compositor (ESC)
Refer (RFC 3515)	Η μέθοδος REFER χρησιμοποιείτε για session redirection. Ουσιαστικά ένας user agent χρησιμοποιεί τη REFER για να απαιτήσει από έναν άλλο user agent να 'μετακινηθεί' σε ένα συγκεκριμένο SIP URI. Αυτή η μέθοδος χρησιμοποιείτε στα call transfers.
Register (RFC 3261)	Η μέθοδος REGISTER χρησιμοποιείτε από έναν SIP user agent για να κοινοποιήσει το τρέχον Contact URI (IP Address) στο SIP δίκτυο. Ένας UA στέλνει μια REGISTER προς έναν SIP Registrar Server, ώστε να καταχωρηθεί η τρέχουσα θέση του στον υπεύθυνο Location Server.
Subscribe (RFC 3265)	Η μέθοδος SUBSCRIBE χρησιμοποείτε από ένα user agent για να ζητήσει πληροφορία από ένα remote device για κάποια events και status updates.
Update (RFC 3311)	Μια UPDATE μέθοδος χρησιμοποιείτε για να αλλάξει την κατάσταση ενός session, χωρίς να επηρεαστεί το υπάρχον dialog.

Πίνακας 7:SIP requests

3.4.4 SIP Responses

Όπως έχουμε ήδη αναφέρει τα SIP Responses αναπαράγονται από ένα UAS ως απάντηση σε ένα request που στάλθηκε από ένα UAC. Τα responses κατηγοριοποιούνται αριθμητικά από 100s σε 600s και χαρακτηρίζονται από ένα reason phrase. Διαφορετικά SIP entities επεξέργαζονται ή καταλαβαίνουν ένα συγκεκριμένο response number. Τα 1XX responses θεωρούνται provisional responses, ενώ όλα τα άλλα responses θεωρούνται final responses. Παρακάτω θα περιγράψουμε κάθε κλάση αναλυτικά.

3.4.4.1 1XX Response Class (Ενα request έχει ληφθεί και επεξεργάζεται)

Η κλάση 1XX χρησιμοποιείτε για να δείξει το στάδιο στο οποίο βρίσκεται το call processing. Ένα αρχικό 1XX response που λαμβάνεται από έναν UAC, διαβεβαιώνει ότι η αρχική INVITE έχει ληφθεί από τον UAS, και δεν υπαρχει λόγος αναμετάδοσης του request. Τα responses 1XX φαίνονται στο παρακάτω πίνακα:

Response Class	Reason Phrase
100	Trying
180	Ringing
181	Call is Being Forwarded
182	Call Queued
183	Session In Progress

Πίνακας 8: 1XX Responses

3.4.4.2 2XX Response Class (Το request έχει ληφθεί επιτυχώς από τον UAS)

Η κλάση 2XX ενός response χρησιμοποιείται για να πιστοποιήσει ότι ένα request έχει ληφθεί και βρίσκεται υπο επεξεργασία από εναν UAS. Ακολουθεί πίνακας με τα πιο σημαντικά responses της κλάσης 2XX:

Response Class	Reason Phrase
200	OK
202	Accepted

204	No Notification

Πίνακας 9: 2XX Responses

3.4.4.3 Redirection (3xx): Επιπλέον προσπάθεια πρέπει να καταβληθεί (τυπικά από το αποστολέα) για να ολοκληρωθεί το request

Η κλάση 3XX ενός response χρησιμοποιείτε για να αναδρομολογηθεί το request σε ένα άλλο location, από τη στιγμή που ο καλούμενος έχει αλλάξει διεύθυνση προσωρινά ή μόνιμα. Τα redirection responses παρέχουν στον UAC τη διεύθυνση ενός άλλου proxy server ή τη τρέχουσα διεύθυνση του called party. Ακολουθεί πίνακας με τα πιο σημαντικά responses της κλάσης 3XX:

Response Class	Reason Phrase
301	Moved Permanently
302	Moved Temporarily
303	Use Proxy

Πίνακας 10: 3XX Responses

3.4.4.4 Client Error (4xx): Το request περιέχει συντακτικό λάθος ή δε μπορεί να ολοκληρωθεί στον παρόν Server.

Τα 4XX κλάσης responses χρησιμοποιούνται από servers ή UAS για να δηλώσουν ότι ένα request δε μπορεί να ολοκληρωθεί. Οι headers που περιέχονται σε ένα response 4XX κλάσης δηλώνουν στον UAC τη φύση του λάθους και τι πρέπει να αλλάξει στο request για να επαναμεταδοθεί. Ακολουθεί πίνακας με τα πιο σημαντικά responses της κλάσης 4XX:

Response Class	Reason Phrase
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found

405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Time out
409	Conflict
410	Gone
411	Length Required
412	Conditional Request Failed
413	Request Entity Too Large
414	Request URI to Long
415	Unsupported Media Type
416	Unsupported URI Scheme
417	Unknown Resource Priority
420	Bad Extension
421	Extension Required
422	Session Interval too long
423	Interval too brief
428	Use Identity header
429	Provide Referred Identity
430	Flow Failed
433	Anonymity Disallowed
436	Bad Identity Info header
437	Unsupported Certificate
438	Invalid Identity header
439	First hop lacks Outbound Support
440	Max Breadth Exceeded
470	Consent Exceeded
480	Temporarily Unavailable
481	Dialog/Transaction doesn't exist
482	Loop detected
483	Too many hops
484	Address Incomplete
485	Ambiguous
486	Busy here
487	Request Terminated
488	Not acceptable here

489	Bad Event
491	Request Pending
493	Request Undecipherable
494	Security Agreement Required

Πίνακας 11: 4XX Responses

3.4.4.5 Server Error (5xx): Ο Server απέτυχε να επεξεργαστεί ένα πιθανότατα valid request.

Η κλάση 5XX χρησιμοποιείτε για να δηλώσει ότι το request δε μπορεί να ολοκληρωθεί εξαιτίας ενός server λάθους. Αυτά τα responses μπορεί να περιέχουν ένα Retry-After header, ο οποίος δηλώνει ότι το request μπορεί να επαναεπεξεργαστεί μετά από κάποιο χρονικό διάστημα. Ακολουθεί πίνακας με τα πιο σημαντικά responses της κλάσης 5XX:

Response Class	Reason Phrase
500	Server Internal Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	Version Not Supported
513	Message Too Large
580	Preconditions Failure

Πίνακας 12: 5XX Responses

3.4.4.6 Global Failure (6xx): Το request δε μπορεί να επεξεργαστεί από κανένα server.

Ένα response κλάσης 6XX στέλνετε από ένα server ο οποίος έχει βάσιμη πληροφορία για το συγκεκριμένο URI, ότι το request δε μπορεί να ολοκληρωθεί. Και αυτής της κλάσης τα responses μπορεί να διαθέτουν έναν Retry-After: header, ο οποίος

επιτρέπει στο request να επαναεπεξεργαστεί μετά από κάποιο χρονικό διάστημα. Ακολουθεί πίνακας με τα πιο σημαντικά responses της κλάσης 6XX:

Response Class	Reason Phrase
600	Busy Everywhere
603	Decline
604	Does not exist anywhere
606	Not acceptable

Πίνακας 13: 6XX Responses

3.5 SIP Transactions

Το SIP είναι ένα πρωτόκολλο συναλλαγής. Μια SIP συναλλαγή (transaction) πραγματοποιείτε μεταξύ ενός client και ενός server και περιλαμβάνει όλα τα μηνύματα από το πρώτο αίτημα που στέλνει ένας client (εκτός από τα 1XX μηνύματα) σε έναν server, μέχρι τη τελευταία απάντηση που θα στείλει ο server στον client. Είναι σαφές ότι τα SIP transactions συσχετίζονται με τα SIP μηνύματα. Πιο συγκεκριμένα ένα SIP transaction αποτελείτε από ένα αίτημα (request) και μία ή περισσότερες τελικές απαντήσεις (response). Στην περίπτωση που το αίτημα ήταν ένα INVITE (γνωστό ως INVITE transaction) η συναλλαγή περιλαμβάνει μια ACK μόνο αν η τελική απάντηση δεν ήταν κάποιο 2XX μήνυμα. Αν η απάντηση ήταν ένα 2XX μήνυμα, η ACK δε θεωρείτε μέρος της συναλλαγής. Αυτός ο διαχωρισμός γίνετε για να δηλώσει τη σημαντικότητα ενός 200 OK μηνύματος. Ένας UAS είναι υπεύθυνος για την αναμετάδοση ενός 200 OK μηνύματος, ενώ ο UAC ευθύνεται για το acknowledgment αυτού του μυνήματος με μία ACK. Η αναμετάδοση της ACK αποτελεί ένα καινούριο transaction. Επομένως τα SIP transactions χωρίζονται σε κατηγορίες ανάλογα με το αν ένα transaction αρχικοποιείται λόγω αποστολής ή λήψης ενός SIP μηνύματος (client και server transaction). Επομένως κάθε transaction έχει μία client και μία server πλευρά. Η client πλευρά ορίζεται ως client transaction και η server πλευρά ως server transaction. Στις παρακάτω υποενότητες θα αναφερθούμε πιο αναλυτικά στις παραπάνω κατηγορίες και θα μιλήσουμε για τις παραμέτρους που ορίζουν ένα transaction.

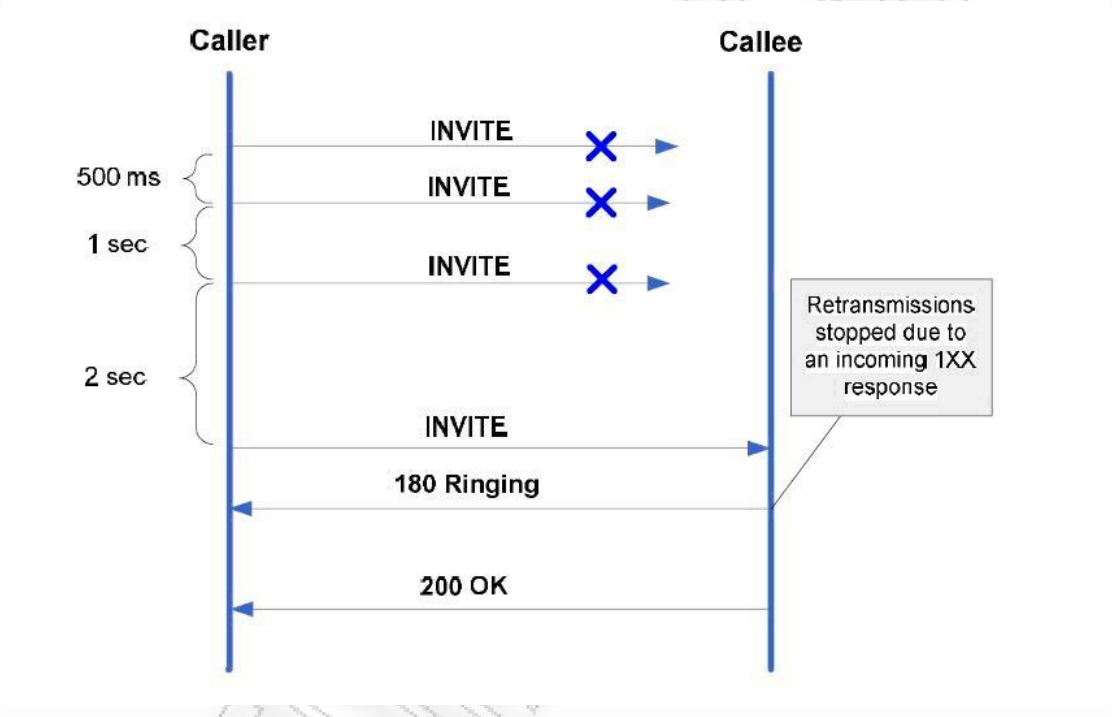
Επίσης τα transactions διαχωρίζονται σε δύο κατηγορίες ανάλογα με το αν το SIP request που προκαλεί τη δημιουργία του SIP transaction είναι ένα INVITE ή ένα non – INVITE request. Στην κατηγορία των non – INVITE requests περιλαμβάνονται όλα τα SIP μηνύματα εκτός από την INVITE και την ACK. Τα INVITE transactions διαφέρουν εξαιτίας της εκτεταμένης διάρκειας τους, εφόσον είναι απαραίτητη η αλληλεπίδραση με το χρήστη για την απόκριση σε ένα INVITE request. Για το λόγο αυτό τα INVITE transactions ακολουθούν μία three-way handshake διαδικασία για να ολοκληρωθούν, σε αντίθεση με τα non-INVITE transactions, τα οποία ολοκληρώνονται άμεσα, ακολουθώντας μία two-way handshake διαδικασία.

3.5.1 INVITE Client Transactions

Ένας transaction user επικοινωνεί με ένα client μέσω ενός απλού interface. Όταν ένας TU επιθυμεί να ξεκινήσει ένα καινούριο transaction, δημιουργεί ένα client transaction και προωθεί ένα SIP request με βάση την IP address, τη πόρτα και το πρωτόκολλο ματαφοράς στον τελικό χρήστη. Στην περίπτωση που ένα client transaction ενεργοποιείτε από μια INVITE μιλάμε για INVITE Client Transactions. Ένα INVITE Client Transaction αποτελείται από μία three-way handshake διαδικασία. Το transaction ξεκινά όταν ο χρήστης δημιουργήσει ένα INVITE request και το προωθήσει στο επίπεδο συναλλαγής (Transaction Layer). Η three-way handshake διαδικασία περιλαμβάνει την αποστολή του INVITE request, την λήψη ενός τελικού response και την αποστολή ενός ACK request για την επιβεβαίωση της λήψης. Στην περίπτωση αναξιόπιστων πρωτοκόλλων μεταφοράς όπως το UDP, ο client επαναμεταδίδει τα requests μετά από χρονικό διάστημα 500ms (T1 timer), το οποίο διπλασιάζεται μετά από κάθε επαναμετάδοση. Στην περίπτωση που χρησιμοποιείτε κάποιο αξιόπιστο πρωτόκολλο μεταφοράς, όπως το TCP, τα μηνύματα δε επαναμεταδίδονται. Οι επαναμεταδόσεις του αρχικού INVITE σταματούν εντελώς όταν ληφθεί κάποιο provisional response 1XX από το server. Ο client θα απαντήσει με ένα τελικό ACK σε κάθε τελικό response που λαμβάνει, και θα σταματήσουν οι επαναμεταδόσεις των responses.

Στην περίπτωση που το transaction είναι ακόμα σε ‘Calling’ state, μετά από χρονικό διάστημα 64*T1, που ισοδυναμεί με την αποστολή επτά επαναμεταδόσεων ένος INVITE μηνύματος, ο client πρέπει να ενημερώσει το TU για τον τερματισμό του transaction. Στην περίπτωση πάλι που ο client λάβει ένα provisional response (eg.

180 Ringing), οι επαναμεταδόσεις της αρχικής INVITE θα σταματήσουν και το response θα προωθηθεί στο TU. Τέλος, η λήψη ενός 2XX response σηματοδοτεί και τη λήξη του transaction. Η διαχείρηση ενός τέτοιου response εξαρτάται από το αν ο TU είναι ένα proxy core ή ένα UAC core. Ένα UAC core θα στείλει πίσω ένα ACK μήνυμα, σε αντίθεση με το proxy core που θα προωθήσει το 200OK μήνυμα. Ο διαφορετικός τρόπος προσέγγισης ενός 200 OK μηνύματος από ένα proxy και ένα client, επεξηγεί γιατί τα μηνύματα αυτά δε τα διαχειρίζετε το επίπεδο συναλλαγής (transaction layer).

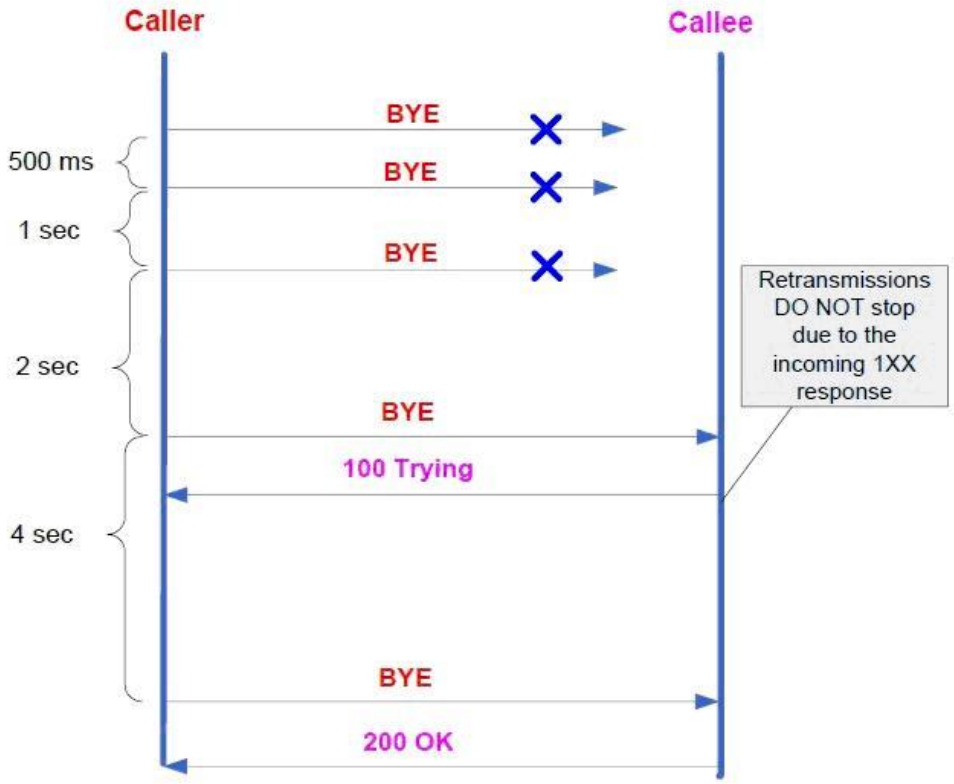


Σχήμα 12: Παράδειγμα ενός INVITE Client Transaction**.**

3.5.2 Non - INVITE Client Transactions

Όπως έχουμε ήδη αναφέρει όταν μιλάμε για Non-Invite Transactions, αναφερόμαστε στα client transactions τα οποία αρχικοποιούνται από έναν UAC με την αποστολή ενός SIP request, διαφορετικού από INVITE. Τα Non – INVITE transactions δε χρησιμοποιούν την ACK για την ολοκλήρωση ενός transaction, είναι μια απλή αλληλεπίδραση ενός request με ένα response. Στην περίπτωση αναξιόπιστων πρωτοκόλλων, τα requests αναμεταδίδονται για πρώτη φορά μετά από χρονικό διάστημα T1(500ms), το οποίο διπλασιάζεται μέχρι να φτάσει την τιμή των 4s (T2). Ο T2 αντιπροσωπεύει το χρονικό διάστημα που ένα non-INVITE server transaction

χρειάζεται για να απαντήσει σε ένα request, στην περίπτωση που δεν απαντήσει αμέσως. Οι συνήθεις τιμές για τους T1 και T2 είναι 500 ms, 1 s, 2 s, 4 s, 4 s, 4 s, etc. Στην περίπτωση που ληφθεί κάποιο provisional response, οι αναμεταδόσεις συνεχίζονται για τα αναξιόπιστα πρωτόκολλα μεταφοράς ανα χρονικά διαστήματα ισοδύναμα με T2. Ένα server transaction αναμεταδίδει το τελευταίο response που έστειλε μόνο όταν λάβει μια αναμετάδοση από το τελευταίο request που του έστειλε ένας client. Έτσι εξηγείτε γιατί ένα request αναμεταδίδεται ακόμα και όταν ληφθεί ένα provisional response, εξασφαλίζοντας την αξιόπιστη παράδοση του τελικού response. Αντίθετα με ένα INVITE transaction, τα non-INVITE transactions δεν προβλέπουν κανέναν ειδικό χειρισμό για τη λήψη των τελικών 2XX responses.



Σχήμα 13: Παράδειγμα ενός non-INVITE Client Transaction**.**

3.5.3 Αντιστοίχιση ενός Response σε ένα Client Transaction

Όταν ένας client λαμβάνει ένα response στο επίπεδο μεταφοράς, πρέπει να καθορίσει πιο client transaction θα διαχειριστεί αυτό το response. Η παράμετρος branch στο Via

header χρησιμοποιείτε για αυτό το λόγο. Ένα response συνδυάζεται με ένα client transaction κάτω από δύο συνθήκες:

1. Εάν το response έχει τον ίδιο branch header με το branch header του request που δημιούργησε το συγκεκριμένο transaction.
2. Εάν η μέθοδος που υποδηλώνεται στο CSeq header ταιριάζει με τη μέθοδο του request που δημιούργησε το transaction. Εξαίρεση αποτελεί ένα CANCEL request το οποίο δημιουργεί ένα καινούριο transaction, ενώ διατηρεί την τιμή της branch παραμέτρου.

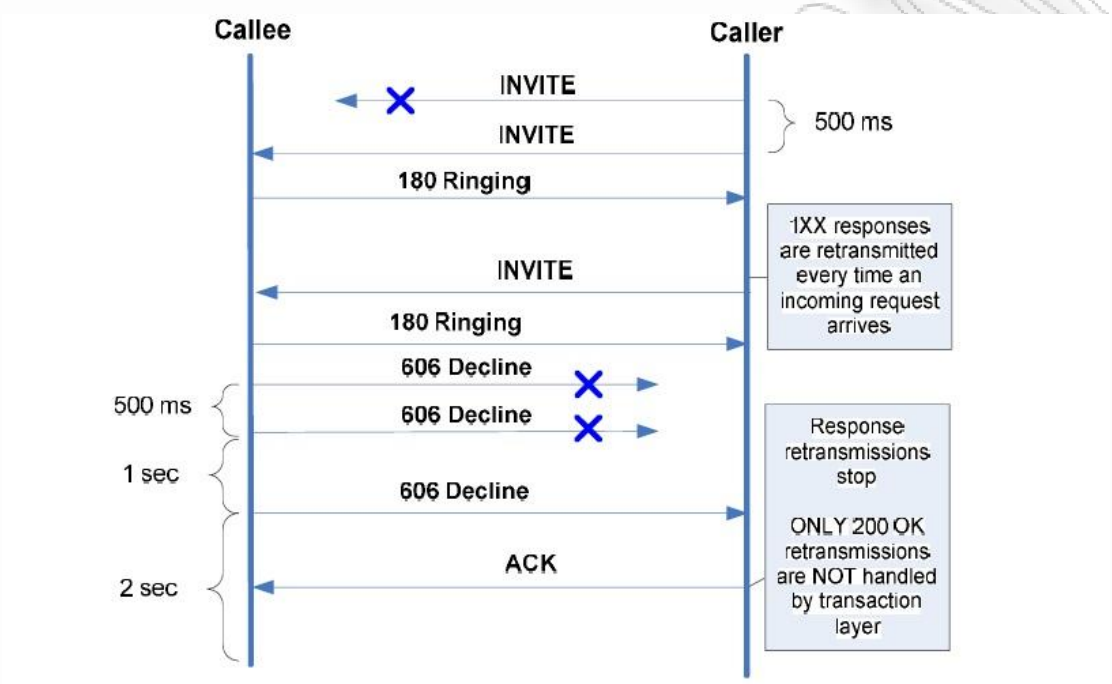
Εάν ένα request στέλνεται με multicast, είναι πιθανό να λάβει πολλαπλά responses από διαφορετικούς servers. Όλα αυτά τα responses θα έχουν την ίδια branch παράμετρο στον Via header, αλλά διαφορετικό To tag. Το πρώτο response που λαμβάνεται, με βάση τους παραπάνω κανόνες, είναι το βασικό μήνυμα, ενώ όλα τα επόμενα θεωρούνται επαναμεταδόσεις. Αυτή η συμπεριφορά δε θεωρείτε εσφαλμένη; Το SIP multicast παρέχει μόνο μια "single-hop-discovery-like" υπηρεσία, η οποία περιορίζεται στο να διαχειριστεί ένα μόνο response.

