Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακό Πρόγραμμα Σπουδών *στα Πληροφοριακά και Επικοινωνιακά Συστήματα*

Μεταπτυχιακή Διατριβή

Κρυπτογράφηση σε VoIP τεχνολογία: Μελέτη περίπτωσης

Δημήτριος Αλβανός

Επιβλέπων Καθηγητής
Δρ. Κωνσταντίνος Λιμνιώτης

Δεκέμβριος 2016

Ανοικτό Πανεπιστήμιο Κύπρου

Σχολή Θετικών και Εφαρμοσμένων Επιστημών

Μεταπτυχιακό Πρόγραμμα Σπουδών *στα Πληροφοριακά και Επικοινωνιακά Συστήματα*

Μεταπτυχιακή Διατριβή

Κρυπτογράφηση σε VoIP τεχνολογία: Μελέτη περίπτωσης

Δημήτριος Αλβανός

Επιβλέπων Καθηγητής
Δρ. Κωνσταντίνος Λιμνιώτης

Η παρούσα μεταπτυχιακή διατριβή υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων για απόκτηση μεταπτυχιακού τίτλου σπουδών στα Πληροφοριακά Συστήματα με κατεύθυνση Ασφάλεια Συστημάτων από τη Σχολή Θετικών και Εφαρμοσμένων Επιστημών του Ανοικτού Πανεπιστημίου Κύπρου.

Δεκέμβριος 2016

ΛΕΥΚΗ ΣΕΛΙΔΑ

Περίληψη

Η ασφάλεια στις –ασύρματες ή μη- επικοινωνίες αποτελεί πρωταρχικό σχεδιαστικό στόχο των επικοινωνιακών συστημάτων και εφαρμογών. Ιδιαίτερα η χρήση της τεχνολογίας VoIP, λόγω της ολοένα αυξανόμενης διείσδυσής της σε καθημερινές εφαρμογές, θέτει εκ των πραγμάτων πολλά ερωτήματα ως προς την παρεχόμενη ασφάλεια ιδίως δε αν λάβει κανείς υπόψη ότι δεδομένα που πρέπει να προστατευθούν ενδέχεται να αποθηκεύονται σε φορητές συσκευές (tablets κ.λπ.).

Αντικείμενο της παρούσας διατριβής είναι η μελέτη της ασφάλειας που παρέχεται από VoIP εφαρμογές, με μελέτη περίπτωσης την εφαρμογή Linphone η οποία αφενός παρουσιάζει ολοένα αυξανόμενη χρήση και αφετέρου πρόκειται για εφαρμογή ανοικτού κώδικα, κάτι που επιτρέπει την ενδελεχέστερη ανάλυσή της. Βάσει αυτών, απώτερος στόχος της διατριβής είναι η πλήρης κατανόηση και αποτίμηση της παρεχόμενης ασφάλειας της εφαρμογής LinPhone, η οποία δεν έχει μελετηθεί εκτενώς στη βιβλιογραφία, καθώς επίσης και η βελτίωση της.

Ειδικότερα, στο πλαίσιο εκπόνησης της παρούσας διατριβής γίνεται αρχικά μία γενική μελέτη στα θέματα ασφάλειας και στους σχετικούς κινδύνους που ανακύπτουν σε εφαρμογές VoIP. Στη συνέχεια, η έρευνα εστιάζει ειδικά στην εφαρμογή Linphone, όπου πραγματοποιείται μελέτη των υποκείμενων λειτουργιών και πρωτοκόλλων της εφαρμογής, με έμφαση στα υποστηριζόμενα πρωτόκολλα ασφαλείας (TLS, ZRTP, SRTP), καθώς επίσης και μελέτη των γνωστών επιθέσεων που έχουν καταγραφεί για τα πρωτόκολλα αυτά.

Στο πλαίσιο αυτό, περιγράφουμε μία νέα διαδικασία που μπορεί να είναι αποτελεσματική για την αντιμετώπιση συγκεκριμένου τύπου επιθέσεων τύπου «άνθρωπος-στο-μέσο» (man-in-the-middle) ως προς το πρωτόκολλο ZRTP οι οποίες έχουν καταγραφεί στη βιβλιογραφία. Περαιτέρω, μέσα από πρακτική εφαρμογή σε εργαστηριακό περιβάλλον διαπιστώθηκαν συγκεκριμένα κενά ασφάλειας στο Linphone, τα οποία σχετίζονται ιδίως με το γεγονός ότι το TLS δεν χρησιμοποιείται εξ ορισμού στην εν λόγω εφαρμογή για τα δεδομένα σηματοδοσίας, καθώς επίσης και το ότι αρκεί το ένα από τα δύο μέλη της επικοινωνίας να μην χρησιμοποιεί TLS – ακόμα κι αν το άλλο μέλος χρησιμοποιεί - προκειμένου να πραγματοποιηθεί μη κρυπτογραφημένη (και, άρα, μη ασφαλής) ανταλλαγή μηνυμάτων σηματοδοσίας. Περαιτέρω, αντιμετωπίσαμε μία πραγματική επίθεση «ανεπιθύμητων κλήσεων» η οποία είχε σκοπό να ελέγξει (scan) το σύστημά μας. Για όλα τα ανωτέρω ζητήματα παρατίθενται προτάσεις αντιμετώπισής τους. Τέλος, ελέγχθηκε η απόδοση του Linphone εντός ενός VPN, όπου διαπιστώθηκε κατ' αρχάς ότι δεν παρατηρείται κάποια αξιοσημείωτη μείωση αυτής – γεγονός που συνηγορεί στο ότι δύνανται πρακτικά να αξιοποιηθούν τεχνολογίες VPN για πρόσθετη ασφάλεια στο Linphone.

Summary

Security in - wireless or wired-communications is a primary design goal of communication systems and applications. Particularly the VoIP applications, due to their increased use, pose many questions with regard to the provided security - especially if we consider that data which should be protected may be stored on portable devices (tablets etc.).

The subject of this thesis is the study of security provided by VoIP applications, whereas the Linphone application is being used as a case study since it is open source allowing a more detailed analysis and it has not been studied to a large extent in the literature.

On this basis, the ultimate aim of the thesis is the full understanding and evaluation of the Linphone application, focusing on possible improvements on its security features.

More precisely, this thesis first presents an extended study on known security issues and related risks occurring in VoIP applications. Subsequently, the research focuses on the specific application Linphone, studying the underlying functions and implementation protocols, with emphasis on supported security protocols (TLS, ZRTP, SRTP), as well as on known attacks that have been recorded for these protocols.In this context, we describe a new procedure that can be effective for treating specific-type "man-in-middle" attacks to the ZRTP protocol that are known in the literature.

In addition, through the practice in a laboratory, the Linphone has been extensively analyzed, revealing several security issues, which mainly rest with the fact the TLS is not used by default for data signaling. In particular, we observe that if the initiator of a call does not support TLS, then the call will be established even if the other party is being configured to use TLS. Moreover, we faced a real attack of “unwanted calls” (ghost calls) which attempted to scan our system. For all the above issues, we discuss approaches to address them. Finally, the performance of Linphone within a VPN has been studied, exhibiting that there is no significant reduction– thus concluding that VPN can be used to enhance security in Linphone without sacrificing the performance.

.

Ευχαριστίες

Πρέπει να δοθούν ιδιαίτερες ευχαριστίες στον επιβλέπων καθηγητή μου Κο Κωνσταντίνο Λιμνιώτη για την υπομονή που επέδειξε απέναντί μου καθώς και για την ταχύτητα ανταπόκρισής του με διορθώσεις και υποδείξεις βελτίωσης της εργασίας.

Στην οικογένεια και ειδικά στα τρία παιδιά μου που υπέμεναν την κατάληψη που έκανα στον υπολογιστή του σπιτιού καθώς και στους συναδέλφους μου στο σχολείο οι οποίοι ανέλαβαν για ένα χρονικό διάστημα τις εξωδιδακτικές εργασίες μου ώστε να έχω λίγο παραπάνω χρόνο για έρευνα.

Περιεχόμενα

Κεφάλαιο 1
Εισαγωγή

Ένα από τα σημαντικότερα επιτεύγματα του προηγούμενου αιώνα ήταν η ανάπτυξη της τηλεφωνίας, η οποία ξεκίνησε από τον Alexander Graham Bell τη δεκαετία του 1870.

Κατά τη διάρκεια της δεκαετίας αυτής, δύο εφευρέτες της εποχής, ο Elisha Gray και ο Alexander Graham Bell, ανεξάρτητα ο ένας από τον άλλο, σχεδιάζουν συσκευές που έχουν τη δυνατότητα να μεταδίδουν ανθρώπινη φωνή με ηλεκτρικά σήματα δημιουργώντας έτσι τα πρώτα τηλέφωνα της ιστορίας. Ο κάθε εφευρέτης σπεύδει να κατοχυρώσει τα πνευματικά δικαιώματα της ευρεσιτεχνίας του, με διαφορά ωρών ο ένας από τον άλλο με τον Bell να "κόβει πρώτος το νήμα" ξεκινώντας μία έντονη διαμάχη με τον Gray για την πατρότητα του τηλεφώνου. Τελικά, κερδισμένος βγήκε ο Bell, του οποίου το όνομα έγινε συνώνυμο του τηλεφώνου.

Όπως είναι φυσικό, η επικοινωνία κατά την "βρεφική" της ηλικία, ακόμη και για μικρές γεωγραφικές αποστάσεις, ήταν κακής έως μέτριας ποιότητας, ενώ η κατοχή τηλεφωνικής συσκευής ήταν προνόμιο ελάχιστων και πολύ εύπορων κοινωνικών τάξεων. Με την πάροδο του χρόνου, έγινε εμφανής η εξέλιξη τόσο των τηλεφωνικών συσκευών, όσο και των τηλεφωνικών κέντρων. Το τηλέφωνο άρχισε να ανάγεται σε απαραίτητη συσκευή κάθε σπιτιού, ενώ η ποιότητα της επικοινωνίας άρχισε να βελτιώνεται όλο και περισσότερο. Σχεδόν ταυτόχρονα όμως άρχισε να παρουσιάζεται παρόμοια εξέλιξη στον κόσμο των υπολογιστών. Ταυτόχρονα με την ευρεία διάδοση των υπολογιστών, εξίσου ευρεία ήταν και η διάδοση των δικτύων επικοινωνιών. Εμφανίστηκαν τα τοπικά δίκτυα σε στρατιωτικές εγκαταστάσεις, σε εκπαιδευτικά ιδρύματα και σε επιχειρήσεις με αποκορύφωμα το "δίκτυο των δικτύων", το Internet, το οποίο αν και ξεκίνησε για στρατιωτικούς σκοπούς, εντούτοις στο τέλος της δεκαετίας του 1980 άρχισε να διαδίδεται ευρέως, τόσο για διαφημιστικούς, όσο και για εκπαιδευτικούς και εμπορικούς σκοπούς. Ο καθένας, πλέον, μπορούσε από το σπίτι του

να διαβάσει ή να στείλει την αλληλογραφία του, να ενημερωθεί, να επικοινωνήσει και γενικά, να γίνει μέλος μιας παγκόσμιας κοινότητας ανταλλαγής πληροφοριών.

Γίνεται λοιπόν αντιληπτό ότι ο κόσμος των επικοινωνιών χαρακτηρίζεται από δύο είδη δικτύων: το τηλεφωνικό δίκτυο (αναλογικό) ή δίκτυο φωνής και το δίκτυο δεδομένων (ψηφιακό) με διαφορετικούς κανόνες και αρχές λειτουργίας αναμεσά τους, εξίσου όμως σημαντικά αμφότερα, απαραίτητα και αλληλοσυμπληρούμενα. Εδώ λοιπόν εισέρχεται η έννοια της τεχνολογίας του **VoIP** που είναι αρκτικόλεξο από τις λέξεις **V**oice **o**ver **I**nternet **P**rotocol, δηλαδή πρωτόκολλο φωνής πάνω από το διαδίκτυο. Το μέχρι πρότινος δίκτυο δεδομένων δεν μεταφέρει πλέον μόνο δεδομένα αλλά και φωνή, μετατρέποντας την ανθρώπινη αναλογική φωνή σε ψηφιακό σήμα που μεταδίδεται υπό μορφή πακέτων με τη χρήση των ίδιων πρωτοκόλλων που χρησιμοποιούνται στο Internet (IP protocols). Συχνά γίνεται χρήση του όρου IP telephony (IPT), εξαιτίας ακριβώς της χρήσης των πρωτοκόλλων αυτών. Το σημαντικότερο ίσως χαρακτηριστικό, χάρη στο οποίο οφείλεται η ευρεία διάδοση του VoIP, είναι το μηδενικό κόστος των κλήσεων εφόσον υπάρχει πρόσβαση στο Διαδίκτυο. Η τηλεφωνία IP εξαπλώνεται ραγδαία, με ολοένα και περισσότερες υπηρεσίες να προσανατολίζονται σ' αυτήν. Όλα δείχνουν ότι η τεχνολογία VoIP ικανοποιεί όλες τις προϋποθέσεις για να αποτελέσει το μελλοντικό τρόπο επικοινωνίας, ως εξέλιξη του υπάρχοντος τηλεφωνικού συστήματος. Σε αυτό συνηγορεί και η στροφή μεγάλων εταιρειών όπως η Samsung στα τηλεφωνικά κέντρα που υποστηρίζουν μόνο IP τηλεφωνία.

Προβάλει όμως και εδώ, όπως και σε οποιαδήποτε μορφή επικοινωνίας, ασύρματη ή μη, επιτακτική η ανάγκη για ασφάλεια της μεταδιδόμενης πληροφορίας αν αναλογιστούμε μάλιστα ότι δεδομένα μπορεί να αποθηκεύονται σε φορητές συσκευές όπως tablets ή smartphones και να μεταδίδονται μέσα από ανοιχτά χωρίς καμία ασφάλεια WiFi δίκτυα.