3.5.4 INVITE Server Transactions

Τα server transactions είναι υπεύθυνα για την παράδοση των requests στους TU και την αξιόπιστη μετάδοση των responses. Πιο συγκεκριμένα τα INVITE Server Transactions αρχικοποιούνται στους UAS όταν ένα INVITE request σταλθεί από ένα UAC. Όταν δημιουργείτε ένα server transaction για κάποιο request, ξεκινά η κατάσταση "Proceeding". Το server transaction πρέπει να φτιάξει ένα 100 Trying response εκτός και αν ξέρει ότι ο TU θα στείλει ένα provisional ή final response στα επόμενα 200ms. Τα responses προωθούνται στη συνέχεια στο επίπεδο μεταφοράς για να μεταδοθούν. Στην περίπτωση που ληφθεί σε επανάληψη κάποιο request, το πιο πρόσφατο provisional response θα προωθηθεί στο επίπεδο μεταφοράς για αναμετάδοση. Μόνο η αναμετάδοση των 200 OK responses σε INVITE requests δεν πραγματοποιείται από το Transaction Layer, αλλά το αναλαμβάνει το Transaction User Layer.

Ένα INVITE transaction διαχειρίζεται με διαφορετικό τρόπο τα τελικά responses από τα provisional responses. Έτσι τα τελικά responses, εκτός των 2XX,

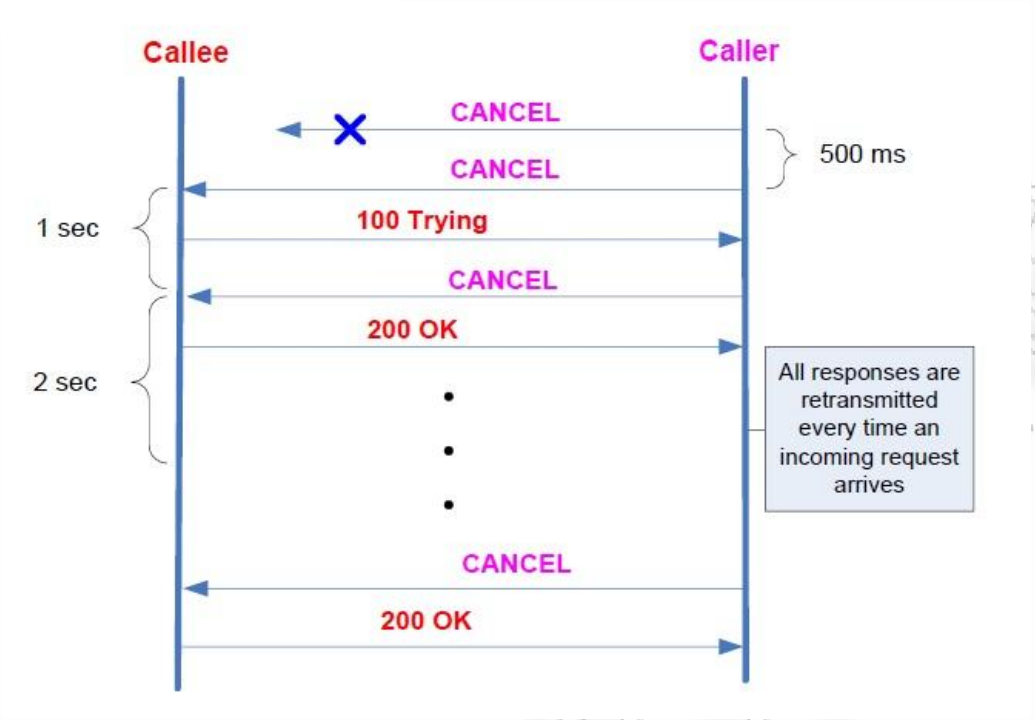
επαναδρομολογούνται μέχρι να ληφθεί μια τελική ACK, σε τακτά χρονικά διαστήματα των 500ms, τα οποία διπλασιάζονται μετά από κάθε αναμετάδοση. Οι αναμεταδόσεις σταματούν μετά την έβδομη επανάληψη, ή αν ληφθεί ένα matching BYE request στην περίπτωση του 200 OK response, ή ένα matching CANCEL request στην περίπτωση των 3XX, 4XX, 5XX, 6XX responses.



Σχήμα 14: Παράδειγμα ενός INVITE Server Transaction**.**

3.5.5 Non INVITE Server Transactions

Αυτού του είδους τα transactions δημιουργούνται από ένα SIP UAS, όταν λάβει ένα non-INVITE request από έναν απομακρυσμένο UAC, όπως CANCEL ή BYE request. Όπως και στην περίπτωση των Non-INVITE Client Transactions και σε αυτή την περίπτωση χρησιμοποιείται η two-way handshake διαδικασία για την ολοκλήρωσή τους, επειδή δεν λαμβάνονται ACK μηνύματα για την επιβεβαίωση των τελικών responses που στάλθηκαν προηγουμένως. Έτσι, στην περίπτωση που χρησιμοποιείται μη αξιόπιστη σύνδεση για την ανταλλαγή δεδομένων, κατά την διάρκεια ενός non-INVITE Server Transaction αναμεταδίδεται το τελευταίο response, ανεξάρτητα από το αν είναι τελικό ή provisional, κάθε φορά που λαμβάνεται μια αναμετάδοση του αρχικού request.



Σχήμα 15: Παράδειγμα ενός INVITE non-Server Transaction**.**

3.5.6 Αντιστοίχιση ενός Request σε ένα Server Transaction

Όταν ένα request καταλήγει μέσω του δικτύου σε ένα server, θα πρέπει να συνδυαστεί με κάποιο υπάρχον transaction. Αυτό πραγματοποιείτε με βάση έναν κανόνα ο οποίος εξετάζει την παράμετρο branch στο VIA header. Επομένως αν η παράμετρος αυτή υπάρχει και ένα κομμάτι της είναι "z9hG4bK", το request αυτό προήλθε από ένα Client transaction, ενώ η παράμετρος branch θα παραμένει μοναδική για όλα τα transactions που αρχικοποιούνται από αυτόν τον client. Ένα request αντιστοιχεί σε ένα transaction όταν συντρέχουν οι παρακάτω προυποθέσεις:

1. Η παράμετρος branch του request να είναι κοινή με την branch του VIA header που δημιούργησε το request.
2. Το κομμάτι του VIA header που προσδιορίζει τη προέλευση του request να είναι ίδιο με αυτό του αρχικού request.
3. Η μέθοδος ενός request να ταιριάζει με τη μέθοδο που δημιούργησε το transaction. Μοναδική εξαίρεση αποτελεί η ACK, όπου η μέθοδος που δημιούργησε το transaction ήταν ένα INVITE μήνυμα.

Οι κανόνες αυτοί συντρέχουν τόσο για τα INVITE όσο και για τα non-INVITE transactions. Στην περίπτωση που η παράμετρος branch απουσιάζει από το VIA header ακολουθούνται κάποιες διαδικασίες για το προσδιορισμό του matching transaction, έτσι ώστε να αποφευχθούν αναντιστοιχίες με άλλους clients.

Έτσι ένα INVITE request αντιστοιχεί σε ένα transaction αν το Request-URI, το To tag, το Call-ID, το CSeq number (όχι η μέθοδο) και ο VIA header, αντιστοιχίζονται ένα προς ένα με τα αντιστοιχα headers του INVITE που δημιούργησε το transaction. Σε αυτή την περίπτωση, η INVITE είναι μια αναμετάδοση του αρχικού INVITE που δημιούργησε το transaction. Ένα ACK request αντιστοιχεί σε ένα transaction εάν το Request-URI, το From tag, το Call-ID, το CSeq number (όχι η μέθοδος), και ο VIA header ταιριάζουν με τις αντίστοιχες παραμέτρους του αρχικού INVITE που δημιούργησε το transaction και το To tag του ACK ισοδυναμεί με το To tag του response που στάλθηκε από το server transaction. Ο To tag header σε ένα ACK μήνυμα διαχωρίζει τις ACK από τα 2XX μηνύματα από τις ACK άλλων responses σε έναν άλλο proxy. Ένα ACK request που αντιστοιχεί σε ένα INVITE transaction και έχει τους ίδιους headers με ένα προηγούμενο ACK, αποτελεί αναμετάδοση του αρχικού μηνύματος.

Σε όλες τις άλλες περιπτώσεις, ένα request αντιστοιχίζεται σε ένα transaction εάν τα Request-URI, To tag, From tag,Call-ID, CSeq (συμπεριλαμβανομένης και της μεθόδου), και ο Via header ταυτίζονται με εκείνα του request που δημιούργησε το transaction.

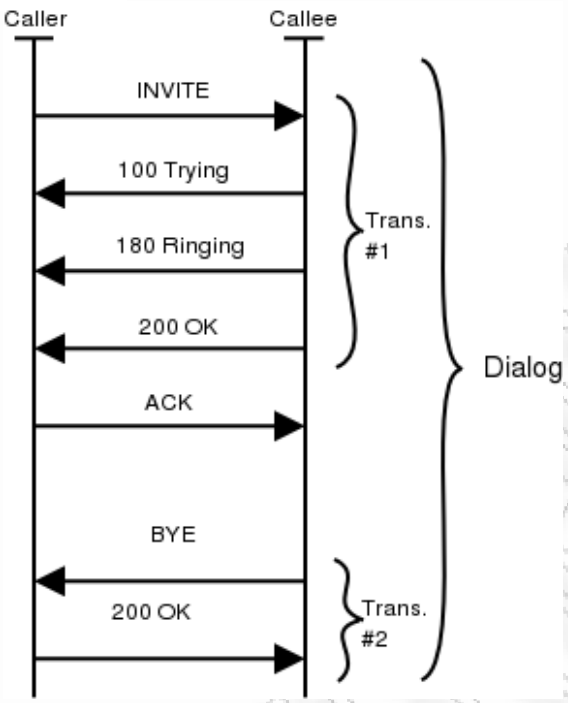
3.6 SIP Dialogs

Στις παραπάνω ενότητες μιλήσαμε για το τι είναι τα SIP transactions, ότι ένα transaction περιλαμβάνει μια INVITE και όλα τα responses που ακολουθούν για την εγκατάσταση μιας κλήσης και ότι ένα άλλο transaction περιλαμβάνει μια BYE και τα αντίστοιχα responses για την ολοκλήρωση ενός session. Τα δύο αυτά transactions θα πρέπει να συσχετιστούν μεταξύ τους μιας και αναφέρονται στην ολοκλήρωση της ίδιας κλήσης, γι' αυτό λέμε ότι ανήκουν στο ίδιο dialog. Ένα dialog αποτελεί μια peer to peer σύνδεση μεταξύ δύο User Agents. Ένα dialog υφίσταται για κάποιο χρονικό διάστημα και διευκολύνει τη σωστή αλληλουχία και τη δρομολόγηση των μηνυμάτων μεταξύ των endpoints.

Τα dialogs προσδιορίζονται από το Call-ID, το From tag και Το tag. Τα μηνύματα που διαθέτουν τους τρεις αυτούς identifiers ανήνουν στο ίδιο dialog. Όπως θα δούμε και παρακάτω ο CSeq header χρησιμοποιείτε για να βάζει σε σειρά τα μηνύματα, και πιο συγκεκριμένα καθορίζει τη σειρά των μηνυμάτων σε ένα dialog. Ο αριθμός πρέπει να αυξάνετε μονοτονικά για κάθε μήνυμα που στέλνετε μέσα στο ίδιο dialog, διαφορετικά το peer θα το διαχειριστεί σα μήνυμα εκτός σειράς ή σαν αναμετάδοση κάποιου άλλου μηνύματος. Στην παραγματικότητα το CSeq αναγνωρίζει ένα transaction μέσα σε ένα dialog γιατί όπως έχουμε ήδη αναφέρει ένα transaction δε είναι τίποτα άλλο από ένα request με τα αντίστοιχα responses του. Αυτό σημαίνει ότι μόνο ένα transaction σε κάθε κατεύθυνση μπορεί να είναι active μέσα σε ένα dialog. Σα συμπέρασμα μπορούμε να πούμε ότι ένα dialog έιναι μια σειρά από transactions.

Κάποια μηνύματα εγκαθιστούν ένα dialog και κάποια όχι. Αυτό βοηθάει στο να διασαφηνίζεται η σχέση των μηνυμάτων και στο να στέλνονται μηνύματα τα οποία δε συνδέονται με μηνύματα εκτός dialog. Αυτό είναι και πιο απλό σαν implementation, από τη στιγμή που ο user agent δε χρειάζεται να αποθηκεύει το dialog state.

Για παράδειγμα, ένα INVITE μήνυμα μπορεί να ξεκινήσει ένα καινούριο dialog, επειδή θα το ακολουθήσει ένα BYE request το οποίο θα τερματίσει και το session. Αυτή η BYE θα σταλεί μέσα στο dialog που έχει εγκατασταθεί από την INVITE. Στο παρακάτω σχήμα παρουσιάζεται το call flow ενός dialog με δύο transactions.



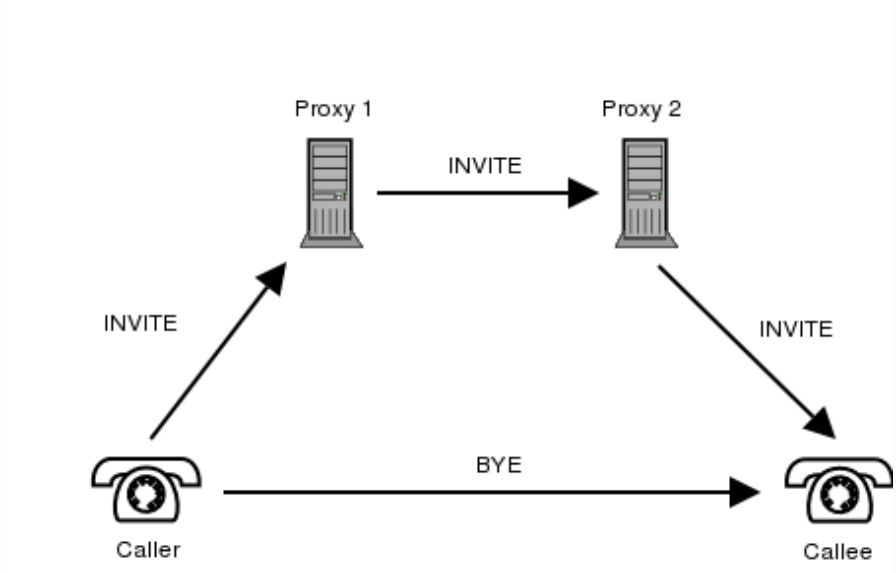
Σχήμα 16: SIP session το οποίο περιλαμβάνει δύο transactions.

3.6.1 Τα Dialogs διευκολύνουν τη δρομολόγηση των μηνυμάτων (Routing)

Σε αυτή την ενότητα θα ασχοληθούμε με τη χρήση των dialogs για τη δρομολόγηση μηνυμάτων μεταξύ των user agents. Ας υποθέσουμε ότι ο χρήστης sip:bob@a.com θέλει να επικοινωνήσει με το χρήστη sip:pete@b.com. Ο bob αν και γνωρίζει τη SIP διεύθυνση του καλούντος (sip:pete@b.com) δε διαθέτει αρκετή πληροφορία για την τρέχουσα θέση του χρήστη, για παράδειγμα δε ξέρει σε ποιο host να στείλει το request. Για αυτό ακριβώς το λόγο η INVITE θα σταλεί σε ένα proxy server. Το request θα σταλεί από proxy σε proxy μέχρι να φτάσει σε εκείνον ο οποίος γνωρίζει την τρέχουσα θέση του callee. Αυτή η διαδικασία ονομάζεται δρομολόγηση (routing). Μόλις το request φτάσει στον callee, ο user agent του callee θα δημιουργήσει ένα response το οποίο θα αποσταλεί πίσω στον caller. Ο user agent του callee θα βάλει ως contact header στο response την τρέχουσα θέση του χρήστη. Το αρχικό request περιείχε επίσης Contact header, γεγονός που καθιστά γνώστες τις τρέχουσες θέσεις και των δύο τερματικών. Επειδή οι δύο users agents ξέρουν ο ένας τη θέση του άλλου, δε χρειάζετε να στείλουν επιπλέον μηνύματα σε κάποιον proxy, αλλά μπορούν απλά να ανταλάσσουν μηνύματα μεταξύ τους. Με αυτό ακριβώς τον τρόπο τα dialogs βοηθούν στη δρομολόγηση των μηνυμάτων.

Όλα τα άλλα μηνύματα μέσα στο dialog στέλνονται απευθείας από user agent σε user agent. Αυτή είναι μια σημαντική βελτίωση γιατί οι proxies δε χρειάζεται να βλέπουν όλα τα μηνύματα ενός dialog και χρησιμοποιούνται απλά για να κατευθύνουν μόνο το πρώτο request που ξεκινά το συγκεκριμένο dialog. Τα απευθείας μηνύματα που ανταλάσσονται μεταξύ των δύο τερματικών έχουν μικρότερη καθυστέρηση και αυτό γιατί ένας τυπικός proxy συνήθως χρησιμοποιεί πιο πολύπλοκη λογική διευθυνσιοδότησης.

Στο σχήμα που ακολουθεί βλέπουμε ένα παράδειγμα από ένα BYE μήνυμα το οποίο παρακάμπτει τους proxies (SIP Trapezoid).



Σχήμα 17: Sip Trapezoid – Δρομολόγηση μυνημάτων μέσω ενός proxy.

3.7 Το Πρωτόκολλο SDP

Το πρωτόκολλο SDP περιγράφτηκε για πρώτη φορά στο RFC 2237 και πρωταρχικός του ρόλος ήταν ο καθορισμός μιας τυποποιημένης σύνταξης για την περιγραφή των δεδομένων που ανταλλάσσονται κατά την διάρκεια ενός session. Η περιγραφή που παρέχει το SDP είναι ένα μικρό κείμενο που περιλαμβάνει το όνομα και τον σκοπό του session, τα μέσα (media), τα πρωτόκολλα, τους κωδικοποιητές και τις πληροφορίες συγχρονισμού. Είναι έτσι σχεδιασμένο ώστε να μπορεί να συνεργαστεί με πολλά πρωτόκολλα όπως τα SIP, HTTP, RTSP, τα emails και άλλα.

Το Session Description Protocol μεταφέρεται στο message body τόσο των SIP requests όσο και τον SIP responses. Αυτό που πρέπει να τονίσουμε είναι ότι το SDP δεν μεταφέρει media, απλά περιγράφει παραμέτρους. Στο πίνακα που ακολοθεί περιγράφονται μερικές από τις σημαντικότερες και πιο σύνηθες παραμέτρους σε ένα SDP πακέτο, για την περιγραφή ενός session.

Παράμετρος	Περιγραφή	Mandatory/Optional
v	Έκδοση του πρωτοκόλλου SDP	m
o	Ο δημιουργός του session	m
s	Το όνομα του session	m
i	Πληροφορίες για το session	o
u	URI- Uniform Resource Identifier	o
e	Email Address	o
p	Τηλεφωνικό νούμερο	o
c	Πληροφορία σύνδεσης	m
b	Πληροφορίες εύρους ζώνης	o
t	Ο χρόνος κατά τον οποίο το session παραμένει ενεργό	m
r	Αριθμός επαναλήψεων	o
z	Ρυθμίσεις χρονικής ζώνης	o
k	Κλειδί κωδικοποίησης	o
a	Χαρακτηριστικά του session (π.χ. codecs και πρωτόκολλο μεταφοράς)	o
m	Περιγραφή των δεδομένων που θα χρησιμοποιηθούν στη	o

διάρκεια του session

Πίνακας 14: SDP parameters.

Ένα session μπορεί να είναι χρονικά περιορισμένο όποτε και θα είναι ενεργό μόνο για συγκεκριμένες χρονικές περιόδους. Το SDP μεταφέρει τη λίστα με τις στιγμές έναρξης και λήξης της συνόδου. Τις πληροφορίες συγχρονισμού ακολουθεί ένα τμήμα όπου περιγράφονται τα μέσα (media) (video, ήχος κ.α.), το πρωτόκολλο μεταφοράς (RTP/UDP/IP/H.320 κ.α.) η μορφοποίηση των μέσων (media) (H.262 video, MPEG video κ.α.), τα χαρακτηριστικά των κωδικοποιητών κ.α.

Οι πληροφορίες που παρέχονται σε ένα SDP πακέτο μπορούν να χωριστούν σε δυο κατηγόριες:

- Σε αυτές που αναφέρονται σε ολόκληρο το session και σε όλες τις ροές πακέτων και αποτελούν το session level. Το τμήμα αυτό ξεκινά με μια γραμμή που αρχίζει με ‘v=’.
- Σε αυτές που μεταφέρουν πληροφορίες για μια συγκεκριμένη ροή πακέτων (media description). Το τμήμα αυτό βρίσκεται αμέσως μετά το επίπεδο συνόδου και ξεκινάει με ‘m=’.

Στον παρακάτω πίνακα φαίνεται η σύνταξη ενός πλήρους SDP πακέτου.

Field	Format
v	**v=0**
o	<username> <session id> <version> <network type> <address type> <address>
s	<session name>
i	<textual description>
u	<uri>
e	<email-address>
p	<phone-number>
c	<network type> <address type> <connection address>
b	<bwtype>:<bandwidth>
t	< time> <stop time>

r	<repeat interval> <active duration> <offsets from start-time>
z	<adjustment time> <offset> <adjustment time> <offset>
k	<method>:<encryption key>
a	<attribute><value>
m	<media> <port> <transport> <format list>

Πίνακας 15: Σύνταξη ενός SDP πακέτου.