Εικόνα 1: Από τον Graham Bell και το πρώτο τηλέφωνο - σε συσκευή VoIP

1.1 Βασικά ερευνητικά ερωτήματα

Θα μελετήσουμε αρχικά θέματα ασφαλείας και τους κινδύνους που ελλοχεύουν κατά τη χρήση εφαρμογών VoIP. Κατόπιν θα εστιάσουμε στην εφαρμογή Linphone και θα την μελετήσουμε σε επίπεδο χρηστικότητας και παρεχόμενων λειτουργιών. Ακολούθως θα μελετήσουμε τις υποκείμενες λειτουργίες και πρωτόκολλα της εφαρμογής, με έμφαση στα υποστηριζόμενα πρωτόκολλα ασφαλείας (TLS, ZRTP, SRTP), και θα ερευνήσουμε τις γνωστές επιθέσεων που έχουν καταγραφεί για τα πρωτόκολλα αυτά. Για την πλήρη μελέτη των παρεχόμενων κρυπτογραφικών λειτουργιών θα αναλύσουμε τον κώδικα της εφαρμογής ώστε να διερευνήσουμε τα εξής θέματα:

α) Έλεγχος της ασφάλειας του ZRTP (αναζήτηση πιθανών εργαλείων λογισμικού για την αποτίμησή της, έλεγχος της παρεχόμενης ασφάλειας στην πράξη καθώς και δυνατότητας καλύτερης υλοποίησης του πρωτοκόλλου)

β) έλεγχος του κατά πόσον μπορεί να υποκλαπεί η κρυπτογραφημένη επικοινωνία με τεχνικές τύπου reverse engineering

γ) έλεγχος της δυνατότητας ενίσχυσης της κρυπτογράφησης με προσθήκη κώδικα που να επιτελεί επιπρόσθετη κρυπτογραφική λειτουργία (ή που να αντικαθιστά την προκαθορισμένη), ελέγχοντας ταυτόχρονα και την απόδοση της εφαρμογής - δημιουργία κατάλληλων κρυπτογραφικών βιβλιοθηκών.

δ) έλεγχος της παρεχόμενης ασφάλειας και της συνολικής απόδοσης όταν η εφαρμογή εκτελείται μέσα σε εικονικό ιδιωτικό δίκτυο (VPN). Για την επίτευξη των ανωτέρω θα αξιοποιηθούν εργαλεία λογισμικού, όπως για παράδειγμα το Wireshark για την καταγραφή και περαιτέρω ανάλυση των μεταδιδόμενων πακέτων κατά την επικοινωνία.

1.2 Στόχος της έρευνας

Η εφαρμογή Linphone δεν έχει μελετηθεί εκτενώς ως προς τα ζητήματα στα οποία στοχεύει η παρούσα διατριβή και περιγράφονται ανωτέρω, και συνεπώς οποιοδήποτε

αποτέλεσμα αναδειχθεί θα είναι εξαιρετικά σημαντικό για την ποιοτική αποτίμηση της παρεχόμενης ασφάλειας, αλλά και ενδεχομένως για τα περιθώρια περαιτέρω βελτίωσης αυτής.

1.3 Δομή της Μεταπτυχιακής Διατριβής

1ο Κεφάλαιο: Παρουσιάζεται το αντικείμενο μελέτης της διατριβής, ο σκοπός, τα ερευνητικά ερωτήματα της έρευνας, η αναγκαιότητα και σπουδαιότητα της έρευνας και η δομή της μεταπτυχιακής διατριβής.

2ο Κεφάλαιο: Ανάλυση των ευπαθειών ασφαλείας και των εργαλείων που χρησιμοποιούνται στα πρωτόκολλα SIP (Session Initiation Protocol) των συστημάτων VoIP.

3ο Κεφάλαιο: Μελέτη των βασικών πρωτοκόλλων που υλοποιεί η εφαρμογή LinPhone καθώς και ανάλυση της ροής δεδομένων του.

4ο Κεφάλαιο: Μελέτη και καταγραφή των επιθέσεων ασφαλείας στα πρωτόκολλα ZRTP και TLS, τα οποία αποτελούν τον «πυρήνα» της ασφάλειας του Linphone..

5ο Κεφάλαιο: Πρακτική μελέτη του Linphone, η οποία συνίσταται στη μελέτη του κώδικα της εφαρμογής, παρακολούθηση και ανάλυση της επικοινωνίας του LinPhone με τη χρήση του WireShark, καθώς και αποκάλυψη ευπαθειών που σχετίζονται με τη μη χρήση του TLS, με αντιμετώπιση – μεταξύ άλλων - πραγματικής επίθεσης που έλαβε χώρα.

6ο Κεφάλαιο: Μελέτη του Linphone μέσα από VPN συνδέσεις. Στο πλαίσιο αυτό, έγινε εγκατάσταση VPN server στο cloud και δημιουργία χρηστών σε αυτό. Αναπτύχθηκε ειδική εφαρμογήσε C# για τη διαχείριση των VPN συνδέσεων από την πλευρά του πελάτη (UA) και πραγματοποιήθηκαν κλήσεις μέσω του VPN καναλιού για έλεγχο της απόδοσης της σύνδεσης.

7ο Κεφάλαιο: Συζήτηση για το παρόν και το μέλλον του VoIP, σε σχέση με τους κινδύνους ασφαλείας που πρέπει να αντιμετωπιστούν. Επίσης, παρατίθενται προτάσεις βελτίωσης της εφαρμογής LinPhone.

Επίλογος: Ανασκόπηση της διατριβής, σύνοψη των ερευνητικών της αποτελεσμάτων και συμπεράσματα.

Κεφάλαιο 2
VoIP-ευπάθειες και απειλές

Η αναδυόμενη τεχνολογία μετάδοσης φωνής αλλά και δεδομένων, εικόνων και βίντεο VoIP έχει μεν το πλεονέκτημα του φτηνού, έως και μηδαμινού, κόστους χρήσης καθώς και της ευελιξίας στην χρήση των διαφόρων υπηρεσιών που παρέχει, αλλά από την άλλη γίνεται πόλος έλξης κακόβουλων επιθέσεων.

2.1 Πλαίσιο – υπόβαθρο

Η τεχνολογία VoIP βασίζεται στο πρωτόκολλο SIP (Session Initiation Protocol) για να εδραιώσει, να διατηρήσει και να τερματίσει την επικοινωνία. Το πρωτόκολλο αυτό καθώς στηρίζεται στη δομή του πρωτοκόλλου IP (TCP, UDP ή και SCTP), είναι από τη φύση του ανοικτού τύπου άρα και ευπαθές σε επιθέσεις. Αυτό δεν σημαίνει ότι δεν είναι σχεδιασμένο με ασφάλεια αλλά όσο καλή και να είναι μία κλειδαριά, αν όλοι έχουν πρόσβαση στον τρόπο σχεδιασμού της μπορεί κάθε στιγμή να βρεθεί ένας νέος τρόπος παραβίασής της. Το **VoIP** λοιπόν κληρονομεί από την μία όλα τα θέματα ασφαλείας του πρωτοκόλλου IP αλλά από την άλλη και τους κινδύνους που ελλοχεύουν στο λειτουργικό σύστημα που εδρεύει η εφαρμογή. Οι κίνδυνοι αυτοί καθώς και το γεγονός ότι η επικοινωνία είναι real-time άρα και ευαίσθητη στην απόδοση (performance) και στην ποιότητα (QoS), έχουν γίνει εμπόδια στην ανάπτυξη του. Όσο μάλιστα θα αυξάνει η ασφάλειά του τόσο θα "αδυνατίζει" η ποιότητα της υπηρεσίας η οποία δεν έχει φτάσει ακόμη στα επίπεδα της παραδοσιακής τηλεφωνίας. Θα εξετάσουμε λοιπόν σ' αυτό το κεφάλαιο, διάφορους τρόπους επιθέσεων στο πρωτόκολλο SIP και θα μελετήσουμε την επίδραση τους στο VoIP. Επίσης θα μελετήσουμε και θα αξιολογήσουμε κάποια από τα εργαλεία ανίχνευσης ευπαθειών που είναι διαθέσιμα για ελέγχους ασφαλείας στο VoIP.