Με βάση όλα όσα είπαμε παραπάνω, οι παράμετροι μιας VoIP κλήσης μπορούν να περιγραφούν πλήρως μέσω ενός SDP πακέτου. Έστω ότι έχουμε το παρακάτω SDP τμήμα:

v=0

o=man 3 2 IN IP4 192.168.16.105

s=CounterPath X-Lite 3.0

c=IN IP4 192.168.16.105

t=0 0

m=audio 32854 RTP/AVP 107 0 8 101 ⇐ media stream

a = rtpmap:0 PCMA/8000 ⇐ attributes

a=sendrecv

Από το παραπάνω SDP τμήμα παίρνουμε την παρακάτω πληροφορία:

- Το version number του πρωτοκόλλου είναι 0 (v parameter).
- Ο χρήστης man είναι υπεύθυνος για τη διαχείρηση του session και η φυσική διεύθυνση του είναι 192.168.16.105 (o parameter).
- Το όνομα του session είναι CounterPath X-Lite 3.0 (s parameter).
- Ο τύπος των media δεδομένων που θα ανταλλάσσονται είναι audio, με port number λήψης 32854 και πρωτόκολλο μεταφοράς RTP (m parameter)
- Οι codecs που υποστηρίζονται για το συγκεκριμένο session είναι PCMA, ενώ ο ρυθμός δειγματοληψίας είναι 8000 samples/sec (a parameter).

Το RFC 3264 παρουσιάζει όλες τις διαδικασίες που είναι απαραίτητες για την διαπραγμάτευση των παραμέτρων μιας κλήσης με τη χρήση του SDP, βασιζόμενο σε ένα offer/answer μοντέλο. Ένα offer περιλαμβάνει τα προτεινόμενα media types και τους codecs ενός session. Ουσιαστικά αναφερόμαστε στο SDP σώμα του αρχικού INVITE request που στάλθηκε από τον UAC του caller. Από την άλλη μεριά, ένα answer περιλαμβάνει ένα υποσύνολο των media types και των codecs που πρότεινε ο caller, τα οποία υποστηρίζονται και από τον callee ο οποίος δέχθηκε το offer. Το answer είναι στην ουσία το SDP σώμα του 200 OK response που αποστέλλει ο callee πίσω στον caller, αν αποδεχθεί την κλήση. Η διαδικασία αυτή ονομάζεται SDP negotiation, και είναι ουσιαστικά η διαδικάσια διαπραγμάτευσης των δυο calling parties για τα media types και τους codecs του session.

Τα offer/answers μπορούν να περιέχουν περισσότερα από ένα media sreams, με τη προυπόθεση ότι όσα media streams υπάρχουν σε ένα offer τόσα θα υπάρχουν και σε ένα answer. Εάν ο answerer δεν υποστηρίζει ένα media stream που εμφανίζεται στο offer ως μια 'm=' line, τότε στην απάντησή του γνωστοποιεί το παραπάνω γεγονός, συμπεριλαμβάνοντας αυτή την 'm=' line, αλλά θέτοντας το port number μηδενικό:

m = audio 0 RTP/AVP 0 1

Τα media streams κατατάσσονται κατά σειρά προτίμησης, ξεκινούν με ένα 'm=' και περιέχουν πληροφορία για τον τύπο των δεδομένων (audio), το port number της λήψης και τις payload τιμές των codecs (media formats) για την κωδικοποίηση /αποκωδικοποίηση των media δεδομένων.

Μετά από τα media lines ακολουθεί μια σειρά από attributes. Ένα από τα πιο σημαντικά attributes αποτελεί το 'a=rtpmap', το οποίο υποδυκνύει τους codecs για την κωδικοποίηση /αποκωδικοποίηση των media δεδομένων. Στην περίπτωση που η μεταφορά δεδομένων γίνεται μέσω RTP, το attribute 'a = rtpmap' περιέχει και την αντιστοίχηση ενός codec με το RTP payload value. Εξαιρετικής σημαντικότητας attribute αποτελεί και εκείνο που δηλώνει τη κατευθυντικότητα ενός media stream. Έτσι στην περίπτωση που έχουμε το attribute 'a = sendonly' σε ένα SDP body υποστηρίζουμε μόνο αποστολή δεδομένων, στην περίπτωση 'a = recvonly' υποστηρίζουμε μόνο τη λήψη δεδομένων, ενώ στην 'a = sendrecv' μπορούμε να έχουμε και αποστολή και λήψη δεδομένων. Τέλος σε περίπτωση που θέλουμε να

βάλουμε σε hold το ένα leg ενός session το attribute θα έχει την τιμή ‘a = inactive’, που σημαίνει ότι αυτή τη στιγμή δε μπορεί να γίνει ανταλλαγή δεδομένων και ο χρήστης είναι σε hold.

3.7.1 Διαδικασία του SDP Negotiation

Όταν ένα offer αποσταλεί μέσω ενός INVITE request, ο callee καλείτε να επιλέξει ανάμεσα στα media lines, τα media types που μπορεί να υποστηρίξει για να χρησιμοποιηθούν και από τα δύο calling legs κατά τη διάρκεια ενός session. Στη συνέχεια στέλνει ένα answer στον caller μέσω ενός 200 OK μηνύματος με τα επιλεγμένα media formats.

Η δημιουργία ενός answer βασίζεται κατά κύριο λόγω στο offer, με την τροποποίηση μερικών στοιχείων όπως η IP διεύθυνση και το port number για τη λήψη των RTP δεδομένων, η αφαίρεση των media types και codecs που δεν υποστηρίζονται από τον answerer, κ.λ.π. Η γραμμή προέλευσης (‘o=’ line) θα αλλάξει, ενώ το πεδίο που καθορίζει τα χρονικά πλαίσια του υπό εγκατάσταση session (‘t=’ line) παραμένει το ίδιο. Εάν ένα offer γίνει αποδεκτό, τότε η απάντηση πρέπει να περιέχει μια unicast διεύθυνση (εάν το offer ήταν unicast) χωρίς να μεταβάλει τα media types. Τέλος, ένα answer πρέπει να χαρακτηρίσει την κατεύθυνση του media stream ανάλογα με την κατεύθυνση του media stream του offer. Έτσι στην περίπτωση που η κατεύθυνση του media stream στο offer ήταν sendrecv, τότε η κατεύθυνση του media stream στο answer μπορεί να είναι recvonly, sendonly ή sendrecv.

Στο παρακάτω παραδειγμα βλέπουμε ένα Offer/Answer Model Session για τη διαχείρηση ενός video stream:

Ας υποθέσουμε ότι ο caller, Alice, έχει το παρακάτω SDP body, το οποίο περιλαμβάνει ένα αμφίδρομο audio stream και δύο αμφίδρομα video streams τα οποία χρησιμοποιούν H.261 (payload type 31) και MPEG (payload type 32) αντίστοιχα.

Το offered SDP είναι:

```
v=0

o=alice 2890844526 2890844526 IN IP4 host.anywhere.com
```

s=

c=IN IP4 host.anywhere.com

t=0 0

m=audio 49170 RTP/AVP 0

a=rtpmap:0 PCMU/8000

m=video 51372 RTP/AVP 31

a=rtpmap:31 H261/90000

m=video 53000 RTP/AVP 32

a=rtpmap:32 MPV/90000

a=sendrecv

Ο callee, Bob, δε θέλει να λάβει ή να στείλει το πρώτο video stream, οπότε επιστρέφει το παρακάτω SDP ως answer:

v=0

o=bob 2890844730 2890844730 IN IP4 host.example.com

s=

c=IN IP4 host.example.com

t=0 0

m=audio 49920 RTP/AVP 0

a=rtpmap:0 PCMU/8000

m=video 0 RTP/AVP 31 ⟸ Το port number θα είναι πλέον 0

m=video 53000 RTP/AVP 32

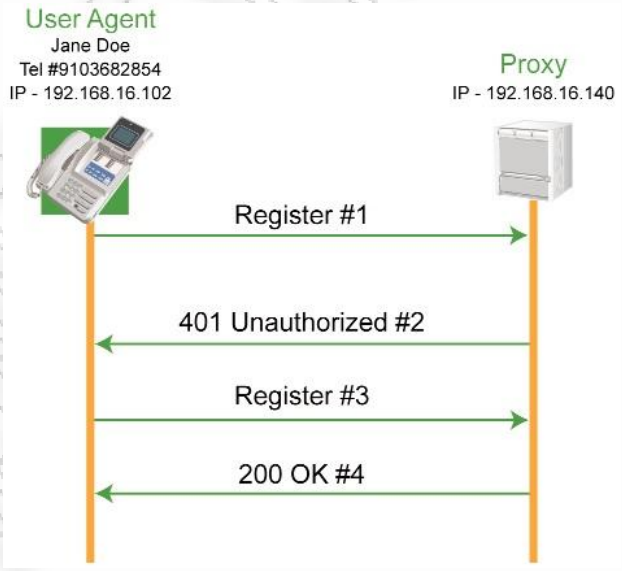
a=rtpmap:32 MPV/90000

a=sendrecv

3.8 Πραγματοποίηση Κλήσεων με το SIP πρωτόκολλο

3.8.1 SIP Registration Process

Η διαδικασία του registration χρησιμοποιείτε από έναν UA για να δεσμεύσει την τρέχουσα θέση του χρήστη και να διευκόλυνει τη δρομολόγηση των μηνυμάτων εντός του SIP δικτύου. Αυτό πραγματοποιείτε από έναν UA με την αποστολή ενός register μηνύματος σε έναν registrar server. Ο registrar διατηρεί το binding του συγκεκριμένου user για ένα συγκεκριμένο χρονικό διάστημα το οποίο καθορίζεται από τον Expires header. Εάν ένας UA επιθυμεί να παραμένει register, πρέπει περιοδικά να στέλνει re-register requests. Μετά το registration ο registrar στέλνει την πληροφορία σε έναν location server. Ένας location και ένας registrar server μπορεί να συνυπάρχουν στην ίδια φυσική οντότητα, για αυτό και η διαφορετικότητα τους έγκειτε κυρίως σε λογικό επίπεδο. Στην πραγματικότητα, οι συνδιασμοί Proxy Servers, Registrar και location registers είναι αρκετά συχνοί σε μικρά δίκτυα. Στο παρακάτω σχήμα ακολουθεί μια ολοκληρωμένη διαδικασία registration, όπου ο registrar "challenges" το register μήνυμα που έχει στείλει ο UA.



Σχήμα 18: Registration Proccess

Στη συνέχεια ακολουθούν αναλυτικά τα μηνύματα που ανταλλάσσονται κατά το registration process:

- **Register #1**

Request Line

REGISTER sip: 192.168.16.140 SIP/2.0

Headers

Via: SIP/2.0/UDP 192.168.16.102:10658;branch=z9hG4bK-d8754z- f64d606d85428835-1---d8754z-;rport

Max-Forwards: 70

Contact: <sip:9103682854@192.168.16.102:10658;rinstance=c1bbc08e764e1a59>

To: “Jane Doe"<sip:9103682854@192.168.16.140>

From: “Jane Doe"<sip:9103682854@192.168.16.140>;tag=74215226

Call-ID: MjNmYjBlZjhiODJiZmI4MjdhMzViNmIyMDFkNzk5MzQ.

CSeq: 1 REGISTER

Expires: 3600

Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE,INFO

User-Agent: X-Lite release 1104o stamp 56125

Content-Length: 0

- **401 Unauthorized #2**

Request Line

SIP/2.0 401 Unauthorized

Headers

Via: SIP/2.0/UDP 192.168.16.102:10658;branch=z9hG4bK-d8754z- f64d606d85428835-1---d8754z-;received=192.168.16.102;rport=10658

To: “Jane Doe"<sip:9103682854@192.168.16.140>

From: “Jane Doe"<sip:9103682854@192.168.16.140>;tag=74215226

Call-ID: MjNmYjBlZjhiODJiZmI4MjdhMzViNmIyMDFkNzk5MzQ.

CSeq: 1 REGISTER

WWW-Authenticate: Digest realm="192.168.16.140",qop="auth",

nonce="b409d25d445aa41753167c2d96645524",opaque="",stale=FALSE

Server: TekSIP/v2.7

Content-Length: 0

- **Register #3**

Request Line

REGISTER sip:192.168.16.140 SIP/2.0

Headers

Via: SIP/2.0/UDP 192.168.16.102:10658;branch=z9hG4bK-d8754z-

8c733834e54a867d-1---d8754z-;rport

Max-Forwards: 70

Contact: <sip:9103682854@192.168.16.102:10658;rinstance=c1bbc08e764e1a59>

To: “Jane Doe"<sip:9103682854@192.168.16.140>

From: “Jane Doe"<sip:9103682854@192.168.16.140>;tag=74215226

Call-ID: MjNmYjBlZjhiODJiZmI4MjdhMzViNmIyMDFkNzk5MzQ.

CSeq: 2 REGISTER

Expires: 3600

Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE,INFO

User-Agent: X-Lite release 1104o stamp 56125

[truncated] Authorization: Digest

username="9103682854",realm="192.168.16.140",nonce="b409d25d445aa41753167c2d96645524",uri="sip:192.168.16.140",response="3017cba4e1a09d6deafdbb876cff13ca",cnonce="6def0392c6f264bc88216693693ec23a",nc=00000001,qo

Content-Length: 0

- **200 OK #4**

Request Line

SIP/2.0 200 OK

Headers

Via: SIP/2.0/UDP 192.168.16.102:10658;branch=z9hG4bK-d8754z-8c733834e54a867d-1---d8754z- ;received=192.168.16.102;rport=10658

Max-Forwards: 69

Contact: <sip:9103682854@192.168.16.102:10658; rinstance=c1bbc08e764e1a59>

To: “Jane Doe"<sip:9103682854@192.168.16.140>;tag=43a8aa5e

From: “Jane Doe"<sip:9103682854@192.168.16.140>;tag=74215226

Call-ID: MjNmYjBlZjhiODJiZmI4MjdhMzViNmIyMDFkNzk5MzQ.

CSeq: 2 REGISTER

Allow: INVITE, BYE, CANCEL, OPTIONS, ACK, REGISTER, SUBSCRIBE, PUBLISH

Server: TekSIP/v2.7

Date: Fri, 30 Jul 2010 15:19:23 GMT

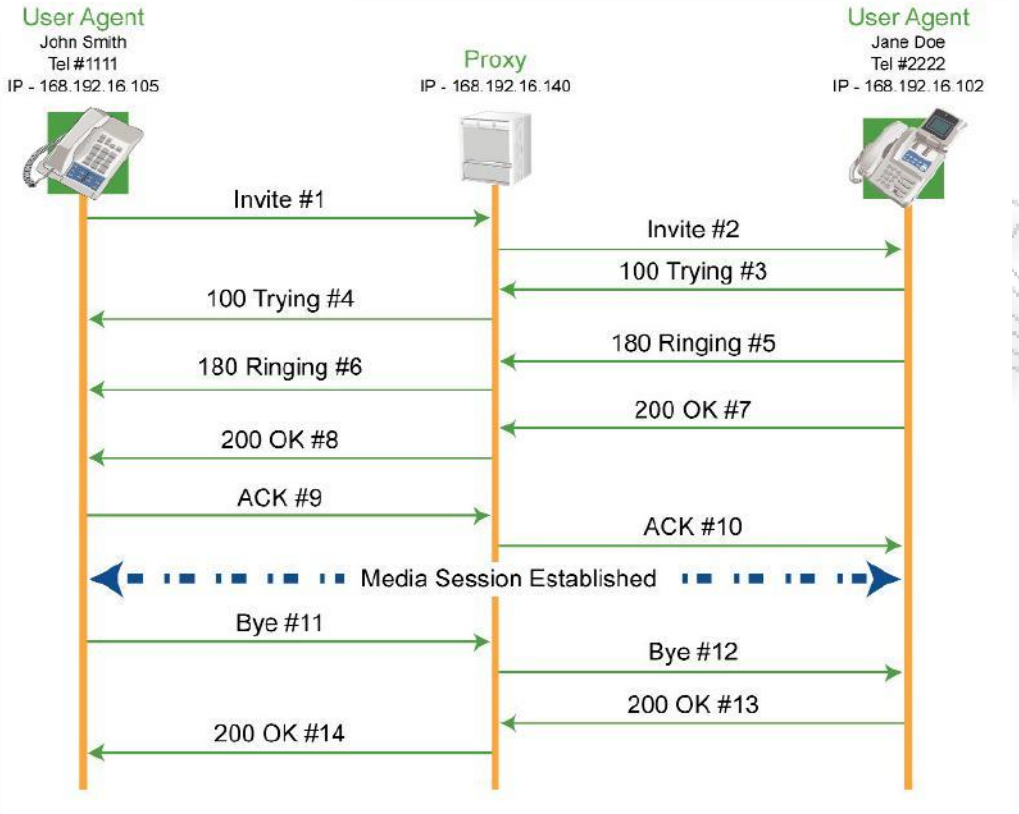
Allow-Events: presence, presence.winfo, message-summary

Expires: 3600

Content-Length: 0

3.8.2 SIP Call Processing

Στο παρακάτω σχήμα φαίνετε αναλυτικά η αλληλουχία των μηνυμάτων από μια SIP to SIP κλήση μέσω ενός Proxy server.



Σχήμα 19: SIP call processing

Αναλυτικά τα μηνύματα που ανταλλάσονται σε αυτή τη κλήση είναι τα εξής:

- **Invite #1**

Ο John Smith επιθυμεί να καλέσει την Jane Doe και πληκτρολογεί 9103682854 στον User Agent του. Ο User Agent είναι σεταρισμένος να δρομολογεί όλες τις εξερχόμενες κλήσεις μέσω ενός proxy με IP adress 192.168.16.140. Ο User Agent του John στέλνει ένα INVITE μήνυμα με όλα τα στοιχεία για τη συγκεκριμένη κλήση στο proxy server.

Request Line

INVITE sip:9103682854@192.168.16.140 SIP/2.0

Headers

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z- 2651db598629a27e-1---d8754z-;rport

Max-Forwards: 70

Contact: <sip:9103683957@192.168.16.105:44646

To: "Jane Doe"<sip:9103682854@192.168.16.140

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 INVITE

Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO

Content-Type: application/sdp

Proxy-Authorization: Digest username="9103683957", realm="192.168.16.140",nonce="2b028e9e929f845f097aa41196e511ac",uri="sip:9103682854@192.168.16.140",response="e714b87b94f9f5d4ca3b288984963fb5",cnonce="dc58e9e10d0a63b19d65b1380324a101",nc=

User-Agent: X-Lite release 1104o stamp 56125

Content-Length: 322

Empty Line

Message Body

v=0

o=- 3 2 IN IP4 192.168.16.105

s=CounterPath X-Lite 3.0

c=IN IP4 192.168.16.105

t=0 0

m=audio 32854 RTP/AVP 107 0 8 101

a=alt:1 2 : oND7cBBb qhQukSDp 10.100.100.250 32854

a=alt:2 1 : napYhQOC H731+lWt 192.168.16.105 32854

a=fmtp:101 0-15

a=rtpmap:107 BV32/16000

a=rtpmap:101 telephone-event/8000

a=sendrecv

- **Invite #2**

Ο Proxy Server δέχεται το INVITE request και το αναλύει, για να διαπιστώσει αν το request πρέπει να σταλθεί στο άλλο άκρο. Ο proxy ξέρει ότι η Jane Doe είναι διαθέσιμη αυτή τη στιγμή και η ip της είναι 192.168.16.102. Οπότε επισυνάπτει την ip address του στον Via header του request και το στέλνει στον user Agent της Jane Doe.

Request Line

INVITE sip:9103682854@192.168.16.102:56912 SIP/2.0

Headers

Via: SIP/2.0/UDP 192.168.16.140:5060;branch=z9hG4bK-85z25d58d7461b9

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-2651db598629a27e-1---d8754z-;received=192.168.16.105;rport=44646

Max-Forwards: 69

Contact: <sip:9103683957@192.168.16.105:44646>

To: "Jane Doe"<sip:9103682854@192.168.16.140>

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 INVITE

Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE,

INFO

Record-Route: <sip:192.168.16.140;lr>

User-Agent: X-Lite release 1104o stamp 56125

Content-Type: application/sdp

Content-Length: 322

Empty Line

Message

Body SDP

v=0

o=- 3 2 IN IP4 192.168.16.105

s=CounterPath X-Lite 3.0

c=IN IP4 192.168.16.105

t=0 0

m=audio 32854 RTP/AVP 107 0 8 101

a=alt:1 2 : oND7cBBb qhQukSDp 10.100.100.250 32854

a=alt:2 1 : napYhQOC H731+lWt 192.168.16.105 32854

a=fmtp:101 0-15

a=rtpmap:107 BV32/16000

- **100 Trying 3#**

Ο User Agent της Jane Doe δέχεται και αναλύει την INVITE που δέχεται από τον Proxy και στέλνει ένα 100 Trying Response ως απάντηση.

Request Line

SIP/2.0 100 Trying

Headers

Via: SIP/2.0/UDP 192.168.16.140:5060;branch=z9hG4bK-85z25d58d7461b9

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-2651db598629a27e-1---d8754z- ;received=192.168.16.105;rport=44646

To: "Jane Doe"<sip:9103682854@192.168.16.140

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 INVITE

Content-Length: 0

- **100 Trying #4**

Ο Proxy δέχεται την 100 Trying από το User Agent της Jane Doe και το προωθεί στον User Agent του John Smith με την ένδειξη ότι η κλήση χρειάζεται κάποιο χρόνο για να εγκατασταθεί.

Request Line

SIP/2.0 100 Trying

Headers

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-2651db598629a27e-1---d8754z- ;received=192.168.16.105;rport=44646

To: "Jane Doe"<sip:9103682854@192.168.16.140

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 INVITE

Max-Forwards: 70

Record-Route: <sip:192.168.16.140;lr>

Content-Length: 0

- **180 Ringing #5**

Ο User Agent της Jane χτυπάει ειδοποιώντας για την εισερχόμενη κλήση και στέλνει ένα 180 ringing για να ενημερώσει ότι το τηλέφωνο του άρχισε να κουδουνίζει.

Request Line

SIP/2.0 180 Ringing

Headers

Via: SIP/2.0/UDP 192.168.16.140:5060;branch=z9hG4bK-85z25d58d7461b9

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-2651db598629a27e-1---d8754z- ;received=192.168.16.105;rport=44646

Record-Route: <sip:192.168.16.140;lr

Contact: <sip:9103682854@192.168.16.102:56912

To: "Jane Doe"<sip:9103682854@192.168.16.140>;tag=f63e876c

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 INVITE

User-Agent: X-Lite release 1104o stamp 56125

Content-Length: 0

- **180 Ringing #6**

Ο proxy λαμβάνει την 180 Ringing και τη προωθεί στον user agent του John, ο οποίος παίζει ένα ring back tone.