2.2 Απειλές στο δίκτυο VoIP (Albers 2005:3)[1]

Η πολυπλοκότητα του VoIP οδηγεί σε μεγάλο αριθμό ευπαθειών που επηρεάζουν το τρίπτυχο της ασφάλειας: Εμπιστευτικότητα, ακεραιότητα, διαθεσιμότητα (τρίπτυχο C.I.A.: Confidentiality, Integrity, Availability). Στον πίνακα 1 παρουσιάζονται οι ευπάθειες ανά περιοχή ανάλογα με το επίπεδο του μοντέλου TCP/IP που επηρεάζουν δηλαδή το φυσικό επίπεδό, το επίπεδο δικτύου, το επίπεδό μεταφοράς και το επίπεδο εφαρμογής. Η λίστα δεν είναι πλήρης διότι περιέχει μόνο τις ευπάθειες οι οποίες μπορούν να εντοπιστούν από διαγνωστικά προγράμματα(Albers 2005:3)[1].

Επίπεδο	Ευπάθεια	Εμπιστευτικότητα Confidentiality	Ακεραιότητα Integrity	Διαθεσιμότητα Availability
Πρόσβασης δικτύου	Φυσικές επιθέσεις - Physical Attacks	√		√
	ARP cache	√	√	√
	ARP flood			√
	MAC spoofing	√	√	√
Δικτύου	IP spoofing			
	Registration server, IP phone, MGCP, DNS, κ.λπ.	√	√	√
	Redirect via IP spoof	√	√	√
	Malformed packets	√	√	√
	IP frag	√	√	√
	Jolt			√
Μεταφοράς	TCP / UDP flood			√
	TCP / UDP replay	√	√	
Εφαρμογής	TFTP server insertion		√	
	DHCP server insertion (redirect)		√	
	DHCP IP address starvation			√
	ICMP flood			√
	SIP			
	Registration Hijacking	√	√	√
	Call Hijacking (MGCP Notified Entity parameter)	√	√	√
	Message body modification	√	√	
	RTP insertion			
	Spoof via header	√	√	√
	Cancel / bye attack			√
	Malformed method			√
	Redirect method	√		√
	RTP			
	SDP redirect			√
	RTP payload			√
	RTP message tampering	√	√	√
	Encryption	√	√	√
	Default settings / passwords	√	√	√
	Disable unnecessary services HTTP, FTP, κ.λπ.	√	√	√
	Buffer overflow	√	√	√
	Legacy Network Interaction	√	√	√
	DNS Availability			√

Πίνακας 1. Ευπάθειες VoIP στους τομείς της εμπιστευτικότητας, της ακεραιότητας και της διαθεσιμότητας (C.I.A.), ταξινομημένες στα τέσσερα επίπεδα του πρωτοκόλλου TCP/IP.

2.2.1 Εμπιστευτικότητα – Confidentiality

Η εμπιστευτικότητα είναι κρίσιμης σημασίας στο VoIP καθώς αναφέρεται στην προστασία των δεδομένων από μη εξουσιοδοτημένη αποκάλυψη. Από τη στιγμή που το VoIP χρησιμοποιεί πρωτόκολλα ανοικτής αρχιτεκτονικής και βασίζεται σε δημόσια δίκτυα, δέχεται σωρεία επιθέσεων εκεί που το κλασικό τηλεφωνικό δίκτυο (PSTN) ήταν προστατευμένο.

2.2.1.1 Επίπεδο Πρόσβασης Δικτύου (Network Interface Layer)

Οι περισσότερες επιθέσεις που αφορούν στο επίπεδο πρόσβασης δικτύου απαιτούν τη συνύπαρξη ευπαθειών που θα εκμεταλλευτεί ο επιτιθέμενος σε άλλο επίπεδο, όπως η φυσική του παρουσία ή προβλήματα στην αυθεντικοποίηση (authentication) του συστήματος. Αν αποκτήσει πρόσβαση μπορεί να εφαρμόσει διάφορες μεθόδους επιθέσεως, όπως η Media Access Control (MAC) address spoofing ώστε να οικειοποιηθεί πληροφορίες που αφορούν στον registry server, gateway, proxy, user agents και άλλες συσκευές. Μπορεί λοιπόν να παραβιάσει την ιδιωτικότητα των συνδιαλέξεων ή και να πραγματοποιήσει κλήσεις VoIP αν και αυτού του είδους η επίθεση συμβαίνει συνήθως στο επίπεδο δικτύου. Στο τρέχον επίπεδο πρόσβασης δικτύου, επιθέσεις τύπου MAC spoofing γίνονται για να καλυφθούν τα ίχνη του επιτιθέμενου(Canavan 2001:4)[2]. Άλλου είδους επίθεση στο τρέχον επίπεδο είναι η ARP επίθεση υπερχείλισης (Address Resolution Protocol (ARP) flood attack). Σε αυτήν ο επιτιθέμενος μπορεί να στείλει εντολές ARP ώστε να διαβρώσει την ARP cache και έτσι να μπορέσει να επαναδρομολογήσει τη κυκλοφορία των πακέτων και να συλλέξει τα δεδομένα που διακινούνται (Kuhn 2005:7)[3]. Ο κίνδυνος επιθέσεων στο τομέα της εμπιστευτικότητας στο επίπεδο αυτό μπορεί να περιοριστεί αν όχι να εξαλειφθεί με περιορισμό της φυσικής πρόσβασης και με την υιοθέτηση ισχυρών μηχανισμών αυθεντικοποίησης.

2.2.1.2 Επίπεδο Δικτύου (Network Layer)

Η κυριότερη ευπάθεια του επιπέδου δικτύου είναι η μεταμφίεση των διευθύνσεων (address spoofing) με την οποία ο επιτιθέμενος μπορεί να υποδυθεί (impersonate) πολλές διαφορετικές συσκευές υφαρπάζοντας και χρησιμοποιώντας τις IP διευθύνσεις αυτών των συσκευών. Οι registration servers, οι SIP proxy servers, τα IP τηλέφωνα, οι Media Gateway Control Protocol (MGCP) servers, καθώς και οι Domain Name Server (DNS) servers είναι μερικές από τις συσκευές αυτές. Οικειοποιούμενος τη ταυτότητα ενός “νόμιμου” χρήστη, μπορεί να πραγματοποιήσει μη εξουσιοδοτημένες κλήσεις.