Request Line

SIP/2.0 180 Ringing

Headers

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z- 2651db598629a27e-1---d8754z-;received=192.168.16.105;rport=44646

Contact: <sip:9103682854@192.168.16.102:56912

To: "Jane Doe"<sip:9103682854@192.168.16.140>;tag=f63e876c

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 INVITE

Max-Forwards: 70

Record-Route: <sip:192.168.16.140;lr

User-Agent: X-Lite release 1104o stamp 5612

Content-Length: 0

- **200 OK #7**

Ο User agent agent της Jane αποφασίζει να αποδεχτεί την εισερχόμενη κλήση από τον John και στέλνει ένα 200OK response στον proxy. Το response περιέχει SDP Message Body, το οποίο περιέχει όλες τις media παραμέτρους που απαιτούνται από τον user agent της Jane για να εγκατασταθεί το media session.

Request Line

SIP/2.0 200 OK

Headers

Via: SIP/2.0/UDP 192.168.16.140:5060;branch=z9hG4bK-85z25d58d7461b9

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-2651db598629a27e-1---d8754z-;received=192.168.16.105;rport=44646

Record-Route: <sip:192.168.16.140;lr

Contact: <sip:9103682854@192.168.16.102:56912

To: "Jane Doe"<sip:9103682854@192.168.16.140>;tag=f63e876c

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 INVITE

Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE,INFO

Content-Type: application/sdp

User-Agent: X-Lite release 1104o stamp 56125

Content-Length: 270

Empty Line

Message Body SDP

v=0

o=- 8 2 IN IP4 192.168.16.102

s=CounterPath X-Lite 3.0

c=IN IP4 192.168.16.102

t=0 0

m=audio 10574 RTP/AVP 107 0 8 101

a=alt:1 1 : EBcwd5ZA XQt5DSSm 192.168.16.102 10574

a=fmtp:101 0-15

a=rtpmap:107 BV32/16000

a=rtpmap:101 telephone-event/8000

a=sendrecv

- **200 OK #8**

Το 200 OK response λαμβάνεται από τον proxy και προωθείτε στον user agent του John, ο οποίος στέλνει τα ports, τους codecs και όλες τι άλλες παραμέτρους για το media session.

Request Line

SIP/2.0 200 OK

Headers

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-2651db598629a27e-1---8754z-;received=192.168.16.105;rport=44646

Contact: <sip:9103682854@192.168.16.102:56912

To: "Jane Doe"<sip:9103682854@192.168.16.140>;tag=f63e876c

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 INVITE

Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO

Max-Forwards: 70

Record-Route: <sip:192.168.16.140;lr

User-Agent: X-Lite release 1104o stamp 56125

Content-Type: application/sdp

Content-Length: 270

Empty Line

Message Body SDP

v=0

o=- 8 2 IN IP4 192.168.16.102

s=CounterPath X-Lite 3.0

c=IN IP4 192.168.16.102

t=0 0

m=audio 10574 RTP/AVP 107 0 8 101

a=alt:1 1 : EBcwd5ZA XQt5DSSm 192.168.16.102 10574

a=fmtp:101 0-15

a=rtpmap:107 BV32/16000

a=rtpmap:101 telephone-event/8000

a=sendrecv

- **ACK #9**

Ο user agent του John στέλνει ένα ACK στον proxy και ξεκινάει το media session με τη χρήση του RTP πρωτοκόλλου.

Request Line

ACK sip:9103682854@192.168.16.102:56912 SIP/2.0

Headers

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-cd63ad64065fb73c-1---d8754z-;rport

Max-Forwards: 70

Route: <sip:192.168.16.140;lr

Contact: <sip:9103683957@192.168.16.105:44646

To: "Jane Doe"<sip:9103682854@192.168.16.140>;tag=f63e876c

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 ACK

Proxy-Authorization: Digestusername="9103683957",realm="192.168.16.140",nonce="2b028e9e929f845f097aa41196e511ac",uri="sip:9103682854@192.168.16.140",response="e714b87b94f9f5d4ca3b288984963fb5",cnonce="dc58e9e10d0a63b19d65b1380324a101",nc=User-Agent: X-Lite release 1104o stamp 56125

Content-Length: 0

- **ACK #10**

Ο proxy λαμβάνει το ACK από το user agent του John και το προωθεί στο user agent της Jane.

Request Line

ACK sip:9103682854@192.168.16.102:56912 SIP/2.0

Headers

Via: SIP/2.0/UDP 192.168.16.140:5060;branch=z9hG4bK-85zc6a60d74d3d4

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-cd63ad64065fb73c-1---d8754z-;received=192.168.16.105;rport=44646

Max-Forwards: 69

Contact: <sip:9103683957@192.168.16.105:44646

To: "Jane Doe"<sip:9103682854@192.168.16.140>;tag=f63e876c

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 2 ACK

Route: <sip:192.168.16.140;lr

Record-Route: <sip:192.168.16.140;lr

User-Agent: X-Lite release 1104o stamp 56125

Content-Length: 0

- **BYE #11**

Μετά την ολοκλήρωση της συνομιλίας ο John κλείνει το τηλέφωνο και στέλνει ένα BYE request στο proxy για να τερματίσει το media session.

Request Line

BYE sip:9103682854@192.168.16.102:56912 SIP/2.0

Headers

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-f8728d218024a502-1---d8754z-;rport

Max-Forwards: 70

Route: <sip:192.168.16.140;lr>

Contact: <sip:9103683957@192.168.16.105:44646>

To: "Jane Doe"<sip:9103682854@192.168.16.140>;tag=f63e876c

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 3 BYE

Proxy-Authorization: Digest

username="9103683957",realm="192.168.16.140",nonce="2b028e9e929f845f097aa41196e511ac",uri="sip:9103682854@192.168.16.102:56912",response="4fe9122045c410cf3b5c2b2f828142e3",cnonce="ffec2900d9477f14248fcbb76b4d020

User-Agent: X-Lite release 1104o stamp 56125

Reason: SIP;description="User Hung Up"

Content-Length: 0

- **BYE #12**

Ο Proxy λαμβάνει ένα BYE request από τον John και το προωθεί στον user agent της Jane για να τερματιστεί το media session.

Request Line

BYE sip:9103682854@192.168.16.102:56912 SIP/2.0

Headers

Via: SIP/2.0/UDP 192.168.16.140:5060;branch=z9hG4bK- 85zf7828d7482d1

Via: SIP/2.0/UDP 192.168.16.105:44646;branch=z9hG4bK-d8754z-f8728d218024a502-1---d8754z-;received=192.168.16.105;rport=44646

Max-Forwards: 69

Contact: <sip:9103683957@192.168.16.105:44646

To: "Jane Doe"<sip:9103682854@192.168.16.140>;tag=f63e876c

From: "John Smith"<sip:9103683957@192.168.16.140>;tag=5c06d71d

Call-ID: Yzk2ZjQ5YzJhOWVmNTJmNjk5MWMxYmMxMmJiNzQwOTg.

CSeq: 3 BYE

Reason: SIP;description="User Hung Up"

Route: <sip:192.168.16.140;lr

Record-Route: <sip:192.168.16.140;lr

User-Agent: X-Lite release 1104o stamp 56125

Content-Length: 0

3.9 Συμπληρωματικά Πρωτόκολλα

Το SIP λειτουργεί παράλληλα με αλλά πρωτόκολλα όπως το RSVP (resource reservation protocol), RTP/RTCP (real time transport protocol/real time control protocol), RSTP (real time streaming protocol), SAP (session announcement protocol) και SDP (session description protocol). Το RTP/RTCP χρησιμοποιείται για τη μεταφορά δεδομένων πραγματικού χρόνου, το RSVP για την εξασφάλιση των απαραίτητων πόρων του δικτύου, το RSTP για τον έλεγχο της παράδοσης μιας ροής δεδομένων (stream), το SAP για τη δημοσιοποίηση ενός multimedia session και το SDP για την περιγραφή multimedia session. Σήμερα τα voice gateways αποτελούνται συνήθως από δυο τμήματα: την πύλη σηματοδότησης (signaling gateway) και την πύλη διαβίβασης φωνής, δεδομένων, εικόνας (media gateway). Τα δυο αυτά τμήματα

επικοινωνούν μεταξύ τους μέσω του MGCP (media gateway access protocol) που συνεργάζεται με το πρωτόκολλο SIP.

3.9.1 MGCP (media gateway control protocol)

Το MGCP είναι ένα πρωτόκολλο το οποίο χρησιμοποιείται από τους media gateway controllers (γνωστούς και ως call agents) για τη διαχείρηση των media gateways (MG). Το MGCP βασίζεται σε ένα master/slave μοντέλο, όπου ο MGC (master) δίνει εντολές σε ένα media gateway (slave). Ένας media gateway εκτελεί το request και ενημερώνει τον MGC για το αποτέλεσμα (επιτυχυμένο ή όχι). Στη συγκεκριμένη αρχιτεκτονική, ο MG διαχειρίζεται τα media functions, όπως η μετατροπή των TDM/αναλογικών σημάτων σε Real-time Transport Protocol (RTP)/Real-time Transport Control Protocol (RTCP) streams, ενώ ο MGC είναι υπεύθυνος για τα call-signaling functions. Ως πρωτόκολλο μεταφοράς για τα MGCP μηνύματα χρησιμοποιείτε το UDP, το οποίο δε διασφαλίζει την ασφαλή παράδοση τους και για αυτό ακριβώς το λόγο κάποια από αυτά μπορεί να επαναμεταδοθούν.

Το MGCP έχει βασιστεί σε δύο άλλα προυπάρχοντα πρωτόκολλα το Simple Gateway Control Protocol (SGCP) και το Internet Protocol Device Control (IPDC) και χρησιμοποιεί το Session Description Protocol (SDP) για να περιγράψει ένα media session. Το SDP περιέχει όλες τις media παραμέτρους για το session μεταξύ των media gateways, όπως οι IP addresses, τα UDP port, τα RTP profiles και multimedia conference capabilities. Αν και το SDP specification ορίζει διάφορα media types, το MGCP περιορίζεται στη χρήση των audio και των access circuits.

Τα MGCP/Megaco και SIP πρωτόκολλα μπορούν να συνυπάρξουν στα converged δίκτυα. Τα MGCP/Megaco δε αποτελούν ένα ολοκληρώμενο σύστημα, αντιθέτως το Session Initiation Protocol (SIP) απαιτείται για την επικοινωνία μεταξύ των gateway controllers. Ουσιαστικά το MGCP είναι ένα πρωτόκολλο το οποίο χρησιμοποιείται για την μετατροπή των σημάτων των τηλεφωνικών κυκλωμάτων σε IP πακέτα, ώστε να μεταφέρονται μέσω του Διαδικτύου ή άλλων packet-switched δικτύων.

Βασικά μηνύματα στο MGCP:

Μήνυμα	Περιγραφή
CreateConnection	Προσαρτεί σε μια διεύθυνση IP και σε μια θύρα ένα τερματικό σημείο. Αν η εντολή εκτελεστεί από την πύλη τότε στέλνει την ταυτότητα της σύνδεσης (connectionId) που μοναδικά προσδιορίζει μια σύνδεση.
ModifyConnection	Χρησιμοποιείται από τον ελεγκτή κλήσης για να αλλάξει τις παραμέτρους μιας υπάρχουσας σύνδεσης. Η απόκριση περιλαμβάνει μια λίστα με τις παραμέτρους της σύνδεσης.
DeleteConnection	Την εντολή αυτή τη χρησιμοποιεί ο ελεγκτής κλήσης ή η πύλη για να διαγράψει μια σύνδεση.
NotificationRequest	Αν ένας ελεγκτής κλήσης θέλει να ενημερωθεί όταν συμβεί κάποιο συγκεκριμένο γεγονός στο τερματικό σημείο που ελέγχει τότε στέλνει αυτή την εντολή στην πύλη. Όπως είπαμε το γεγονός μπορεί να είναι η κλήση αριθμού, το σήκωμα ή το κλείσιμο ακουστικού κ.α.
Notify	Η απάντηση σε μια εντολή NotificationRequest στέλνεται μέσω της εντολής notify που στέλνει η πύλη. Περιλαμβάνει μια λίστα με τα γεγονότα που «παρατήρησε» η πύλη.
AditEndpoint	Αυτή η εντολή χρησιμοποιείται από τον ελεγκτή κλήσης για να πάρει πληροφορίες για την κατάσταση κάποιων τερματικών σημείων.
AuditConnection	Για να πάρει πληροφορίες για μια συγκεκριμένη σύνδεση, ο ελεγκτής κλήσης στέλνει αυτή την εντολή. Ως παράμετρο δέχεται την ταυτότητα της σύνδεσης (connectionId).
RestartInProgress	Με την εντολή αυτή μια πύλη δηλώνει ότι ένα ή περισσότερα τερματικά σημεία τέθηκαν εκτός υπηρεσίας.

Πίνακας 16: MGCP μηνύματα

3.9.2 Real time transport protocol (RTP) & real time transport control protocol (RTCP)

Οι ιδιαιτερότητες της τηλεπικοινωνιακής κίνησης πραγματικού χρόνου υποχρέωσαν σε αναθεώρηση ή ακόμα και επανασχεδίαση κάποιων πλευρών των πρωτοκόλλων μεταφοράς.

- Διαφορετικά flow για κάθε ροή μέσου (media stream): Δεν είναι απαραίτητο όλα τα μέσα επικοινωνίας (εικόνα, ήχος κ.α.) μιας συνόδου να μεταφέρονται με τα ίδια πακέτα. Το αντίθετο συμβαίνει. Αυτό απλοποιεί και τη δομή των δεκτών. Επίσης μπορεί κάθε τύπος πακέτων να έχει διαφορετική ποιότητα εξυπηρέτησης (quality of service). Για παράδειγμα ένας διακομιστής σε περίπτωση συμφόρησης μπορεί να ρίξει τα πακέτα video ώστε να εξυπηρετηθούν καλύτερα τα πακέτα ήχου.
- Προσαρμογή αποδέκτη: Η τηλεπικοινωνιακή κίνηση που μεταφέρεται με τεχνικές μετάδοσης πακέτων με την κατά το δυνατόν καλύτερη προσπάθεια (best effort) καθυστερεί στους ενδιάμεσους διακομιστές. Ακόμα και κίνηση με μεγαλύτερη προτεραιότητα ίσως συναντήσει κάποιες μικρές καθυστερήσεις. Στην περίπτωση αυτή τα πακέτα που αποτελούν μια ροή θα έχουν διαφορετική καθυστέρηση. Αυτή η στατιστική διακύμανση της καθυστέρησης λέγεται jitter. Σε εφαρμογές πραγματικού χρόνου όπως το video και ο ήχος είναι απαραίτητο να διατηρήσουμε τα πακέτα σε μια προσωρινή μνήμη ώστε να εξαλειφθεί η στατιστική μεταβλητότητα καθυστέρησης. Για να γνωρίζουμε πόσο χρόνο πρέπει να μείνει στην προσωρινή μνήμη, κάθε πακέτο μεταφέρει πληροφορίες για το πότε δημιουργήθηκε. Πρέπει να σημειωθεί ότι δεν είναι απαραίτητο να ξέρουμε τον απόλυτο χρόνο αλλά το σχετικό χρόνο σε σχέση με τα άλλα πακέτα.
- Συγχρονισμός: Όπως είπαμε η ροή πακέτων ήχου και video δέχεται διαφορετικές καθυστερήσεις και ίσως και διαφορετική ποιότητα εξυπηρέτησης (quality of service). Αυτό θα έχει σαν αποτέλεσμα πακέτα που δημιουργήθηκαν την ίδια στιγμή να φτάσουν στον παραλήπτη σε διαφορετικούς χρόνους. Με την προσωρινή αποθήκευση μπορούμε να απομακρύνουμε την καθυστέρηση σε κάθε ροή πακέτων (flow). Πρέπει να

υπάρχει πρόβλεψη ώστε πακέτα που δημιουργήθηκαν μαζί και είναι διαφορετικής ροής να εκτελούνται ταυτόχρονα.

Το RTP υποστηρίζει τη μεταφορά δεδομένων πραγματικού χρόνου (ήχος & video) πάνω από δίκτυα μεταγωγής πακέτου. Χρησιμοποιείται από το H.323 και το SIP. Το πρωτόκολλο μεταφοράς πρέπει να επιτρέπει στον αποδέκτη να ανιχνεύει απώλειες πακέτων αλλά και να παρέχει τις κατάλληλες πληροφορίες χρονισμού ώστε ο αποδέκτης να μπορεί να εξουδετερώσει τη στατιστική μεταβλητότητα καθυστέρησης. Δεν παρέχει όμως κανένα μέσο εξασφάλισης πόρων δικτύου. Το RTP παρέχει τις ακόλουθες λειτουργίες:

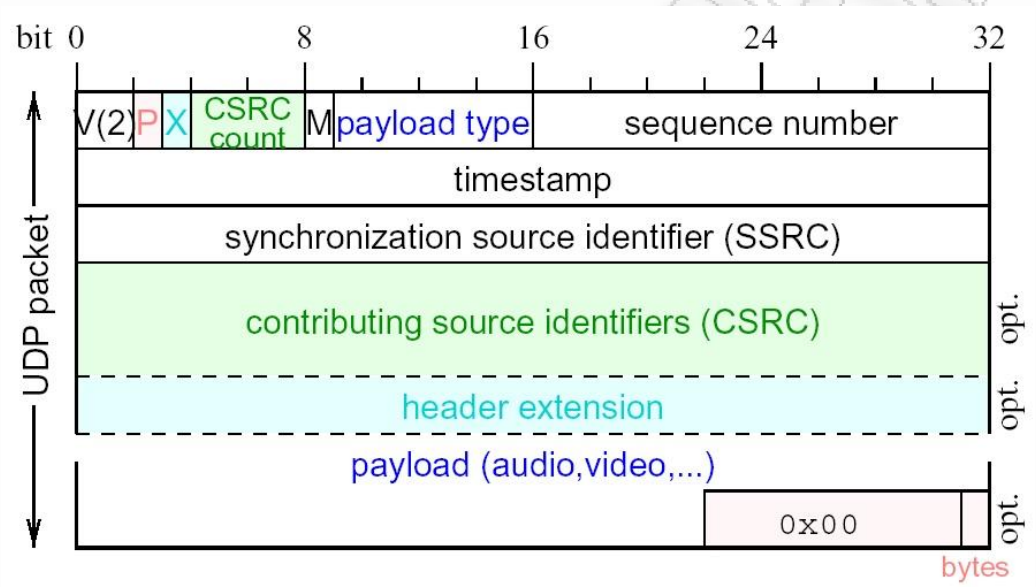
- Διαδοχή: Κάθε πακέτο RTP έχει έναν αριθμό που χρησιμοποιείται για την ανίχνευση χαμένων πακέτων.
- Αναγνώριση φορτίου: Στο internet είναι συχνά απαραίτητο να αλλάξει δυναμικά η κωδικοποίηση ανάλογα με το εύρος ζώνης. Για να επιτευχθεί αυτό κάθε πακέτο περιέχει την ταυτότητα του φορτίου (payload identifier).
- Ένδειξη πλαισίου: Το video και ο ήχος στέλνονται σε λογικά τμήματα που λέγονται πλαίσια (frames). Η αναγνώριση της αρχής και του τέλους του πλαισίου γίνεται με τη χρήση ενός bit που λέγεται σημειωτής πλαισίου (frame marker bit).
- Αναγνώριση πηγής: Σε μια εκπομπή προς πολλούς αποδεκτές (multicast) είναι απαραίτητη η γνώση του αποστολέα. Αυτό γίνεται με το δείκτη συγχρονισμού πηγής (synchronization source identifier SSRC).

Το RTCP είναι ένα πρωτόκολλο ελέγχου και λειτουργεί σε συνδυασμό με το RTP. Σε μια σύνοδο RTP (RTP session) οι συμμετέχοντες στέλνουν περιοδικά πακέτα RTCP για να πάρουν πληροφορίες για την ποιότητα εξυπηρέτησης (QoS) και αλλά. Οι υπηρεσίες που παρέχει είναι:

- Ενημέρωση για την ποιότητα εξυπηρετήσεις (QoS feedback): Το RTCP χρησιμοποιείται για να ενημερώνει για την ποιότητα εξυπηρέτησης. Στις πληροφορίες που παρέχονται περιλαμβάνονται ο αριθμός χαμένων πακέτων, η στατιστική μεταβλητότητα καθυστέρησης, ο συνολικός χρόνος μετάδοσης μετ' επιστροφής (round trip time) κ.α. Με βάση τις πληροφορίες αυτές ρυθμίζεται ο ρυθμός μετάδοσης των δεδομένων (data rate).

- Έλεγχος συνόδου: Με το πακέτο BYE το RTCP επιτρέπει στους συμμετέχοντες να ενημερώνουν όταν αφήνουν μια σύνοδο.
- Αναγνώριση: Πληροφορίες όπως διεύθυνση email, όνομα και αριθμός τηλεφώνου περιέχονται σε πακέτα RTCP ώστε όλοι οι χρήστες να ξέρουν την «ταυτότητα» των άλλων συμμετεχόντων.

Μορφή πακέτου RTP



Σχήμα 20: Δομή RTP πακέτου

Τα πρώτα 32 bits βρίσκονται σε κάθε πακέτο

- **Έκδοση (V)**: 2 bits

  Το πεδίο αυτό δείχνει την έκδοση του RTP.

- **Συμπλήρωση (Padding) (P)**: 1 bit

  Αν το πεδίο αυτό έχει τιμή 1, το πακέτο περιέχει μια ή περισσότερες οκτάδες bits στο τέλος που δεν είναι μέρος του φορτίου.

- **Προέκταση (X)**: 1 bit

Αν το πεδίο προέκτασης έχει τιμή 1, την επικεφαλίδα την ακολουθεί μια επικεφαλίδα-προέκταση.

- **Μετρητής CSRC (CC)**: 4 bits

  Ο μετρητής είναι ίσος με τον αριθμό των δεικτών CSRC που ακολουθούν την επικεφαλίδα.

- **Δείκτης (M)**: 1 bit

  Σημειώνει τα όρια ενός πλαισίου.

- **Τύπος φορτίου (PT)**: 7 bits

  Δείχνει τη μορφή του φορτίου RTP και καθορίζει τον τρόπο που θα ερμηνευτεί από την εφαρμογή.

- **Αριθμός ακολουθίας (sequence number)**: 16 bits

  Αυξάνεται κατά ένα για κάθε πακέτο RTP που στέλνεται και χρησιμοποιείται από το δέκτη για την ανίχνευση χαμένων πακέτων.

- **Χρονικό αποτύπωμα (timestamp)**: 32 bits

  Αντανακλά τη χρονική στιγμή που δημιουργήθηκε το πρώτο δείγμα που μεταφέρει το πακέτο.

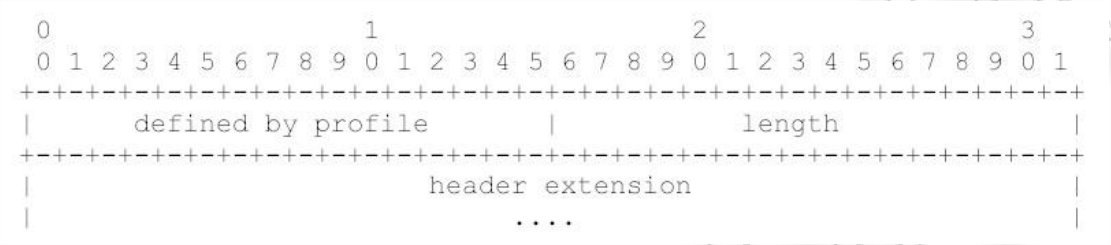
- **Δείκτης συγχρονισμού πηγής (SSRC)**: 32 bits

  Χρησιμοποιείται για το συγχρονισμό των πακέτων.

- **CSRC**: 0-15 στοιχεία από 32 bits το καθένα

  Ο αριθμός των στοιχείων δίνεται στο πεδίο CC. Η λίστα αυτή περιγράφει τις πηγές που συνεισφέρουν στο φορτίο του πακέτου.

Παρέχεται ένα μηχανισμός προέκτασης ο οποίος επιτρέπει ανεξάρτητες υλοποιήσεις που πειραματίζονται με νέες μεθόδους ανεξάρτητες από το φορτίο. Τις περισσότερες φορές είναι απαραίτητα επιπλέον πεδία για την επικεφαλίδα.



Σχήμα 21: Header extension

Αν το πεδίο X στην επικεφαλίδα είναι 1, τότε τοποθετείται μια μεταβλητού μήκους προέκταση μετά τα πεδία CSRC αν υπάρχουν. Περιλαμβάνει ένα πεδίο μήκους 16 bits που έχει τον αριθμό των πεδίων μήκους 32 bits. Σε αυτά δεν περιλαμβάνονται τα πρώτα 32 bits.

4ο ΚΕΦΑΛΑΙΟ

Εργαστηριακή Άσκηση πάνω σε ένα απλό VOIP δίκτυο

4.1 Εισαγωγή

Στόχος της προκείμενης εργαστηριακής άσκησης είναι να στηθεί ένα απλό εργαστηριακό περιβάλλον που θα επιτρέψει τη μελέτη κάποιων βασικών χαρακτηριστικών και των βασικών call flows σε μια SIP το SIP συνδιαλλαγή. Μια μικρογραφία ενός VOIP δικτύου μπορεί απλά να περιλαμβάνει δύο User Agents οι οποίοι ανταλλάσουν SIP μηνύματα μέσω ενός διακομιστή. Στη θέση του διακομιστή θα χρησιμοποιηθεί το open source λογισμικό ASTERISK.

Τυπικά ένας ASTERISK είναι μια open sourse TDM, PBX και IVR πλατφόρμα με ACD ικανότητες. Ωστόσο άτυπα ο ASTERISK αποτελεί ένα πανίσχυρο, ευέλικτο και επεκτάσιμο κομμάτι λογισμικού για τηλεπικοινωνιακά συστήματα. Το όνομα του προέρχεται από το σύμβολο του αστερίσκου που σε όλα τα UNIX (συμμεριλαμβανομένου και του Linux) και DOS περιβάλλοντα αντιπροσωπεύει ένα σύμβολο υποκατάστασης το οποίο αντιστοιχεί σε οποιοδήποτε όνομα αρχείου. Έτσι ακριβώς και το Asrerisk PBX έχει σχεδιαστεί με τέτοιο τρόπο ώστε να παρέχει interface σε οποιοδήποτε hardware και software. Παραδοσιακά, τα προιόντα τηλεφωνίας σχεδιάζονται έτσι ώστε να ικανοποιούν κάποια συγκεκριμένη τεχνική απαραίτητη στο δίκτυο. Ωστόσο το Asterisk υποστηρίζει πολλές εφαρμογές που χρησιμοποιούν πληθώρα τεχνολογιών και προσφέρει ένα κοινό περιβάλλον το οποίο μπορεί να συνδυάσει διαφορετικές εφαρμογές ανάλογα με την επιθυμία του χρήστη.

Επομένως το asterisk μπορεί να χρησιμοποιηθεί σαν ένα παραδοσιακό τηλεφωνικό κέντρο πάνω από το internet (IPBX), το οποίο δύναται να διαχειρίζεται και να δρομολογεί τηλεφωνικές κλήσεις αξιοποιώντας μια σειρά από χαρακτηριστικά που περιλαμβάνονται στον κώδικα πηγής κάθε έκδοσης. Μπορεί να παίξει το ρόλο ενός

SIP διακομιστή, διαμεσολαβητή ή Registrar, ενώ ταυτόχρονα μπορεί να χρησιμοποιηθεί σαν διακομηστής διαχείρισης πολυμεσικών ροών.

Ακολουθούν κάποιες εφαρμογές στις οποίες μπορεί να χρησιμοποιηθεί ο Asterisk:

- Heterogeneous Voice over IP gateway (MGCP, SIP, IAX, H.323)
- Private Branch eXchange (PBX)
- Custom Interactive Voice Response (IVR) server
- Softswitch
- Conferencing server
- Number translation
- Calling card application
- Predictive dialer
- Call queuing with remote agents
- Remote offices for existing PBX

Πρωτόκολλα φωνής όπως τα Session Initiation Protocol (SIP), Inter-Asterisk eXchange (IAX) versions 1 and 2, Media Gateway Control Protocol (MGCP), ITU H.3232, Voice over Frame Relay (VOFR), τα οποία χρησιμοποιούνται για επικοινωνία πάνω από IP and Frame Relay δίκτυα και είναι τα μοναδικά interfaces τα οποία δε απαιτούν κάποιο εξειδικευμένο hardware υποστηρίζονται από τον Asterisk. Πάνω σε αυτή τη δυνατότητα του Asterisk θα βασιστούμε στην παρούσα έργαστηριακή άσκηση.

4.2 Αρχιτεκτονική δομή της άσκησης

4.2.1 User Agents

Στη συγκεκριμένη εργαστηριακή άσκηση χρησιμοποιήθηκε ως User Agent το open source λογισμικό X-lite. Υπάρχουν πολλά open source softphones, ώστοσο η απλότητα χρήσης του συγκεκριμένου ήταν και ο κύριος λόγος για την επιλογή του. Το συγκεκριμένο λογισμικό τρέχει πάνω σε ένα απλό προσωπικό υπολογιστή με windows περιβάλλον και συνδιάζει απλές τηλεφωνικές κλήσεις με βίντεο κλήσεις και υπηρεσίες Instant Messaging σε ένα απλό interface.

4.2.2 Asterisk

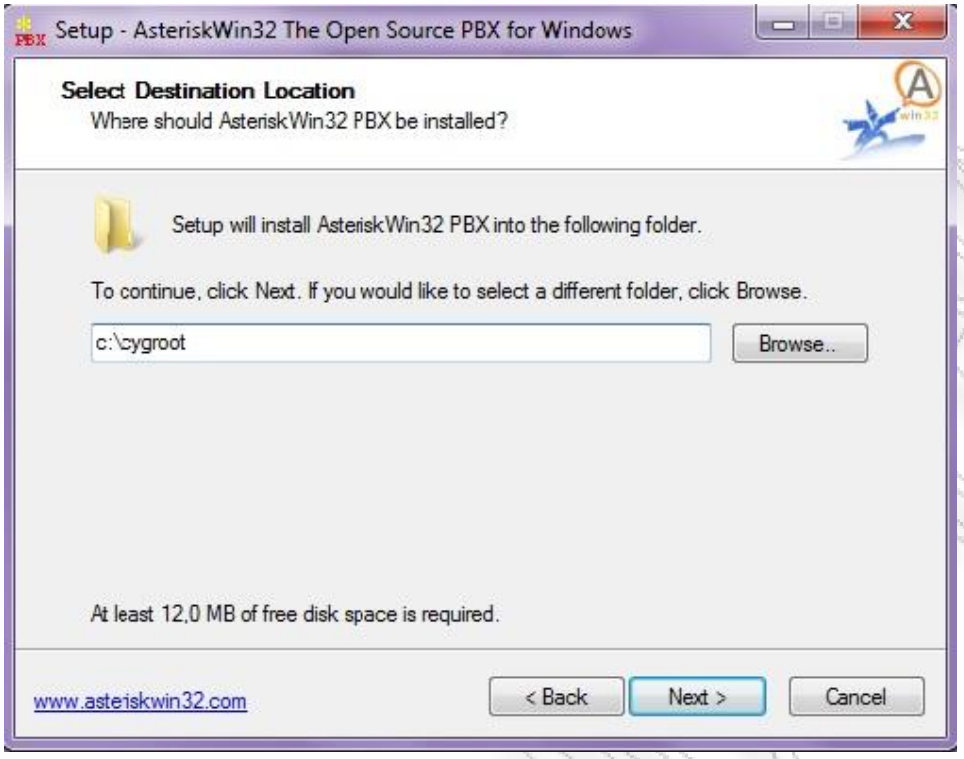
Ο Asterisk όπως αναφέραμε ήδη, είναι ένα λογισμικό open source το οποίο στη συγκεκριμένη περίπτωση θα παίξει το ρόλο του SIP Proxy server. Ως proxy server ο asterisk θα διαχειρίζεται τις κλήσεις που έρχονται από ένα UA ο οποίος είναι register στο συγκεκριμένο proxy server. Επομένως εκτός από Proxy ο asterisk θα παιζει και το ρόλο του Registrar Server. Κάθε User agent έχει το δικό του user name και password, τα οποία αποτελούν και τα βασικά credentials για να γίνει authorized ο συγκεκριμένος χρήστης έτσι ώστε να μπορεί να πραγματοποιεί κλήσεις μέσω του συγκεκριμένου proxy. Σαν Registrar server ο Asterisk αποθηκεύει στη βάση του τα credentials κάθε συνδρομητή.

Στη συγκεκριμένη εργαστηριακή άσκηση ο Asterisk τρέχει σε windows περιβάλλον. Ωστόσο το συγκεκριμένο λογισμικό μπορεί να τρέχει πάνω από ένα πλήθος λειτουργικών συστημάτων όπως: GNU/Linux, FreeBSD, OpenBSD και Mac OS X.

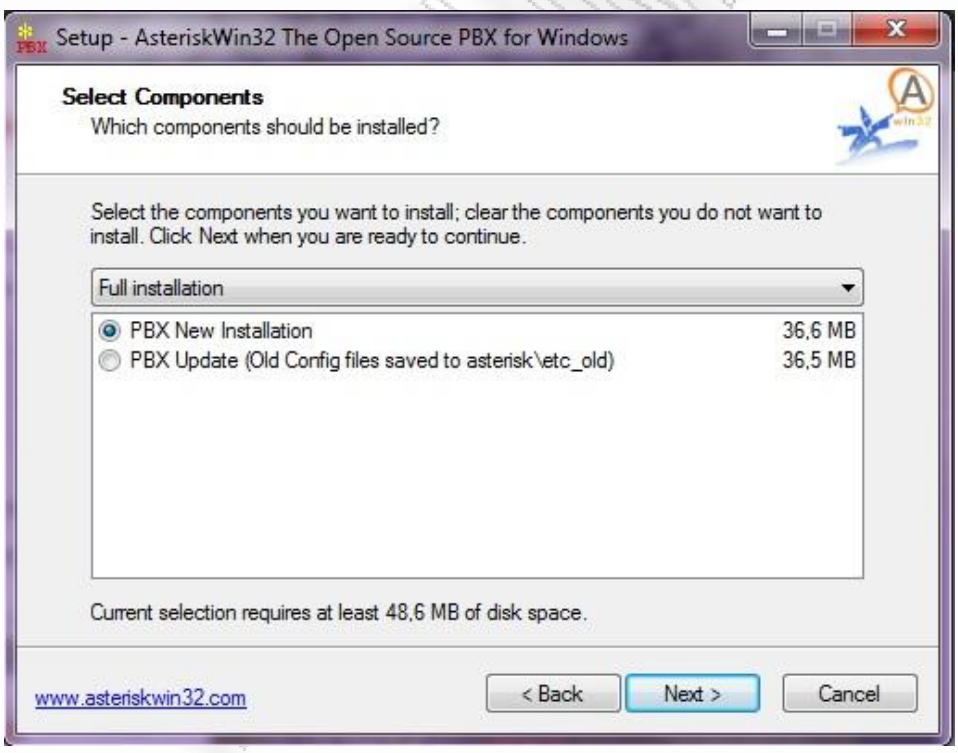
4.3 Προεργασία εργαστηριακής Άσκησης

- **Εγκατάσταση του Asterisk**

Το AsteriskWin32-0.60-Setup είναι freeware και πολύ απλό στην εγκατάστασή του. Τρέχουμε το setup και επιλέγουμε **New PBX Installation** όπως φαίνετε χαρακτηριστικά και στα παρακάτω screenshots:

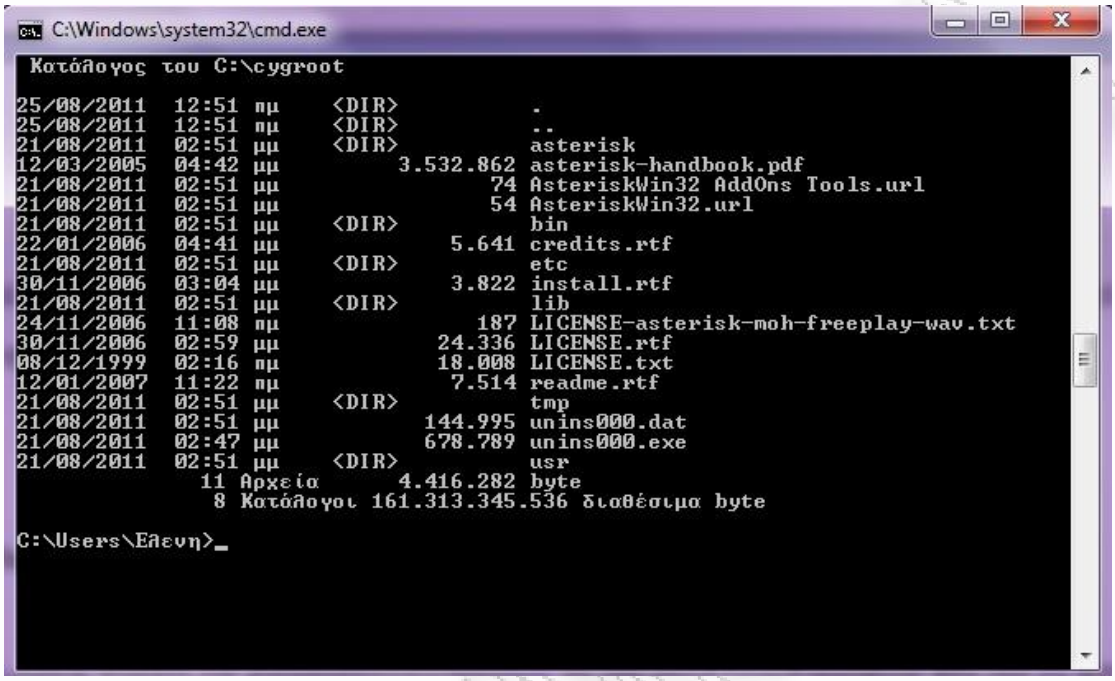


Σχήμα 22



Σχήμα 23

Μετά την εγκατάσταση του Asterisk στον προσωπικό σας υπολογιστή δημιουργείτε ένας φάκελος c:\cygroot με όλα τα configuration αρχεία του Asterisk.



Σχήμα 24:Τα περιεχόμενα του c:\cygroot μετά την εγκατάσταση του Asterisk.

- **Βήματα σεταρίσματος του Asterisk:**

1. Αρχικά πρέπει να παραμετροποιήσουμε το sip.conf (configuration file για το SIP) το οποίο βρίσκεται στο παρακάτω path C:\cygroot\asterisk\etc. Στο συγκεκριμένο αρχείο πρέπει να ορίσουμε σε πρώτη φάση το bindport ανάλογα με το πρωτόκολλο μεταφοράς που θα χρησιμοποιήσουμε και την bindaddress. Επίσης στο sip.conf και πιο συγκεκριμένα στο authentication section, θα δηλώσουμε τους χρήστες που θα γίνουν register πάνω στο συγκεκριμένο Asterisk και θα έχουν τα παρακάτω χαρακτηριστικά:

   [3000] → user 3000

   type = friend →ο τύπος του SIP account δηλώνετε ως friend

   context = default

   username = 3000

   host = dynamic

mailbox = 3000

dtmfmode = rfc2833

Το **sip.conf** που χρησιμοποιήσαμε είναι το παρακάτω:

[general]

context = demo ; Default context for incoming calls

; Allow or reject guest calls (default is yes, this can also be set to 'osp'

; if asterisk was compiled with OSP support.

; Realm for digest authentication

; defaults to "asterisk"

; Realms MUST be globally unique according to RFC 3261

; Set this to your host name or domain name

bindport = 5060 ; UDP Port to bind to (SIP standard port is 5060)

bindaddr = 192.168.1.3 ; IP address to bind to (0.0.0.0 binds to all)

srvlookup = no ; Enable DNS SRV lookups on outbound calls

; Note: Asterisk only uses the first host

; in SRV records

; Disabling DNS SRV lookups disables the

; ability to place SIP calls based on domain

; names to some other SIP users on the Internet

; Set default domain for this host

; If configured, Asterisk will only allow

; INVITE and REFER to non-local domains

; Use "sip show domains" to list local domains

;domain=mydomain.tld,mydomain-incoming

; Add domain and configure incoming context

; for external calls to this domain

; Add IP address as local domain

; You can have several "domain" settings

; Disable INVITE and REFER to non-local domains

; Default is yes

; Turn this on to have Asterisk add local host

; name and local IP to domain list.

; Enable slow, pedantic checking for Pingtel

; and multiline formatted headers for strict

; SIP compatibility (defaults to "no")

; Set IP QoS to either a keyword or numeric val

; lowdelay,throughput,reliability,mincost,none

; Max length of incoming registration we allow

; Default length of incoming/outgoing registration

; Allow overriding of mime type in MWI NOTIFY

; Default time between mailbox checks for peers

; dialplan extension to reach mailbox sets the

; Message-Account in the MWI notify message

; defaults to "asterisk"

; Turn on support for SIP video

; Record SIP history by default

; (see sip history / sip no history)

; First disallow all codecs

; Allow codecs in order of preference

; Sets the default music on hold class for all SIP calls

; This may also be set for individual users/peers

; Default language setting for all users/peers

; This may also be set for individual users/peers

; Relax dtmf handling

; Terminate call if 60 seconds of no RTP activity

; when we're not on hold

; Terminate call if 300 seconds of no RTP activity

; when we're on hold (must be > rtptimeout)

; If Remote-Party-ID should be trusted

; If Remote-Party-ID should be sent

; If we should generate in-band ringing always

; use 'never' to never use in-band signalling, even in cases

; where some buggy devices might not render it

; Valid values: yes, no, never Default: never

; Allows you to change the user agent string

; If yes, allows 302 or REDIR to non-local SIP address

; Note that promiscredir when redirects are made to the

; local system will cause loops since SIP is incapable

; of performing a "hairpin" call.

;user=phone" is added to uri that contains

; a valid phone number

; Set default dtmfmode for sending DTMF. Default: rfc2833

; Other options:

; info : SIP INFO messages

; inband : Inband audio (requires 64 kbit codec -alaw, ulaw)

; auto : Use rfc2833 if offered, inband otherwise

; send compact sip headers.

; Turn on SIP debugging by default, from

; the moment the channel loads this configuration

; Set a specific context for SUBSCRIBE requests

; Useful to limit subscriptions to local extensions

; Settable per peer/user also

; Notify subscriptions on RINGING state

; When an incoming INVITE or REGISTER is to be rejected,

; for any reason, always reject with '401 Unauthorized'

; instead of letting the requester know whether there was

; a matching user or peer for their request

; If regcontext is specified, Asterisk will dynamically create and destroy a

; NoOp priority 1 extension for a given peer who registers or unregisters with

; us. The actual extension is the 'regexten' parameter of the registering

; peer or its name if 'regexten' is not provided. More than one regexten may

; be supplied if they are separated by '&'. Patterns may be used in regexten.

;

;regcontext=sipregistrations

;

; Asterisk can register as a SIP user agent to a SIP proxy (provider)

; Format for the register statement is:

; register => user[:secret[:authuser]]@host[:port][/extension]

;

; If no extension is given, the 's' extension is used. The extension needs to

; be defined in extensions.conf to be able to accept calls from this SIP proxy

; (provider).

;

; host is either a host name defined in DNS or the name of a section defined

; below.

;

; Examples:

;

```
;register => 1234:password@mysipprovider.com
;
;     This will pass incoming calls to the 's' extension
;
;register => 2345:password@sip_proxy/1234
;
;    Register 2345 at sip provider 'sip_proxy'.  Calls from this provider
;    connect to local extension 1234 in extensions.conf, default context,
;    unless you configure a [sip_proxy] section below, and configure a
;    context.
;    Tip 1: Avoid assigning hostname to a sip.conf section like [provider.com]
;    Tip 2: Use separate type=peer and type=user sections for SIP providers
;           (instead of type=friend) if you have calls in both directions
; retry registration calls every 20 seconds (default)
; Number of registration attempts before we give up
; 0 = continue forever, hammering the other server until it
; accepts the registration
; Default is 0 tries, continue forever
; generate manager events when sip ua performs events (e.g. hold)
;----------------------------------------- NAT SUPPORT ------------------------
; The externip, externhost and localnet settings are used if you use Asterisk
; behind a NAT device to communicate with services on the outside.
; Address that we're going to put in outbound SIP messages
; if we're behind a NAT
; The externip and localnet is used
; when registering and communicating with other proxies
; that we're registered with
```

```
; Alternatively you can specify an
; external host, and Asterisk will
; perform DNS queries periodically.  Not
; recommended for production
; environments!  Use externip instead
; How often to refresh externhost if
; used
; You may add multiple local networks.  A reasonable set of defaults
; are:
; All RFC 1918 addresses are local networks
; Also RFC1918
; Another RFC1918 with CIDR notation
;Zero conf local network
; The nat= setting is used when Asterisk is on a public IP, communicating with
; devices hidden behind a NAT device (broadband router).  If you have one-way
; audio problems, you usually have problems with your NAT configuration or your
; firewall's support of SIP+RTP ports.  You configure Asterisk choice of RTP
; ports for incoming audio in rtp.conf
;
; Global NAT settings  (Affects all peers and users)
; yes = Always ignore info and assume NAT
; no = Use NAT mode only according to RFC3581
; never = Never attempt NAT mode or RFC3581 support
; route = Assume NAT, don't send rport
; (work around more UNIDEN bugs)
; Cache realtime friends by adding them to the internal list
; just like friends added from the config file only on a
```

; as-needed basis? (yes|no)

; Send registry updates to database using realtime? (yes|no)

; If set to yes, when a SIP UA registers successfully, the ip address,

; the origination port, the registration period, and the username of

; the UA will be set to database via realtime. If not present, defaults to 'yes'.