Μπορεί να αποκρυπτογραφήσει την IP διεύθυνση ενός τηλεφώνου καλώντας απλώς τον αριθμό του και καταγράφοντας τα πακέτα που μεταδίδονται. Η αποκάλυψη της IP διεύθυνσης ενδεχομένως δεν είναι κρίσιμη για την ασφάλεια ούτε δηλώνει ευπάθεια για το σύστημα αλλά σε κάθε περίπτωση μπορεί ο επιτιθέμενος να εστιάσει την επίθεση σε αυτό το τηλέφωνο για υποκλοπές. Ένας τρόπος υποκλοπής των συζητήσεων γίνεται πλαστογραφώντας την default gateway's IP διεύθυνση ώστε να δρομολογηθούν όλα τα VoIP πακέτα σε συσκευή του επιτιθέμενου. Οι μηχανισμοί IDS (Intrusion Detection Systems) δεν μπορούν πάντα να εντοπίσουν την διαδικασία προώθησης IP (forwarding) η οποία γίνεται με τον παραδοσιακό τρόπο. Αν μάλιστα τα IP τηλέφωνα υποστηρίζουν απομακρυσμένη διαχείριση τότε αυξάνεται ο κίνδυνος τέτοιων επιθέσεων. Η χρήση firewall με δημιουργία φίλτρων στο επίπεδο δικτύου μπορεί να μετριάσει τον κίνδυνο τέτοιων επιθέσεων(Kuhn 2005:8) [3].

2.2.1.3 Επίπεδο Μεταφοράς (Transport Layer)

Οι ευπάθειες αυτού του επιπέδου σχετίζονται με το RTP (Real Time Protocol) όπου ο επιτιθέμενος μπορεί να υποκλέψει τα RTP πακέτα και με τη χρήση εφαρμογών κατάλληλων για packet sniffing όπως το VOMIT (Voice Over Misconfigured Internet Telephones) να συλλέξει όλη την RTP συνεδρία. Το VOMIT όχι μόνο συλλέγει πακέτα VoIP την ώρα που μεταφέρονται αλλά κατόπιν μετατρέπει τα δεδομένα σε ηχητικά αρχεία.

Οι τεχνολογίες κρυπτογράφησης IPSec (IP Security), TLS (Transport Layer Security) και SRTP (Secure Real Time Protocol) είναι σε θέση να ελαχιστοποιήσουν τον κίνδυνο αυτής της μορφής επιθέσεως (Dadoun 2002:5) [4].

Η κρυπτογράφηση στη τεχνολογία VoIP δεν είναι πανάκεια, αφού εξακολουθούν να υπάρχουν ευπάθειες που θα πρέπει να αντιμετωπιστούν, όπως για παράδειγμα επιθέσεις τύπου buffer overflow, malformed πακέτων, replay attacks και προβλήματα φυσικής ασφαλείας. Για παράδειγμα σε μία ανταλλαγή κλειδιών TLS, ο επιτιθέμενος μπορεί να συγκεντρώσει τα πακέτα και να προσπαθήσει να τα αποκρυπτογραφήσει ή να εφαρμόσει μία τυπική man-in-the-middle επίθεση (Thalhammer 2002:5) [5].

2.2.1.4 Επίπεδο Εφαρμογής (Application Layer)

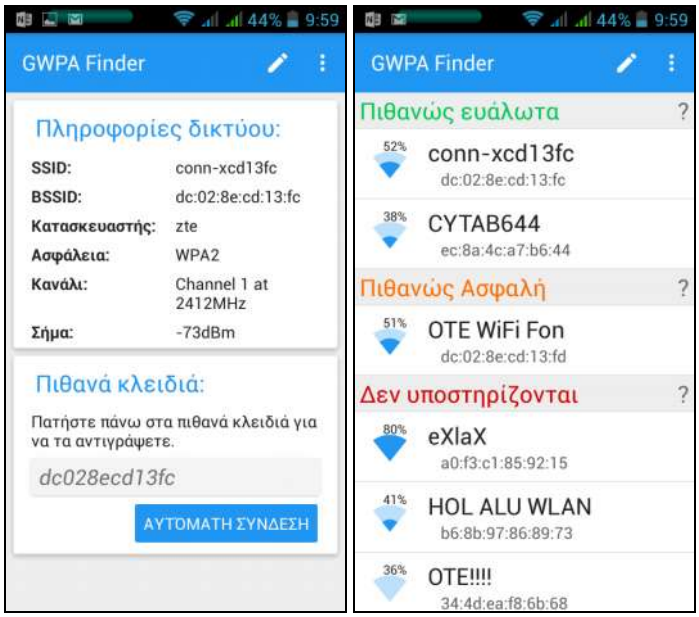
Το πρωτόκολλο SIP καθορίζει συγκεκριμένα πρότυπα για αυθεντικοποίηση (authentication). Στην αρχική του έκδοση μετέδιδε το username και το password σε μορφή απλού κειμένου. Πλέον το SIP υποστηρίζει αυθεντικοποίηση μέσω HTTP και μέσω S/MIME το οποίο περιλαμβάνει έναν ασφαλέστερο μηχανισμό κρυπτογράφησης. Βέβαια το πρωτόκολλο HTTP δεν ανταποκρίνεται στα διεθνή πρότυπα ασφάλειας όπως το PKI (Public Key Infrastructure) εξ' αιτίας ανεπαρκούς διαχείρισης κλειδιών και έλλειψη μίας αρχής πιστοποίησης (Certification Authority). Όσον αφορά το S/MIME πρωτόκολλο έχει διαπιστωθεί ότι είναι επιρρεπές σε επιθέσεις τύπου man-in-the-middle όπου ο επιτιθέμενος μπορεί να αποκτήσει το κλειδί κατά την έναρξη της συναλλαγής (Thomas 2001:4) [6].

Μία άλλη μορφή απλής επίθεσης στη VoIP τηλεφωνία που βασίζεται στο πρωτόκολλο SIP είναι αυτή που γίνεται την ώρα της εγγραφής (registration) κατά την οποία ο μηχανισμός εγγραφής του SIP σε έναν user agent δηλαδή στο λογισμικό που εκτελείται στη συσκευή του χρήστη ώστε να επιτευχθεί η σύνδεση VoIP, του επιτρέπει να πιστοποιηθεί. Αλλάζοντας απλώς πεδία στην SIP αίτηση (request), ο επιτιθέμενος μπορεί να πραγματοποιήσει κλήσεις ή να ανακατευθύνει τη κλήση εκεί που θέλει.

Η σχεδίαση του SIP εμπεριέχει εντολές ανακατεύθυνσης (REDIRECT commands) που επιτρέπουν στον χρήστη να μεταφέρει την κλήση του ενώ είναι σε εξέλιξη σε άλλο περιβάλλον π.χ. από επίγειο σε ασύρματο τηλέφωνο. Εδώ μπορεί να λάβει χώρα μία επίθεση κατάληψης κλήσης (call hijack attack) με τον επιτιθέμενο να στέλνει μήνυμα REINVITE ώστε να δρομολογηθεί η ροή RTP σε IP διεύθυνση του επιτιθέμενου. Ο καλλών δέχεται επίθεση τύπου DOS (Denial Of Service) με αποτέλεσμα να χάσει τη σύνδεσή του. Ο παραλήπτης του μηνύματος REINVITE συνδέεται πλέον απευθείας στον επιτιθέμενο ο οποίος και ακούει πλέον ότι λέει ο παραλήπτης χάνοντας έτσι η σύνδεση την εμπιστευτικότητα της καθώς και την ιδιωτικότητα της (Wu 2004:435) [7].

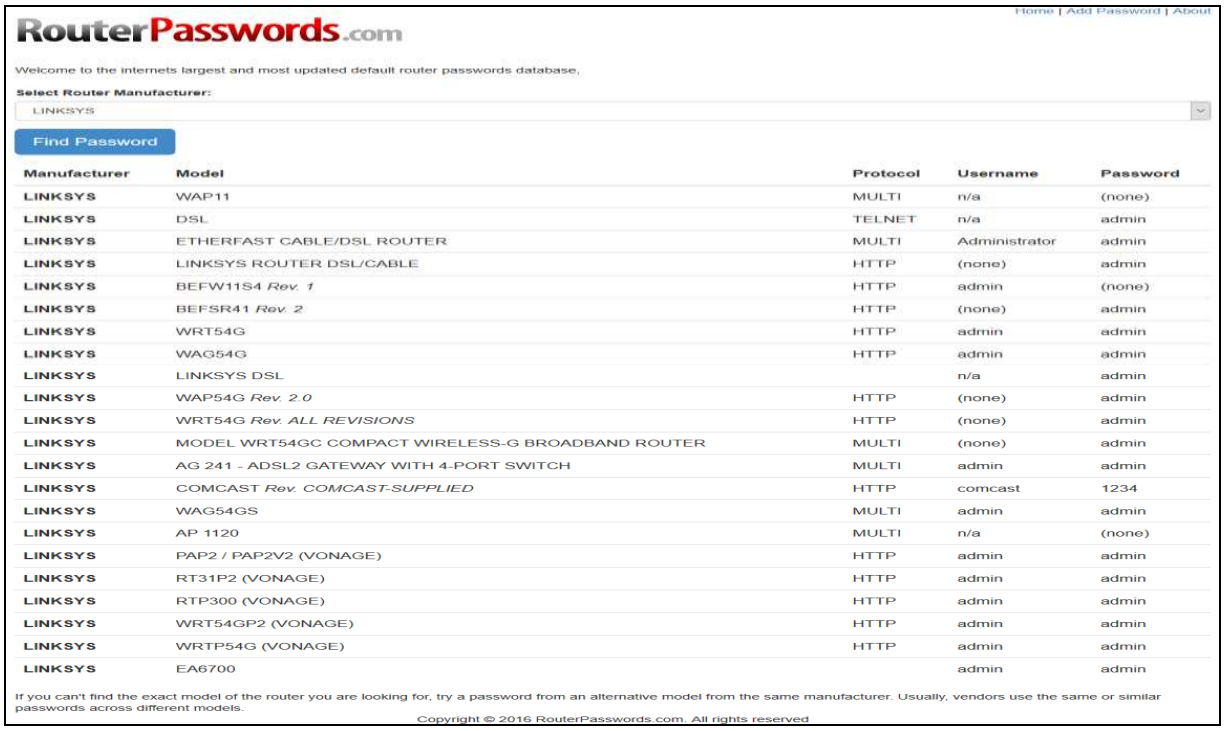
Το Instant Messaging (IM) είναι άλλη μία υπηρεσία που υποστηρίζει το SIP, το οποίο έχει παρουσιάσει διάφορες ευπάθειες ασφαλείας όπως τη παραποίηση της επικεφαλίδας (header) του IM πακέτου ώστε να σταλθεί μήνυμα το οποίο να φαίνεται ότι το έχει στείλει ο "νόμιμος" χρήστης. Το μήνυμα μπορεί να περιέχει συνδέσμους σε malware ή να ζητάει εμπιστευτικές πληροφορίες (phishing) (Wu 2004:435) [7].

Στην πράξη, όλες οι σύγχρονες συσκευές δικτύωσης διακινούνται με προκαθορισμένα user names, passwords αλλά και ρυθμίσεις. Αυτές τις "εργοστασιακές" ρυθμίσεις είναι διαθέσιμες στο κοινό από τις εταιρείες κατασκευής άρα είναι πάρα πολύ εύκολο να τις βρει κάποιος με απλή αναζήτηση στο Διαδίκτυο. Όπως είναι φυσικό, η ύπαρξη των αρχικών ρυθμίσεων μπορεί να οδηγήσει σε ένα πλήθος επιθέσεων όπως να στέλνει ένα switch τα πακέτα του ΚΑΙ σε άλλη πόρτα από την αρχική (port mirroring) χωρίς να σημάνει κανένας συναγερμός στο σύστημα. Υπάρχουν εφαρμογές ακόμη και σε smartphones οι οποίες δημιουργούν σε ελάχιστο χρόνο τα προκαθορισμένα κλειδιά των modem-router για ασύρματη πρόσβαση σε αυτά.



Εικόνα 2: Η εφαρμογή GWPA Finder για Android εμφανίζει τα προκαθορισμένα από τους κατασκευαστές κλειδιά Wi-Fi πρόσβασης στους οικιακούς modem-routers.

Ακόμη και ιστοσελίδες στις οποίες αφού επιλέξεις την εταιρεία κατασκευής του router που σε ενδιαφέρει, εμφανίζει όλα τα default user-names και passwords ανά μοντέλο.



RouterPasswords.com

Home | Add Password | About

Welcome to the internets largest and most updated default router passwords database,

Select Router Manufacturer:

LINKSYS

Find Password

Manufacturer	Model	Protocol	Username	Password
LINKSYS	WAP11	MULTI	n/a	(none)
LINKSYS	DSL	TELNET	n/a	admin
LINKSYS	ETHERFAST CABLE/DSL ROUTER	MULTI	Administrator	admin
LINKSYS	LINKSYS ROUTER DSL/CABLE	HTTP	(none)	admin
LINKSYS	BEFW11S4 *Rev. 1*	HTTP	admin	(none)
LINKSYS	BEFSR41 *Rev. 2*	HTTP	(none)	admin
LINKSYS	WRT54G	HTTP	admin	admin
LINKSYS	WAG54G	HTTP	admin	admin
LINKSYS	LINKSYS DSL		n/a	admin
LINKSYS	WAP54G *Rev. 2.0*	HTTP	(none)	admin
LINKSYS	WRT54G *Rev. ALL REVISIONS*	HTTP	(none)	admin
LINKSYS	MODEL WRT54GC COMPACT WIRELESS-G BROADBAND ROUTER	MULTI	(none)	admin
LINKSYS	AG 241 - ADSL2 GATEWAY WITH 4-PORT SWITCH	MULTI	admin	admin
LINKSYS	COMCAST *Rev. COMCAST-SUPPLIED*	HTTP	comcast	1234
LINKSYS	WAG54GS	MULTI	admin	admin
LINKSYS	AP 1120	MULTI	n/a	(none)
LINKSYS	PAP2 / PAP2V2 (VONAGE)	HTTP	admin	admin
LINKSYS	RT31P2 (VONAGE)	HTTP	admin	admin
LINKSYS	RTP300 (VONAGE)	HTTP	admin	admin
LINKSYS	WRT54GP2 (VONAGE)	HTTP	admin	admin
LINKSYS	WRTP54G (VONAGE)	HTTP	admin	admin
LINKSYS	EA6700		admin	admin

If you can't find the exact model of the router you are looking for, try a password from an alternative model from the same manufacturer. Usually, vendors use the same or similar passwords across different models.