; Auto-Expire friends created on the fly on the same schedule

; as if it had just registered? (yes|no|<seconds>)

; If set to yes, when the registration expires, the friend will vanish from

; the configuration until requested again. If set to an integer,

; friends expire within this number of seconds instead of the

; registration interval.

; Enabling this setting has two functions:

;

; For non-realtime peers, when their registration expires, the information

; if you attempt

; to place a call to the peer, the existing information will be used in spite

; of it having expired

;

; For realtime peers, when the peer is retrieved from realtime storage,

; the registration information will be used regardless of whether

; if it expires while the realtime peer is still in

; memory (due to caching or other reasons), the information will not be

; removed from realtime storage

; Incoming INVITE and REFER messages can be matched against a list of 'allowed'

; domains, each of which can direct the call to a specific context if desired.

; By default, all domains are accepted and sent to the default context or the

; context associated with the user/peer placing the call.

; Domains can be specified using:

; domain=<domain>[,<context>]

; Examples:

; domain=myasterisk.dom

; domain=customer.com,customer-context

;

; In addition, all the 'default' domains associated with a server should be

; added if incoming request filtering is desired.

; autodomain=yes

;

; To disallow requests for domains not serviced by this server:

; allowexternaldomains=no

; When making outbound SIP INVITEs to

; non-peers, use your primary domain "identity"

; for From: headers instead of just your IP

; address. This is to be polite and

; it may be a mandatory requirement for some

; destinations which do not have a prior

; account relationship with your server.

[authentication]

[3000]

type = friend

context = default

username = 3000

host = dynamic

mailbox = 3000

dtmfmode = rfc2833

[3001]

type = friend

context = default

username = 3001

host = dynamic

mailbox = 3001

dtmfmode = rfc2833

[3002]

type = friend

username = 3002

context = default

host = dynamic

mailbox = 3002

dtmfmode = rfc2833

[3003]

type = friend

username = 3003

context = default

host = dynamic

mailbox = 3003

; Global credentials for outbound calls, i.e. when a proxy challenges your

; Asterisk server for authentication. These credentials override

; any credentials in peer/register definition if realm is matched.

;

; This way, Asterisk can authenticate for outbound calls to other

; realms. We match realm on the proxy challenge and pick an set of

; credentials from this list

; Syntax:

; auth = <user>:<secret>@<realm>

; auth = <user>#<md5secret>@<realm>

; Example:

;auth=mark:topsecret@digium.com

;

; You may also add auth= statements to [peer] definitions

; Peer auth= override all other authentication settings if we match on realm

;--

; Users and peers have different settings available. Friends have all settings,

; since a friend is both a peer and a user

;

; User config options: Peer configuration:

; ---------- ----------

; context context

; permit permit

; deny deny

; secret secret

; md5secret md5secret

; dtmfmode dtmfmode

; canreinvite canreinvite

; nat nat

; callgroup callgroup

; pickupgroup pickupgroup

; language language

; allow allow

; disallow disallow

; insecure insecure

; trustrpid trustrpid

; progressinband progressinband

; promiscredir promiscredir

; useclientcode useclientcode

; accountcode accountcode

; setvar setvar

; callerid callerid

; amaflags amaflags

; call-limit call-limit

; restrictcid restrictcid

; subscribecontext subscribecontext

; mailbox

; username

; template

; fromdomain

; regexten

; fromuser

; host

; port

; qualify

; defaultip

; rtptimeout

; rtpholdtimeout

; sendrpid

; outboundproxy

;[sip_proxy]

```
; For incoming calls only. Example: FWD (Free World Dialup)
; We match on IP address of the proxy for incoming calls
; since we can not match on username (caller id)
;type=peer
;context=from-fwd
;host=fwd.pulver.com
;[sip_proxy-out]
; we only want to call out, not be called
;secret=guessit
; Authentication user for outbound proxies
; Many SIP providers require this!
;fromdomain=provider.sip.domain
;host=box.provider.com
;user=phone" on URI
; permit only 5 simultaneous outgoing calls to this peer
; send outbound signaling to this proxy, not directly to the peer
;------------------------------------------------------------------------
; Definitions of locally connected SIP phones
;
; type = user        a device that authenticates to us by "from" field to place calls
; type = peer        a device we place calls to or that calls us and we match by host
; type = friend two configurations (peer+user) in one
;
; For local phones, type=friend works most of the time
;
; If you have one-way audio, you probably have NAT problems.
; If Asterisk is on a public IP, and the phone is inside of a NAT device
```

```
; you will need to configure nat option for those phones.
; Also, turn on qualify=yes to keep the nat session open
;[grandstream1]
;type=friend
; Where to start in the dialplan when this phone calls
; Full caller ID, to override the phones config
; we have a static but private IP address
; No registration allowed
; there is not NAT between phone and Asterisk
; allow RTP voice traffic to bypass Asterisk
; either RFC2833 or INFO for the BudgeTone
; permit only 1 outgoing call and 1 incoming call at a time
; from the phone to asterisk
; (1 for the explicit peer, 1 for the explicit user,
; remember that a friend equals 1 peer and 1 user in
; memory)
; mailbox 1234 in voicemail context "default"
; need to disallow=all before we can use allow=
; Note: In user sections the order of codecs
; listed with allow= does NOT matter!
;allow=alaw
; Asterisk only supports g723.1 pass-thru!
; Pass-thru only unless g729 license obtained
;[xlite1]
; Turn off silence suppression in X-Lite ("Transmit Silence"=YES)!
; Note that Xlite sends NAT keep-alive packets, so qualify=yes is not needed
;type=friend
```

```
; When they register, create extension 1234
;callerid="Jane Smith" <5678>
; This device needs to register
; X-Lite is behind a NAT router
; Typically set to NO if behind NAT
;disallow=all
; GSM consumes far less bandwidth than ulaw
;allow=ulaw
;allow=alaw
; Subscribe to status of multiple mailboxes
;[snom]
; Friends place calls and receive calls
; Context for incoming calls from this user
;secret=blah
; Only allow SUBSCRIBE for local extensions
; Use German prompts for this user
; This peer register with us
; Choices are inband, rfc2833, or info
; IP used until peer registers
; Mailbox(-es) for message waiting indicator
; dialplan extension to reach mailbox
; sets the Message-Account in the MWI notify message
; defaults to global vmexten which defaults to "asterisk"
; To have the callerid restriced -> sent as ANI
;disallow=all
; dtmfmode=inband only works with ulaw or alaw!
;[polycom]
```

```
; Friends place calls and receive calls
; Context for incoming calls from this user
;secret=blahpoly
; This peer register with us
; Choices are inband, rfc2833, or info
; Username to use in INVITE until peer registers
; Normally you do NOT need to set this parameter
;disallow=all
; dtmfmode=inband only works with ulaw or alaw!
; Polycom phones don't work properly with "never"
;[pingtel]
;type=friend
;secret=blah
;host=dynamic
; Allow matching of peer by IP address without matching port number
; Do not require authentication of incoming INVITEs
; (both)
; Consider it down if it's 1 second to reply
; Helps with NAT session
; qualify=yes uses default value
;
; Call group and Pickup group should be in the range from 0 to 63
;
; We are in caller groups 1,3,4
; We can do call pick-p for call group 1,3,4,5
; IP address to use if peer has not registered
;[cisco1]
```

;type=friend

;secret=blah

; Qualify peer is no more than 200ms away

; This phone may be natted

; Send SIP and RTP to the IP address that packet is

; received from instead of trusting SIP headers

; This device registers with us

; Asterisk by default tries to redirect the

; RTP media stream (audio) to go directly from

; the caller to the callee. Some devices do not

; support this (especially if one of them is

; behind a NAT).

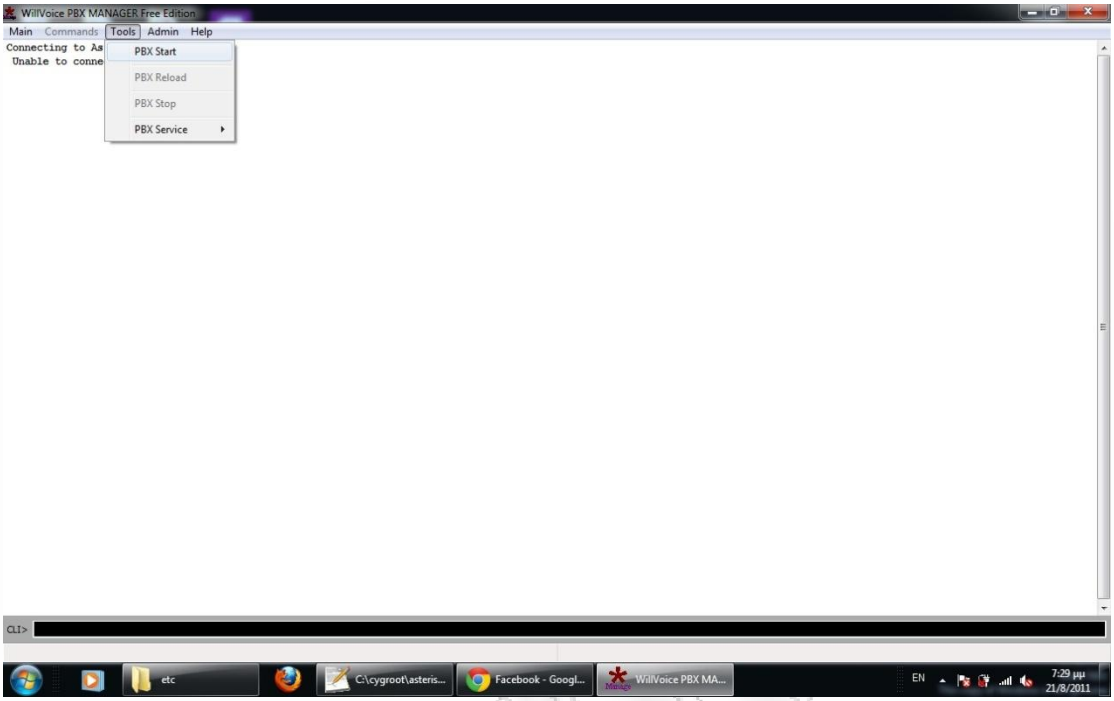
; IP address to use until registration

; Username to use when calling this device before registration

; Normally you do NOT need to set this parameter

; Channel variable to be set for all calls from this device

2. Το επόμενο βήμα είναι να ξεκινήσουμε τον Asterisk. Γι' αυτόν ακριβώς το λόγο πρέπει να ξεκινήσουμε την κονσόλα του asterisk. Αφού ξεκινήσουμε την κονσόλα από το μενού Tools επιλέγουμε PBX start, όπως φαίνετε και στο παρακάτω σχήμα:

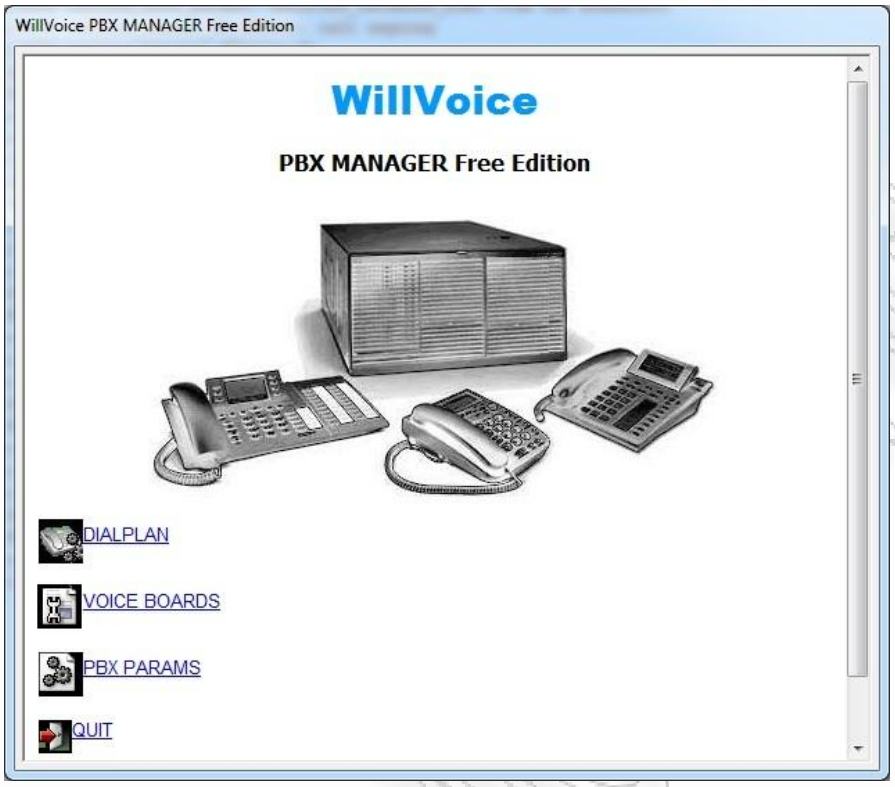


Σχήμα 25: PBX start

Ένα μήνυμα στη κονσόλα του PBX μας πιστοποιεί ότι ο server έχει ξεκινήσει σωστά και είναι έτοιμος να χρησιμοποιηθεί:

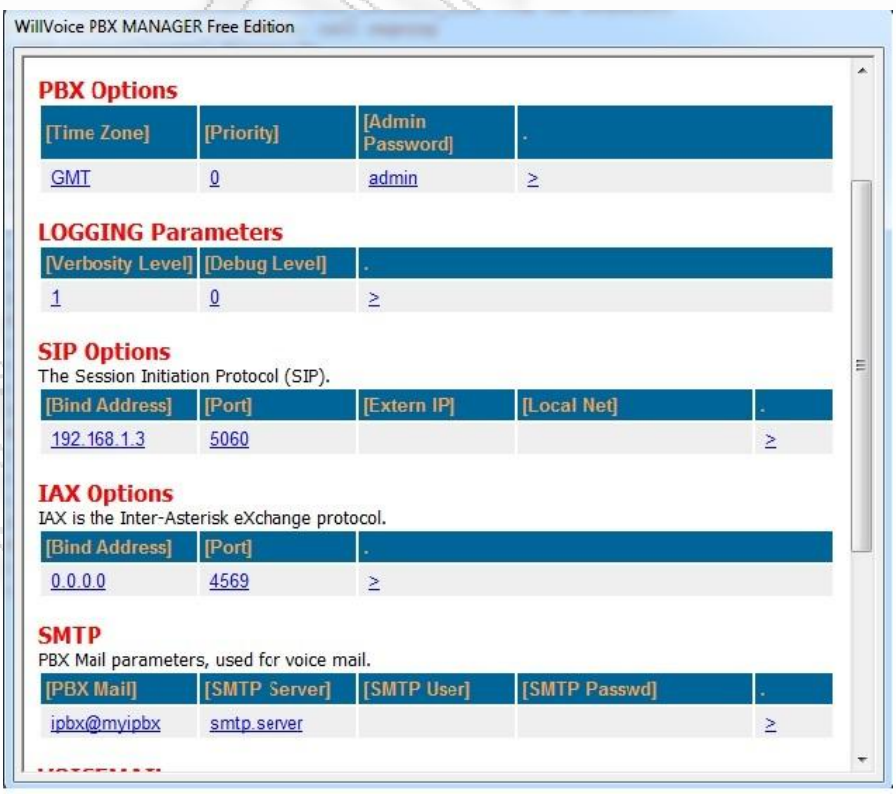
CLI> Asterisk Ready

3. Το PBX διαθέτει και ένα αρκετά εύχρηστο GUI, το οποίο μας επιτρέπει να αλλάξουμε τις παραμέτρους που θέλουμε χωρίς να ανατρέχουμε συνέχεια στο sip.conf. Για να ξεκινήσουμε το PBX manager από το μενού Admin επιλέγουμε PBX manager. Το default password είναι **admin**.



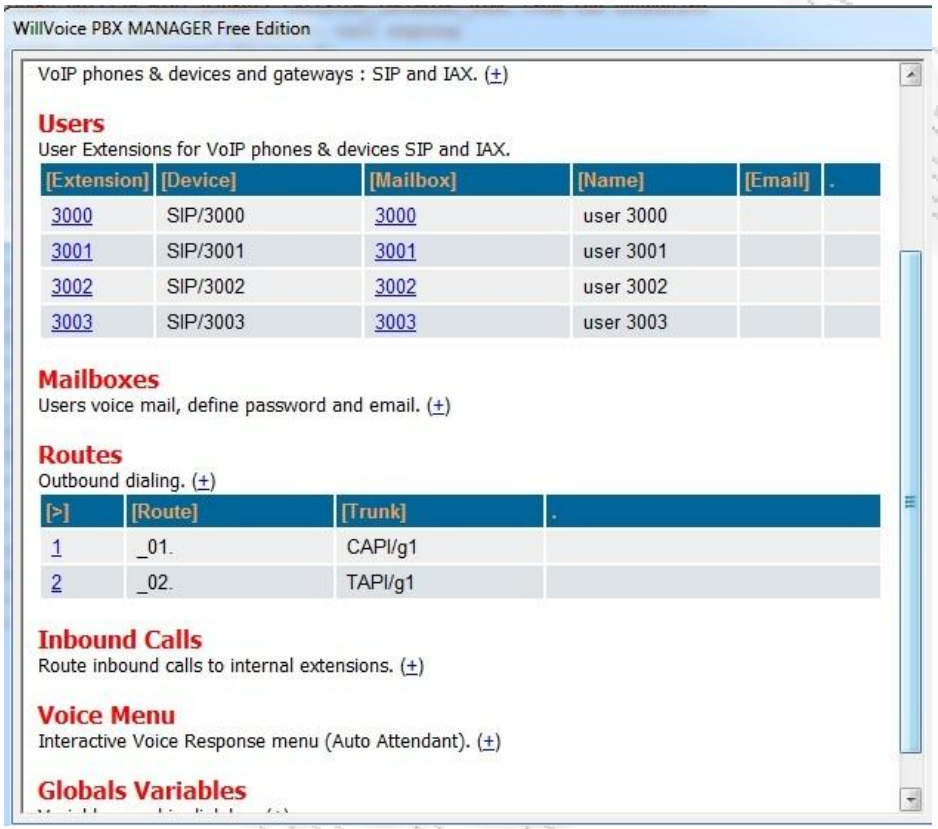
Σχήμα 26: Asterisk GUI

Αν περιηγηθούμε στο μενού, βλέπουμε ότι στο PBX params tab μπορούμε να τροποποιήσουμε τα PBX options, τα SIP options κτλ.



Σχήμα 27:PBX params

Επίσης, στο tab Dialpan μπορούμε να προσθέσουμε users, να τροποποιήσουμε το Routing table, να σετάρουμε το Voicemail κτλ.



Σχήμα 28: Dilplan

4. Η κονσόλα του PBX διαθέτει ένα πλήθος εντολών. Αν πληκτρολογήσουμε help στο tab του CLI θα εμφανιστεί ένα πλήθος από αυτές όπως χαρακτηριστικά φαίνετε στο παρακάτω σχήμα:

   CLI>help

```
          sip no debug  Disable SIP debugging
        sip no history  Disable SIP history
           sip notify  Send a notify packet to a SIP peer
    sip prune realtime  Prune cached Realtime object(s)
sip prune realtime peer  Prune cached Realtime peer(s)
sip prune realtime user  Prune cached Realtime user(s)
           sip reload  Reload SIP configuration
      sip show channels  Show active SIP channels
      sip show channel  Show detailed SIP channel info
      sip show domains  List our local SIP domains.
      sip show history  Show SIP dialog history
        sip show inuse  List all inuse/limits
      sip show objects  Show all SIP object allocations
         sip show peer  Show details on specific SIP peer
        sip show peers  Show defined SIP peers
     sip show registry  Show SIP registration status
      sip show settings  Show SIP global settings
sip show subscriptions  Show active SIP subscriptions
        sip show users  Show defined SIP users
         sip show user  Show details on specific SIP user
          skinny debug  Enable Skinny debugging
       skinny no debug  Disable Skinny debugging
   skinny show devices  Show defined Skinny devices
     skinny show lines  Show defined Skinny lines per device
           soft hangup  Request a hangup on a given channel
       stop gracefully  Gracefully shut down Asterisk
             stop now  Shut down Asterisk immediately
  stop when convenient  Shut down Asterisk at empty call volume
           tapi debug  Enable TAPI debugging
            tapi info  Show TAPI info
        tapi no debug  Disable TAPI debugging
              unload  Unload a dynamic module by name
```

Σχήμα 29: Asterisk useful commands

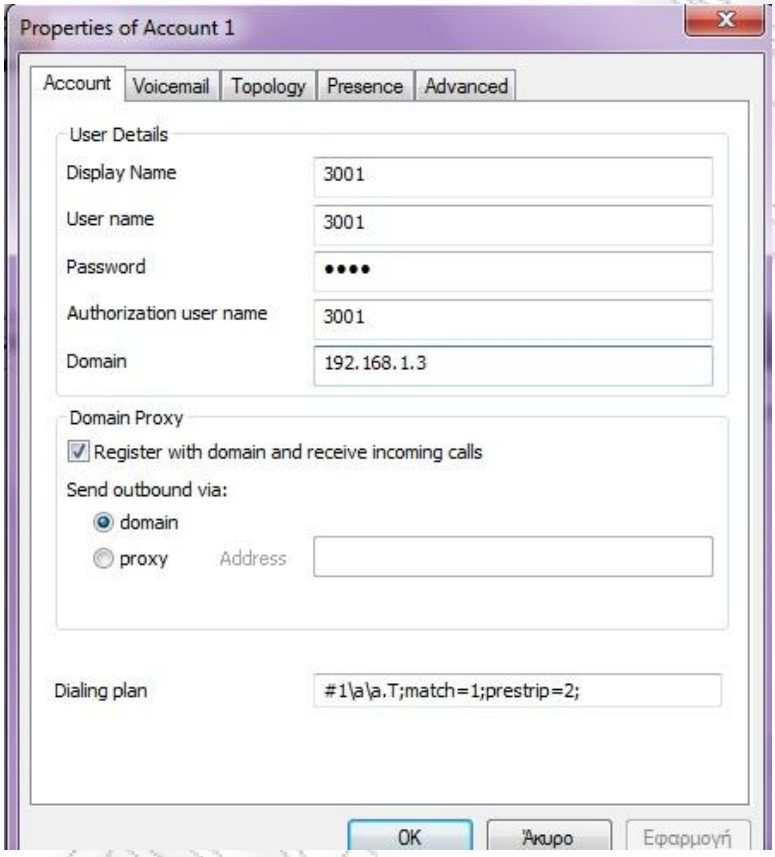
- **Βήματα σεταρίσματος του x-lite:**

  Για να σετάρουμε ένα X-lite πρέπει να επιλέξουμε από το μενού SIP account settings και να ορίσουμε τις παρακάτω παραμέτρους:

  Στο Account section ορίζουμε τα παρακάτω:

  - Display name: το username του account έτσι όπως έχει οριστεί στη βάση του Asterisk
  - User name : το username του account
  - Password: το password του συγκεκριμένου account έτσι όπως έχει οριστεί στον Astersisk, εφόσον μαζί με το username αποτελούν τα credentials για το registration του user
  - Domain: σαν domain θα ορίσουμε την ip του Asterisk στην οποία ο user θα στείλει τα request του.

- Στο tab Domain Proxy επιλέγουμε να στέλνουμε όλο το outbound traffic μέσω του domain που έχουμε ορίσει ήδη.
- Όλες οι άλλες παράμετροι παραμένουν ως έχουν.



Σχήμα 30:X-lite configuration

Όταν τελειώσουμε με το σετάρισμα του SIP account και αν έχουμε χρησιμοποιήσει τα σωστά credentials ο συνδρομητής θα γίνει register στον asterisk και θα είμαστε πλέον έτοιμοι να χρησιμοποιήσουμε το softphone μας.

Σχήμα 31: X-lite registered to Asterisk

- Αφού έχουμε φτιάξει τους χρήστες στον asterisk και τους έχουμε σετάρει στα softphones αν τρέξουμε την εντολή sip show peers θα δούμε ποιοι χρήστες είναι register στον asterisk:

```
CLI> sip show peers
Name/username              Host            Dyn Nat ACL Port     Status
3003/3003                  (Unspecified)   D           0        Unmonitored
3002/3002                  (Unspecified)   D           0        Unmonitored
3001/3001                  192.168.1.2     D           44096    Unmonitored
3000/3000                  192.168.1.3     D           28120    Unmonitored
4 sip peers [4 online , 0 offline]
```

Επίσης με την εντολή show datebase στην κονσόλα του asterisk μπορούμε να δούμε τα Register peers. Η πληροφορία που παίρνουμε από αυτή την εντολή είναι τα sip bindings μετά τα registration των accounts:

```
CLI> database show
/SIP/Registry/3000                    : 192.168.1.3:7084:3600:3000:sip:3000@192.168.1.3:7084;rinstance=785c74f4fe3f1c28
/SIP/Registry/3001                    : 192.168.1.2:64536:3600:3001:sip:3001@192.168.1.2:64536;rinstance=087dcd5e9f1132e3
/dundi/secret                         : oVX2LBySemepvkUvkpJ9Ww
/dundi/secretexpiry                   : 1314037561
```

4.4 Εργαστηριακή Άσκηση

4.4.1 Γενικά χαρακτηριστικά του πρωτοκόλλου SIP

Δώστε σύντομες απαντήσεις στα παρακάτω ερωτήματα σχετικά με το πρωτόκολλο SIP:

1. Απαριθμήστε πέντε διαφορετικές παραμέτρους απαραίτητες για την εγκατάσταση και το τερματισμό μιας SIP to SIP κλήσης.
2. Περιγράψτε τα βασικά στρώματα της δομής του SIP πρερωτοκόλλου.
3. Ποιά είναι τα βασικά δομικά στοιχεία του SIP. Δώστε μια σύντομη περιγραφή για καθένα από αυτά.

4.4.2 Call Proccessing

4.4.2.1 Registration Proccess

Η διαδικασία του registration χρησιμοποιείτε από έναν UA για να δεσμεύσει την τρέχουσα θέση του χρήστη και να διευκόλυνει τη δρομολόγηση των μηνυμάτων εντός του SIP δικτύου. Αυτό πραγματοποιείτε από έναν UA με την αποστολή ενός register μηνύματος σε έναν registrar server. Ο registrar διατηρεί το binding του συγκεκριμένου user για ένα συγκεκριμένο χρονικό διάστημα το οποίο καθορίζεται από τον Expires header.

Με βάση τα παραπάνω σχόλια θα προχωρήσουμε στην εκτέλεση της άσκησης. Για να ξεκινήσουμε, ανοίγουμε ένα packet sniffering tool (π.χ wireshark) και ξεκινάμε το x-lite που έχουμε εγκαταστήσει και σετάρει στον προσωπικό μας υπολογιστή. Μετά την ολοκλήρωση του registration τερματίστε την εφαρμογή.

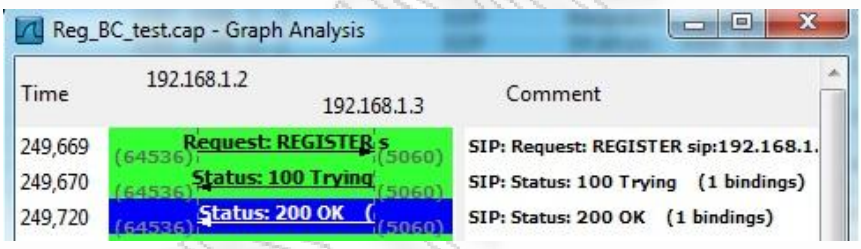
Απαντήστε στα παρακάτω ερωτήματα:

1. Ποιο display filter πρέπει να χρησιμοποιηθεί για την εμφάνιση μόνο των SIP μηνυμάτων κατά το registration του account σας;
2. Ποια είναι η IP address και το port του SIP Registrar που κατευθύνονται τα πακέτα;
3. Ποιο πρωτόκολλο μεταφοράς και ποια έκδοση του SIP πρωτοκόλλου χρησιμοποιείται;
4. Από τα ληφθέντα SIP μηνύματα διαχωρίστε τα SIP responses από τα SIP requests κατά τη διάρκεια του registration process.
5. Ποίο είναι το SIP binding του UA και ποια τα βασικά χαρακτηριστικά (πχ. όνομα,διεύθυνση).
6. Ορίστε τους κυριότερους headers του μηνύματος.

7. Ποια είναι η τιμή του πεδίου επικεφαλίδας C-Seq και ποιός ο ρόλος του.
8. Περιγράψετε την αλληλουχία των μηνυμάτων που παρατηρείτε κατά τη διάρκεια του registration και ποια η σημασία καθενός από αυτά.
9. Ποιος αλγόριθμος κρυπτογράφησης δεδομένων υποστηρίζεται από το διακομιστή και σε ποιο μήνυμα δηλώνεται.
10. Τι θα συμβεί σε περίπτωση που το registration του συγκεκριμένου χρήστη αποτύχει; Μπορούν άλλοι χρήστες να πραγματοποιήσουμ κλήσεις προς αυτόν το συνδρομητη;

Τα διαγράμματα που ακολουθούν είναι αποτέλεσμα ενός Register request που στάλθηκε από ένα sip account στον Asterisk. Στη συγκεκριμένη περίπτωση δε έχουμε χρησιμοποιήσει digest authentication. Στην περίπτωση αυτή ο Asterisk θα έστελνε ένα challenge request στον UA με τη μορφή ένος 401 UnAuthorised μηνύματος.

Η ακολουθία των μηνυμάτων είναι η εξής:



Σχήμα 32: Ανταλλαγή μηνυμάτων κατά τη διαδικασία του registration

```
⊞ Frame 445 (574 bytes on wire, 574 bytes captured)
⊞ Ethernet II, Src: HonHaiPr_35:b8:b5 (90:4c:e5:35:b8:b5), Dst: IntelCor_bb:b1:20 (00:1b:77:bb:b1:20)
⊞ Internet Protocol, Src: 192.168.1.2 (192.168.1.2), Dst: 192.168.1.3 (192.168.1.3)
⊞ User Datagram Protocol, Src Port: 64536 (64536), Dst Port: sip (5060)
⊟ Session Initiation Protocol
  ⊟ Request-Line: REGISTER sip:192.168.1.3 SIP/2.0
      Method: REGISTER
    ⊞ Request-URI: sip:192.168.1.3
      [Resent Packet: False]
  ⊟ Message Header
    ⊞ Via: SIP/2.0/UDP 192.168.1.2:64536;branch=z9hG4bK-d8754z-fe7eec4328562b00-1---d8754z-;rport
      Max-Forwards: 70
    ⊞ Contact: <sip:3001@192.168.1.2:64536;rinstance=087dcd5e9f1132e3>
    ⊞ To: "3001"<sip:3001@192.168.1.3>
    ⊞ From: "3001"<sip:3001@192.168.1.3>;tag=af107309
      Call-ID: NTA5MWI3ZjA2MmY4M2YxZWQxNmE4YzIzNWM0ZjIwNmU.
    ⊞ CSeq: 1 REGISTER
      Expires: 3600
      Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO
      User-Agent: X-Lite release 1104o stamp 56125
      Content-Length: 0
```

Σχήμα 33: Ένα Register request όπως φαίνετε αναλυτικά με τη βοήθεια του sniffer.

4.4.2.2 Call Proccessing

Σε αυτή την ενότητα θα μελετήσουμε τα βασικά μηνύματα για την πραγματοποιήση μιας peer to peer κλήσης. Το πιο σημαντικό μήνυμα το οποίο οριοθετεί και την έναρξη ενός νέου transaction είναι ένα INVITE request.

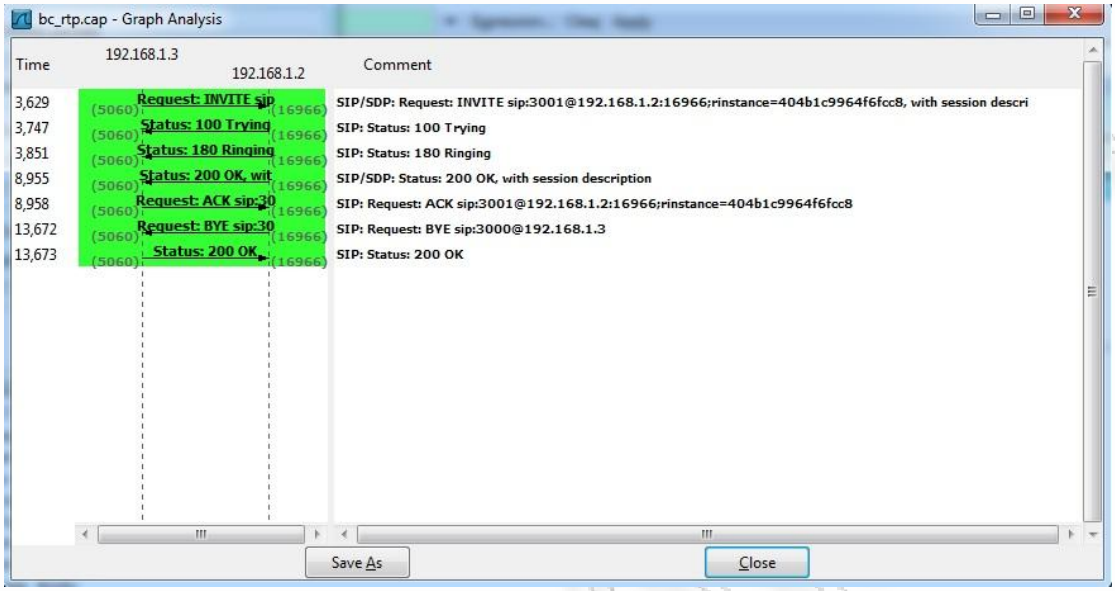
Για να ξεκινήσετε την άσκηση ανοίξτε ένα νέο παράθυρο στο sniffer που τρέχει στον προσωπκό υπολογιστή σας. Στη συνέχεια ξεκινήστε μια κλήση από το x-lite σας προς κάποιον άλλο user. Μετά την ολοκλήρωση της συνομιλίας τερματίστε την κλήση σας και σταματήστε τον sniffer.

Απαντήστε στα παρακάτω ερωτήματα:

1. Ποιο display filter πρέπει να χρησιμοποιηθεί για την εμφάνιση μόνο των SIP μηνυμάτων που ανταλλάσσονται κατά τη διάρκεια της κλήσης;
2. Ποίο πρωτόκολλο μεταφοράς χρησιμοποιήθηκε σε αυτή την κλήση;
3. Ποια είναι η αλληλουχία των μηνυμάτων και ποία από αυτά είναι requests και ποια responses.
4. Αναλύστε τους headers σημαντικότερων μηνυμάτων.
5. Διαχωρίστε τα transactions που συμμετέχουν στο συγκεκριμένο Session.
6. Με βάση ποιές παραμέτρους αντιστοιχίζετε ένα response σε ένα transaction.
7. Ποιοί headers καθορίζουν ένα transaction;
8. Ποιός ο ρόλος του CSeq header;
9. Τι πληροφορίες παίρνουμε από το message body ;
10. Το SDP ποιών πακέτων χρησιμοποιείτε για το SDP negotiaton;
11. Με βάση τα sniffered πακέτα αναλύστε το μοντέλο offer/aswer.

Παρακάτω ακολοθούν τα graph flows από διάφορα basic calls στα οποία θα δούμε τη συμπεριφορά του SIP και την ανταλλαγή μηνυμάτων στην εκάστωτε περίπτωση.

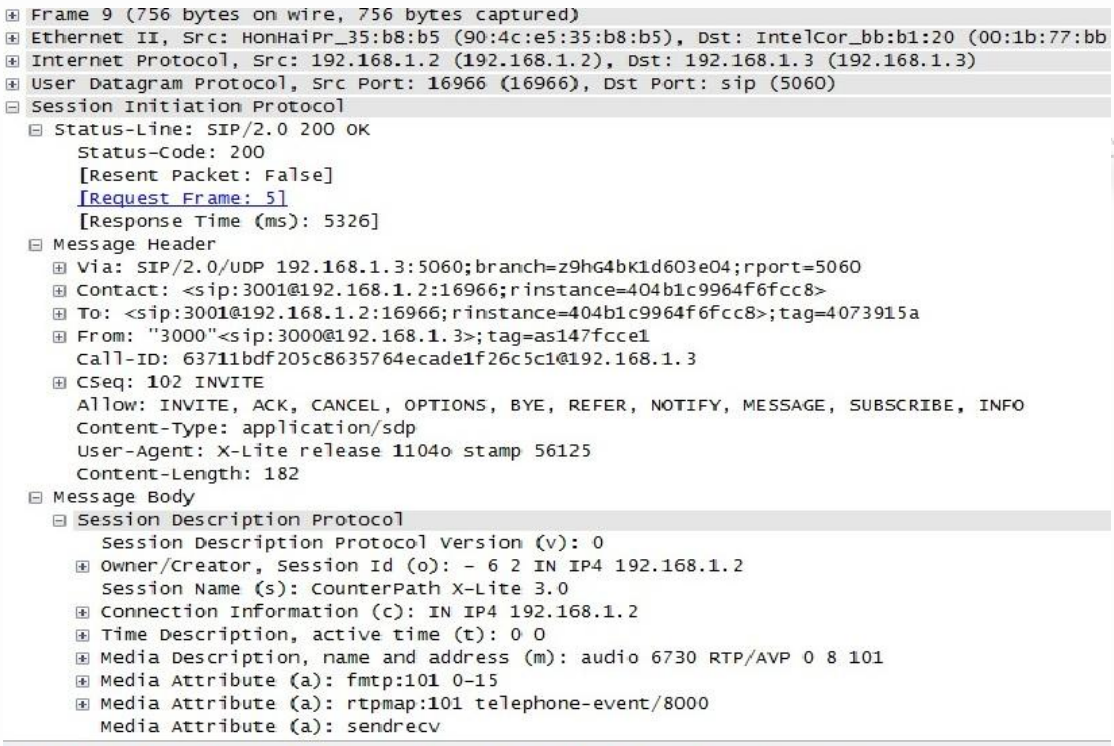
1. Scenario 1: SIP to SIP basic call



Σχήμα 34: Η ανταλλαγή μηνυμάτων σε ένα basic call.

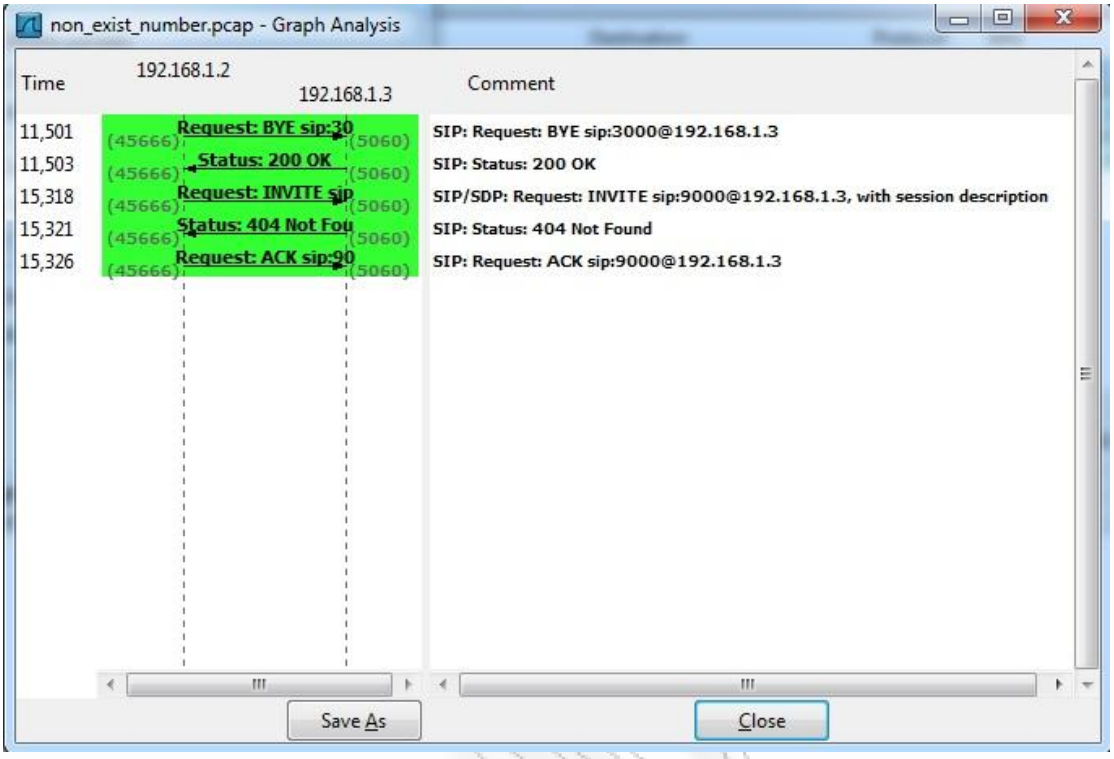
```
⊞ Internet Protocol, Src: 192.168.1.3 (192.168.1.3), Dst: 192.168.1.2 (192.168.1.2)
⊞ User Datagram Protocol, Src Port: sip (5060), Dst Port: 16966 (16966)
⊟ Session Initiation Protocol
  ⊟ Request-Line: INVITE sip:3001@192.168.1.2:16966;rinstance=404b1c9964f6fcc8 SIP/2.0
      Method: INVITE
    ⊞ Request-URI: sip:3001@192.168.1.2:16966;rinstance=404b1c9964f6fcc8
      [Resent Packet: False]
  ⊟ Message Header
    ⊞ Via: SIP/2.0/UDP 192.168.1.3:5060;branch=z9hG4bK1d603e04;rport
    ⊞ From: "3000" <sip:3000@192.168.1.3>;tag=as147fcce1
    ⊞ To: <sip:3001@192.168.1.2:16966;rinstance=404b1c9964f6fcc8>
    ⊞ Contact: <sip:3000@192.168.1.3>
      Call-ID: 63711bdf205c8635764ecade1f26c5c1@192.168.1.3
    ⊞ CSeq: 102 INVITE
      User-Agent: Asterisk PBX
      Max-Forwards: 70
      Date: Wed, 24 Aug 2011 18:38:20 GMT
      Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY
      Content-Type: application/sdp
      Content-Length: 259
  ⊟ Message Body
    ⊟ Session Description Protocol
        Session Description Protocol Version (v): 0
      ⊞ Owner/Creator, Session Id (o): root 4072 4072 IN IP4 192.168.1.3
        Session Name (s): session
      ⊞ Connection Information (c): IN IP4 192.168.1.3
      ⊞ Time Description, active time (t): 0 0
      ⊞ Media Description, name and address (m): audio 18352 RTP/AVP 0 3 8 101
      ⊞ Media Attribute (a): rtpmap:0 PCMU/8000
      ⊞ Media Attribute (a): rtpmap:3 GSM/8000
      ⊞ Media Attribute (a): rtpmap:8 PCMA/8000
      ⊞ Media Attribute (a): rtpmap:101 telephone-event/8000
```

Σχήμα 34: Το πρώτο INVITE request με SDP body.



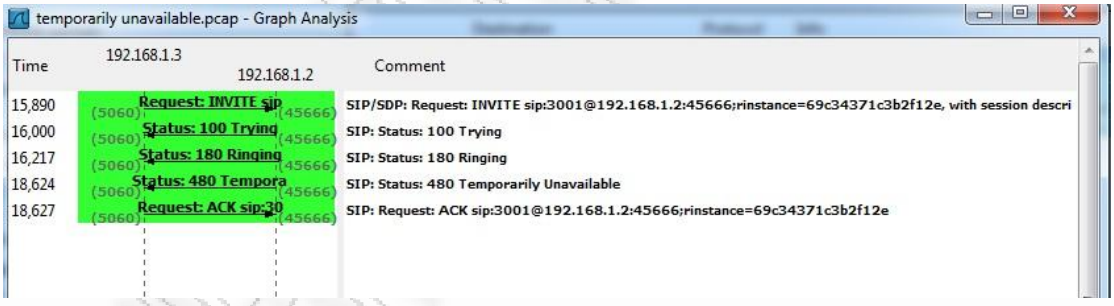
Σχήμα 35: Το 200 OK με SDP body που στέλνει το second leg για να ξεκινήσει το SDP negotiaton

2. **Scenario 2:** Ο user A καλεί έναν αριθμό που δεν υπάρχει στη βάση του Asterisk και ο Asterisk του απαντάει με 404 Not Found.