Εικόνα 3: Η ιστοσελίδα routerpassords.com με τα προκαθορισμένα user-names και passwords όλων των κατασκευαστών modem-router.

Η άμεση αλλαγή των προκαθορισμένων ρυθμίσεων θεωρείται λοιπόν εκ των ουκ άνευ καθώς και η απενεργοποίηση του port mirroring. Επίσης, η λειτουργία απομακρυσμένης διαχείρισης (remote administration) πρέπει να απενεργοποιείται ή να κρυπτογραφείται ισχυρά ώστε να εμποδίζεται η σύλληψη (capturing) από packet sniffers των ρυθμίσεων της συσκευής και των διαπιστευτηρίων (credentials δηλ. user-name, password).

Σήμερα, πάρα πολλές συσκευές όπως switches και τερματικά VoIP, υλοποιούν και παρέχουν υπηρεσίες HTTP server. Η κυκλοφορία όμως των δεδομένων αυτής της υπηρεσίας τα καθιστά ευάλωτα στην υποκλοπή από οποιονδήποτε έχει πρόσβαση στο τοπικό δίκτυο. Κρίνεται λοιπόν σκόπιμο να απενεργοποιείται αυτή η υπηρεσία ή αν είναι απαραίτητη τότε να υλοποιείται με ισχυρή κρυπτογράφηση όπως SSL (Secure Sockets Layer) (Kuhn 2005:14) [3], χωρίς να παραλείπουμε το γεγονός ότι και άλλες υπηρεσίες απομακρυσμένης διαχείρισης όπως Telnet ή υπηρεσίες μεταφοράς όπως FTP (File Transfer Protocol) και TFTP (Trivial File Transfer Protocol) πρέπει να υλοποιούνται μόνο με ενεργοποιημένη κρυπτογράφηση (Dadoun 2002:8) [4].

Οι εφαρμογές που υλοποιούν υπηρεσίες VoIP σε αντίθεση με τις συσκευές VoIP προσθέτουν ευπάθειες στο σύστημα που λειτουργούν καθώς η κυκλοφορία φωνής και δεδομένων στο ίδιο δίκτυο μεταδίδει τις αδυναμίες του ενός στο άλλο. Το δίκτυο φωνής κληρονομεί τις ευπάθειες όλων των servers, των Λειτουργικών Συστημάτων (Λ.Σ.), των switches, των routers και όλων των εφαρμογών του δικτύου δεδομένων το οποίο με τη σειρά του κληρονομεί τις αδυναμίες της αρχιτεκτονικής VoIP. Αν κάνουμε κάποιο συμβιβασμό σε θέμα ασφαλείας στο ένα δίκτυο θα επηρεαστεί και το άλλο. Αυτή η αλληλεξάρτηση των δικτύων αυξάνει τον κίνδυνο ζημιών σε περίπτωση επιτυχούς επίθεσης στο σύστημα. Επίσης, οι VoIP εφαρμογές κληρονομούν τα ελαττώματα των λειτουργικών συστημάτων που τις φιλοξενούν διότι και τα λειτουργικά συστήματα όπως και άλλες εφαρμογές περιέχουν προγραμματιστικές ατέλειες όπως ανεπαρκής διαχείριση μνήμης η οποία μπορεί να οδηγήσει σε σενάρια άρνησης εξυπηρέτησης (DOS: Denial Of Service), ή ακόμη χειρότερα στην απόκτηση ελέγχου του συστήματος παρέχοντας έτσι πρόσβαση σε εμπιστευτικές πληροφορίες (Thalhammer 2002:7) [5].

Εφαρμογές με αδυναμίες στη διαχείριση της μνήμης που φιλοξενούνται σε Λ.Σ. τα οποία δεν είναι σε θέση να τις προφυλάξουν μπορεί να δεχθούν επιθέσεις υπερχείλισης προσωρινής μνήμης (Buffer Overflow Attacks). Για να αποφύγουμε αυτό το σκόπελο πρέπει όλες οι εφαρμογές καθώς και το ίδιο το Λ.Σ. να ενημερώνεται τακτικά και να είναι πάντα στη τελευταία έκδοση. Προτείνεται επίσης η απενεργοποίηση των υπηρεσιών (services) που δεν χρειάζονται διότι κάθε τι που “τρέχει” σε ένα Λ.Σ. αυξάνει τον κίνδυνο επιτυχούς επίθεσης και οι υπηρεσίες που είναι απαραίτητες, να λειτουργούν με χαμηλά δικαιώματα (privileges).

Εκτός από τις κληρονομικές απειλές, τα συστήματα VoIP υπόκεινται και σε επιθέσεις όχι άμεσες αλλά καταστροφικές. Τέτοιο παράδειγμα είναι η ύπαρξη αδύναμων κωδικών οι οποίοι μπορούν εύκολα να βρεθούν με επιθέσεις τύπου brute force και να αποκτηθεί πρόσβαση στο σύστημα VoIP (Si DF 2004:575) [8]. Ο επιτιθέμενος μετά, μπορεί να έχει πρόσβαση στους πόρους του συστήματος, σε εμπιστευτικές πληροφορίες ή να “φυτέψει” malware εφαρμογές. Λόγω της πληθώρας κακόβουλων εφαρμογών οι οποίες κυκλοφορούν στα δημόσια δίκτυα, τα συστήματα VoIP είναι πιθανό να βασιστούν σε τεχνολογίες που επηρεάζονται από viruses, worms κ.λπ.

2.2.2 Ακεραιότητα - Integrity

Ακεραιότητα σημαίνει προστασία από μη εξουσιοδοτημένη τροποποίηση των δεδομένων και στην περίπτωση του VoIP σημαίνει τη μη εξουσιοδοτημένη τροποποίηση ή διαγραφή των φωνητικών ως επί το πλείστο δεδομένων, κωδικών πρόσβασης, ρυθμίσεων κ.λπ. Για την αποφυγή αυτών των απειλών οι εφαρμογές VoIP χρησιμοποιούν κρυπτογράφηση, αλγορίθμους κατακερματισμού (hash) κ.λπ.

2.2.2.1 Επίπεδο Πρόσβασης Δικτύου

Στο τομέα της εμπιστευτικότητας αναφέραμε ήδη ότι τα VoIP συστήματα είναι ευαίσθητα σε επιθέσεις υπερχείλισης ARP με αποτέλεσμα την ανακατεύθυνση της συνομιλίας VoIP που είναι σε εξέλιξη. Υπάρχει όμως και μία επίδραση στον τομέα της Ακεραιότητας όπου ο επιτιθέμενος μπορεί να μεταβάλει τον πίνακα ARP του αποστολέα μεταβάλλοντας έτσι την ακεραιότητά του.