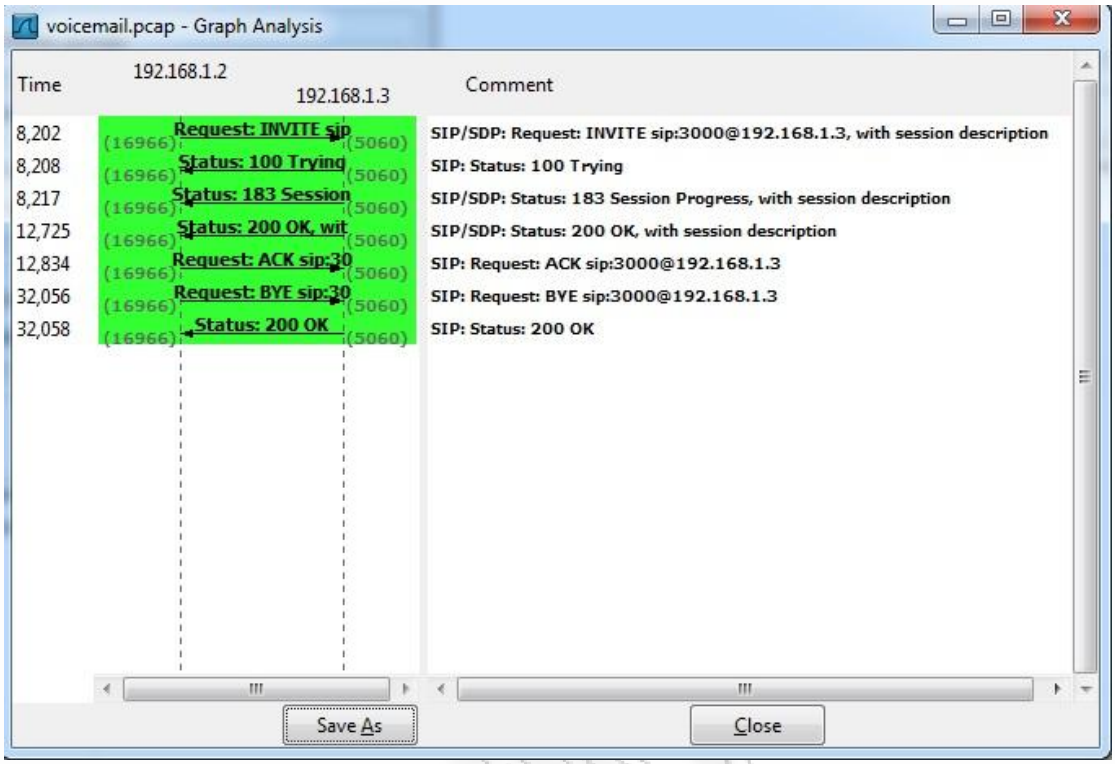
Σχήμα 36:Ο Asterisk απαντάει με 404 Not Found στο request του user agent.

3. **Scenario 3:** Ο user A καλεί το User B ο οποίος απορρίπτει την εισερχόμενη κλήση. Ένα 480 Temporarily unavailable θα σταλεί στον user A.



Σχήμα 37: Ο user agent λαμβάνει ένα 480 Temporarily Unavailable.

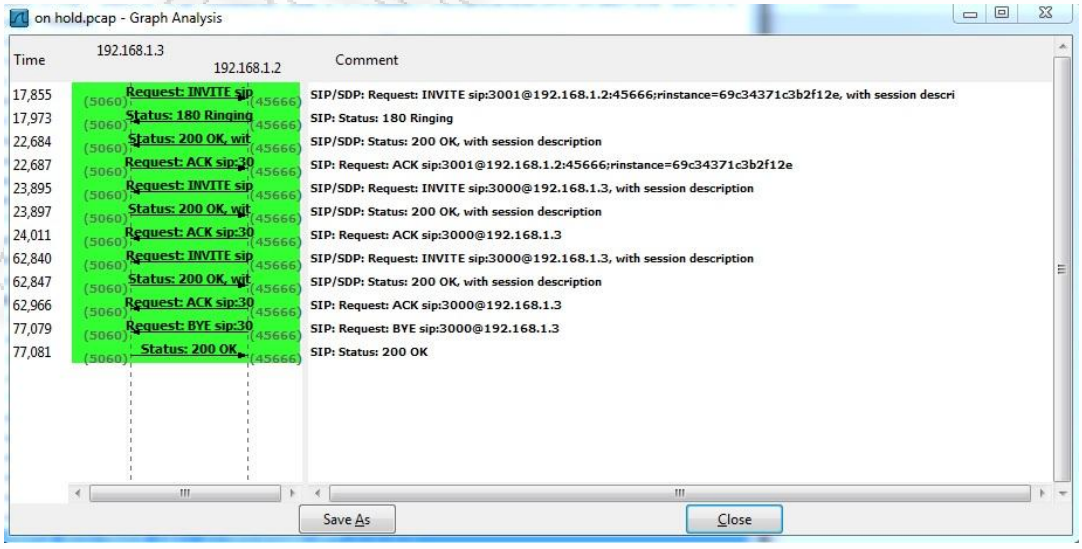
4. **Scenario 4:** Ο user A καλεί τον user B ο οποίος δε απαντά και έχει ενεργοποιήμένο το Voicemail service του. Ο user A θα λάβει ένα 183 Session in Progress μήνυμα και θα ακούσει ένα announcement που θα τον κατευθύνει να αφήσει το μήνυμα του.



Σχήμα 38: Voicemail Call Flow

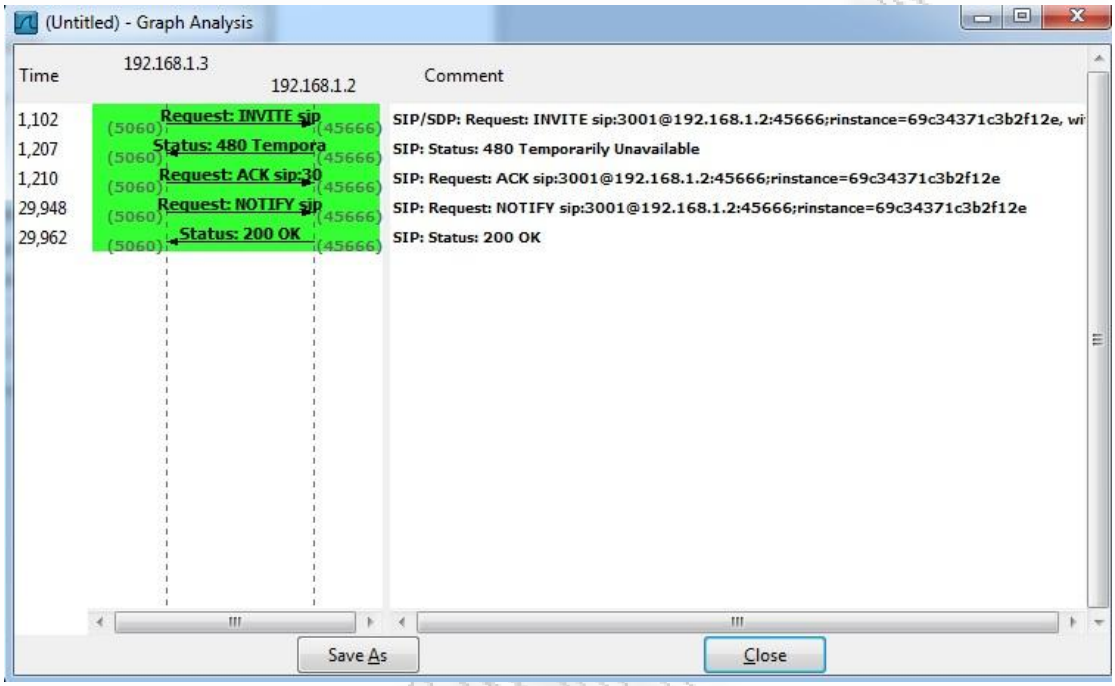
5. **Scenario 5:** Ο user A καλεί το user B και ξεκινά το conference. Ο user B αποφασίζει να βάλει σε hold τον Α. Στη συνέχεια αφου γίνει ξανά διαθέσιμος το session θα συνεχιστεί και θα τερματιστεί κανονικά.

   Αυτό που πρέπει να σημειωθεί στο συγκεκριμένο σενάριο είναι ότι η πρώτη INVITE που στέλνει ο user A έχει SDP body το οποίο έχει τη παράμετρο a=sendreceive, ενώ η δεύτερη INVITE η οποία θα βάλει ουσιαστικά το second leg on hold θα έχει attribute a=inactive.



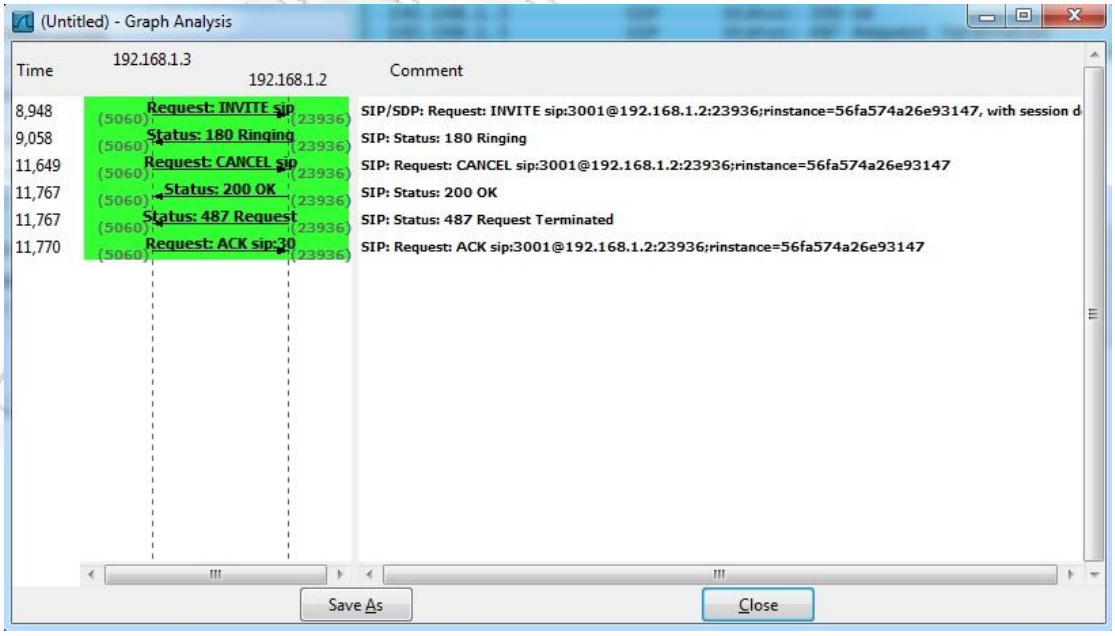
Σχήμα 39: On hold call flow

6. **Scenario 6:** Ο user A καλεί το User B ο οποίος έχει το Do Not Disturb feature ενεργοποιημένο. Και σε αυτή τη περίπτωση ο συνδρομητής θα λάβει ένα 480 Temporarily Unavailable.



Σχήμα 40: Do not Disturb call flow.

7. **Scenario 7:** Ο user **A** καλεί το User B, ενώ τελικά αποφασίζει ότι δε θέλει να ολοκληρώσει την κλήση και κλείνει το τηλέφωνο, οπότε ένα CANCEL μήνυμα σταματά την περαίωση της κλήσης.



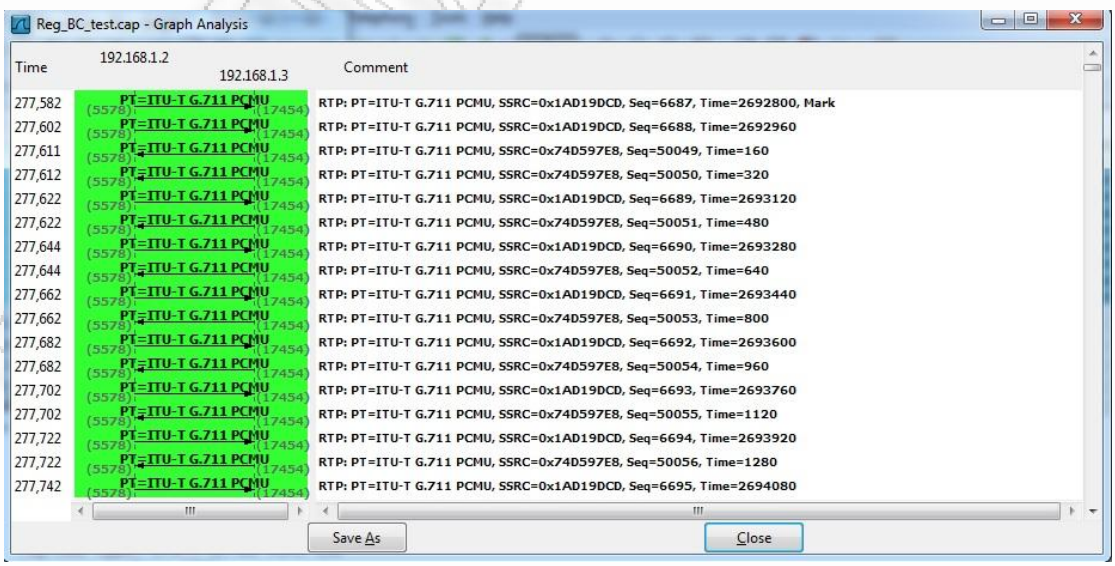
Σχήμα 41: CANCEL ένος session.

4.4.2.3 RTP Stream

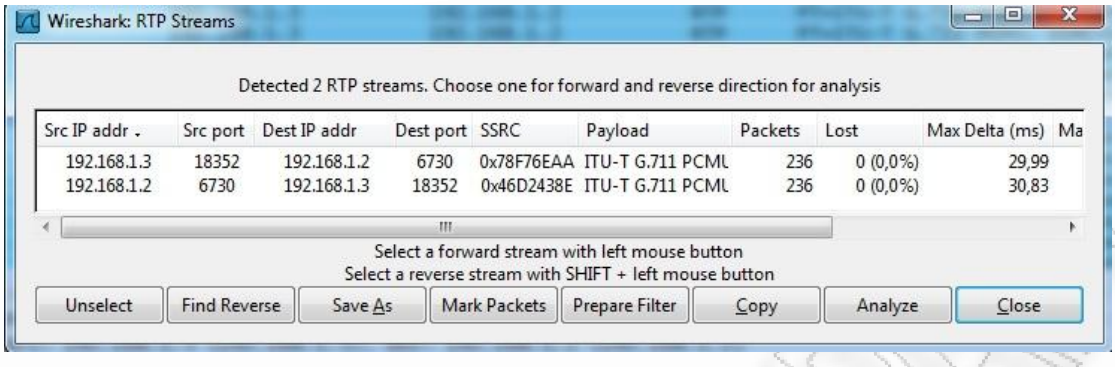
Το πρωτόκολλο RTP χρησιμοποιείτε για την μετάδοση πολυμεσικών ροών μεταξύ δύο τερματικών. Στο σημείο αυτό της άσκησης θα μελετήσουμε ένα end to end RTP stream. Όπως και παραπάνω ανοίγουμε ένα sniffer και κάνουμε ένα basic call.

Απαντήστε στα παρακάτω ερωτήματα:

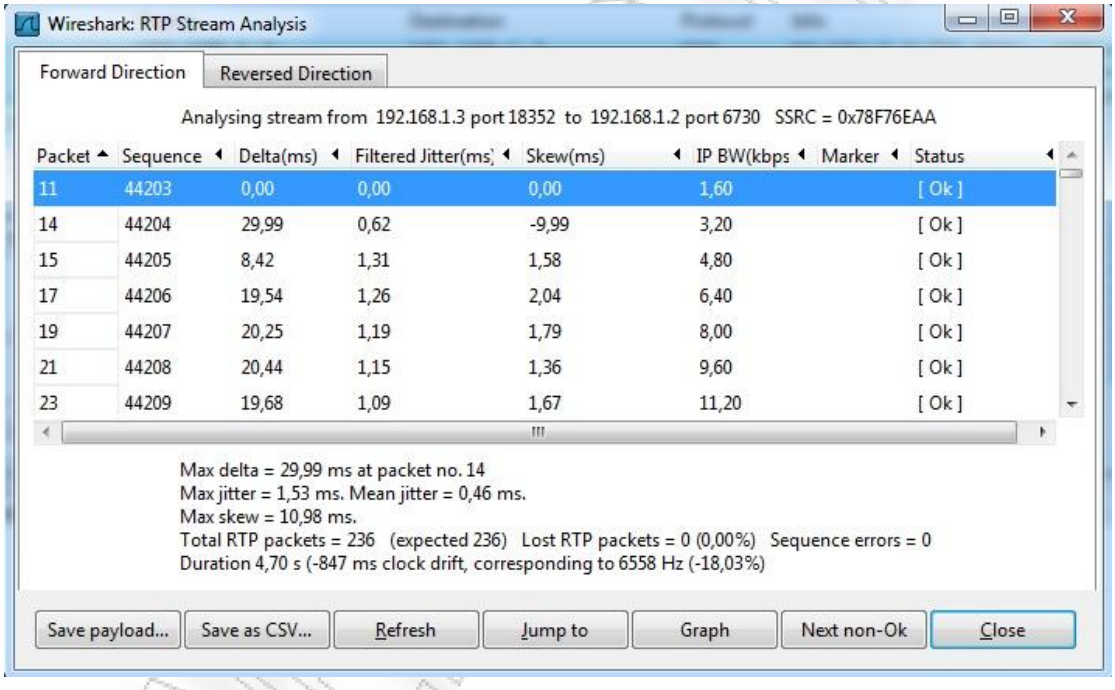
1. Ποιο φίλτρο θα χρησιμοποιήσετε για να συλλέξετε τα RTP πακέτα;
2. Ποιο πρωτόκολλο είναι υπεύθυνο για τον καθορισμό των ports και των IP addresses που χρσησιμοποιούνται και τη διάρκεια της ανταλλαγής RTP πακέτων;
3. Ποία έκδοση του πρωτοκόλλου χρησιμοποιείτε;
4. Ποίος τύπος codec έχει χρησιμοποιηθεί στη συγκεκριμένη κλήση; Δώστε παραδείγματα άλλων codec που θα μπορούσαμε να χρησιμοποιήσουμε.
5. Ποίο είναι το μέγεθος του φορτίου του RTP πακέτου;
6. Ποια παράμετρος διαχωρίζει ποια RTP πακέτα προέρχονται από τον πομπό και ποια από τον δέκτη;
7. Τι είναι ο sequence number και ποια η χρησιμότητά του; Γιατί η τιμή του αυξάνετε κατά 1;
8. Με τη βοήθεια του wireshark δώστε μερικά στατιστικά για τη συγκεμριμένη κλήση.



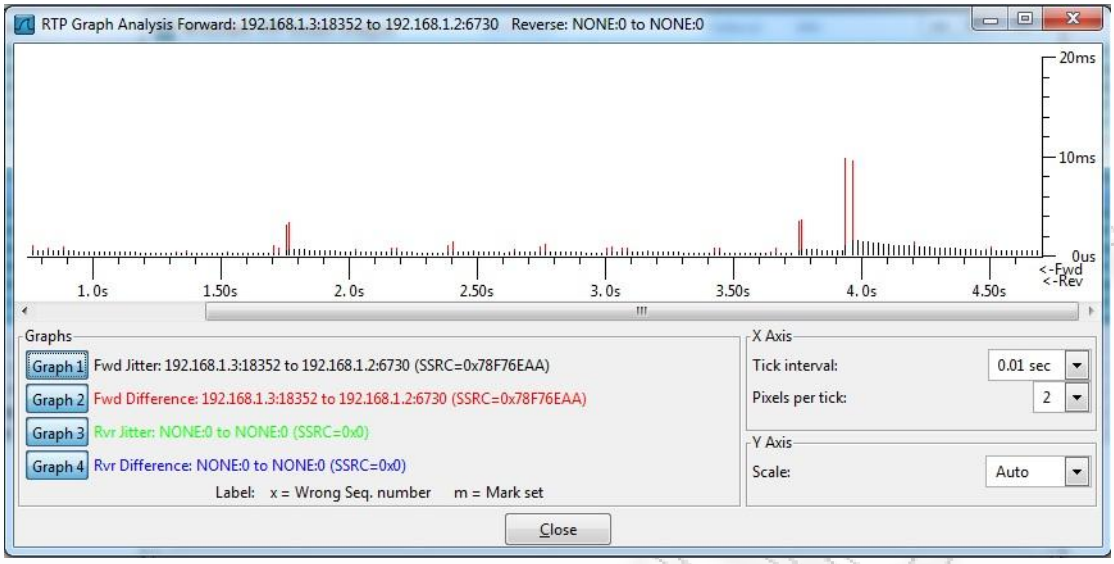
Σχήμα 42: Ανταλλαγή RTP πακέτων.



Σχήμα 43: Active RTP streams. Για τον εντοπισμό κάθε ροής ήχου χρησιμοποείται το πεδίο SSRC που αποτελεί το αναγνωριστικό χαρακτηριστικό της πηγής κάθε ροής.



Σχήμα 44: Stream analysis



Σχήμα 44: Graph analysis of RTP streaming με τη χρήση του Wireshark.

.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Joe Hallock, Evolution and Trends in Digital Media Technologies – COM 538 /Masters of Communication in Digital Media/University of Washington
2. Institute for Computer Communications and Applicatins -ICA (EPFL),Voice Service Interworking for PSTN and IP Networks
3. Cisco, "Session Initiation Protocol Gateway Call Flows and Compliance Information", Cisco Publ.
4. Javvin , "Network Protocols Handbook 2nd Edition", Javvin Technologies Inc
5. Mark Spencer, Mack Allison, Christopher Rhodes, "The Asterisk Handbook Version 2", Digium Inc.
6. Tom Jenkins, A guide to Session Initiation Protocol (SIP)
7. Sip server technical overview
8. Overview of the PSTN and Comparisons to Voice over IP
9. Matthew Hurley, VoIP VULNERABILITIES
10. Lay G. Ding and Lin Liu, School of Computer and Information Science University of South Australia, Modelling and Analysis of the INVITE Transaction of the Session Initiation Protocol Using Coloured Petri Nets
11. RFC 3435 - Media Gateway Control Protocol (MGCP) Version 1.0
12. RFC 1889 - RTP: A Transport Protocol for Real-Time Applications
13. RFC 3261, SIP: Session Initiation Protocol
14. ITU-T, "T-REC E.164", International Telecommunication Union
15. ITU-T, An Offer/Answer Model with the Session Description Protocol (SDP)
16. ITU - Telecommunication Standardization Sector, GLOBAL IMPLEMENTATION OF ENUM: A TUTORIAL PAPER
17. ITU - Telecommunication Standardization Sector, ENUM Administration ad interim