2.2.2.2 Επίπεδο Δικτύου

Ελλείψει πρωτοκόλλου ελέγχου αυθεντικοποίησης από υψηλότερο επίπεδο, είναι δυνατόν να εμφανιστεί επίθεση τύπου IP address spoofing όπου μπορεί να μεταβληθεί η IP διεύθυνση του αποστολέα και να μεταβληθούν δεδομένα ενδεχομένως και σε όλο το σύστημα (Canavan 2001:6)[2]

2.2.2.3 Επίπεδο Μεταφοράς

Σε μία κλασική επίθεση τύπου man-in-the-middle, ο επιτιθέμενος μπορεί να συλλάβει πακέτα κατά τη στιγμή της μετάδοσής τους και είτε να τα προωθήσει ως έχουν ή αφού τα μεταβάλλει αφαιρώντας πληροφορίες από αυτά ή και προσθέτοντας πληροφορίες. Επιπλέον, το VoIP είναι ευάλωτο σε επιθέσεις επανάληψης (replay attack) τόσο σε TCP όσο και σε UDP πρωτόκολλα, όπου ο επιτιθέμενος είναι σε θέση να συλλάβει ολόκληρη συνεδρία (full session) και να αναμεταδίδει τα μηνύματα των μελών που επικοινωνούν (ή ακόμη και κάποιου τρίτου) μετά το πέρας της συνεδρίας. (Si DF 2004:575) [8]

2.2.2.4 Επίπεδο Εφαρμογής

Στο επίπεδο αυτό εμφανίζεται η πληθώρα των απειλών σε θέματα ακεραιότητας σε ένα σύστημα VoIP. Μία τέτοια απειλή είναι η επίθεση εισαγωγής πλαστού DHCP server (DHCP server insertion attack), όπου ο επιτιθέμενος εγκαθιστά ένα πλασματικό DHCP server έτσι ώστε όταν ένας UA (User Agent) κάνει αίτηση DHCP, ο server του απαντά με

ψευδή στοιχεία μεταβάλλοντας έτσι την IP διεύθυνση του, τους DNS servers του και το πίνακα δρομολόγησης του (routing table) (Kuhn 2005:17) [3]. Παρόμοια επίθεση μπορεί να γίνει εισάγοντας στο σύστημα ένα TFTP server, κάτι που οδηγεί συνήθως σε ολοκληρωτική αλλαγή των ρυθμίσεων της συσκευής VoIP (Dadoun 2002:11) [4]. Και στις δύο αυτές επιθέσεις, ο κίνδυνος μπορεί να μετριαστεί χρησιμοποιώντας μεθόδους αυθεντικοποίησης (authentication) και φιλτραρίσματος των διευθύνσεων.

Το πρωτόκολλο SIP έχει αρκετά τρωτά σημεία σε θέματα ακεραιότητας όπως είναι οι επιθέσεις που έχουν στόχο την τροποποίηση του (SIP modification attack). Σε αυτές ο επιτιθέμενος διακόπτει τη διαδρομή του σήματος και προσπαθεί να τροποποιήσει τα μηνύματα σκοπεύοντας στη μεταβολή κάποιων χαρακτηριστικών της υπηρεσίας όπως στην αλλαγή της διαδρομής του σήματος ή στην αλλαγή της εγγραφής του χρήστη (user registration) ή στη μεταβολή του προφίλ μίας υπηρεσίας (Salsano 2002:40) [9].

Μία άλλη μορφή επίθεσης μεταβάλλει το κυρίως τμήμα ενός μηνύματος αλλάζοντας τα κλειδιά κρυπτογράφησης της συνόδου, τα τμήματα MIME και τα SDP και τα ενθυλακωμένα (encapsulated) σήματα μέσα στο ίδιο το SIP. Έτσι μπορεί να γίνει ανακατεύθυνση της ροής RTP ή να υποκλαπεί η συνομιλία όπως επίσης και να μεταβληθούν πληροφορίες στο πεδία της επικεφαλίδας (header) (Si DF 2004:576) [8].

Σε προηγούμενη ενότητα (2.2.1.4) αναφέραμε ότι ο μηχανισμός SIP κατά την ώρα του registration, είναι ευάλωτος σε επιθέσεις εμπιστευτικότητας. Η κυριότερη όμως ευπάθεια του registration server είναι στον τομέα της ακεραιότητας όπου οι επιθέσεις που δέχεται μπορεί να οδηγήσουν σε ψευδείς καταχωρήσεις στο μητρώο (registry) επηρεάζοντας έτσι την ακεραιότητα των εγγραφών (Dadoun 2002:14) [4].

Η εξάρτηση του VoIP από το RTP το καθιστά ευάλωτο σε επιθέσεις εισαγωγής (insertion attacks), όπου ο επιτιθέμενος εισάγει μηνύματα όπως λέξεις ή άλλους θορύβους στη ροή (stream) του RTP και αυτά παραδίδονται σε έναν από τους χρήστες(Kuhn 2005:21) [3]. Τα RTP πακέτα που τοποθετούνται εσκεμμένα σε μία άλλη ροή RTP λειτουργούν ως ένα είδος επίθεσης “επανάληψης/εισαγωγής” (replay/insertion attack) για να μιμηθεί ένα νόμιμο χρήστη.

Η διαδικασία αυθεντικοποίησης του χρήστη είναι ακόμη ένας φορέας επίθεσης στα συστήματα VoIP. Η πρόσβαση σε λογαριασμούς χρηστών παρέχει έναν τρόπο στον εισβολέα για να διαβάσει, να διαγράψει ή να τροποποιήσει δεδομένα και ρυθμίσεις. Αδύναμοι, προεπιλεγμένοι και κενοί κωδικοί πρόσβασης καθιστούν εύκολο για έναν εισβολέα να αποκτήσει πρόσβαση στους λογαριασμούς. Όπως σε όλους τους τομείς της ασφάλειας των δικτύων, τα κακογραμμένα λογισμικά προσθέτουν σημαντικότατο κίνδυνο στην ακεραιότητα του VoIP, π.χ. μία ευπάθεια buffer overflow που έχει ανακαλυφθεί μπορεί να επιτρέψει μια επίθεση τροποποίησης δεδομένων και ρυθμίσεων.

2.2.3 Διαθεσιμότητα - Availability

Η διαθεσιμότητα διαβεβαιώνει ότι τα αποθηκευτικά μέσα και τα μέσα μεταφοράς είναι προσβάσιμα στους εξουσιοδοτημένους χρήστες κάθε στιγμή που ζητούν πρόσβαση. Οι επιθέσεις που απευθύνονται στη διαθεσιμότητα ανήκουν κατά κύριο λόγο στις επιθέσεις γνωστές ως επιθέσεις "άρνησης παροχής υπηρεσιών (DOS-Denial of Service)" και μπορούν να ταξινομηθούν σε τέσσερεις κατηγορίες(Canavan 2001:12)[2]:

- **κατανάλωσης εύρους ζώνης - bandwidth consumption,**

  Οι επιθέσεις κατανάλωσης του εύρους ζώνης (BandWidth consumption) δημιουργούνται συνήθως "πλημμυρίζοντας" το δίκτυο με κάποια συγκεκριμένης μορφής κυκλοφορία. Η ποσότητα της κυκλοφορίας καταναλώνει τελικά αρκετό εύρος ζώνης ώστε να μειωθεί σημαντικά ή και να εξαλειφθεί ακόμη η ροή της κυκλοφορίας. Όμως ένας υπολογιστής μόνος του δεν είναι σε θέση να δημιουργήσει τη ποσότητα της κίνησης που απαιτείται για να δημιουργήσει πρόβλημα και έτσι αυτού του είδους οι επιθέσεις πραγματοποιούνται με τη συμμετοχή πολλών υπολογιστών που βρίσκονται κάτω από τον έλεγχο ενός χάκερ μέσω της χρήσης ενός ιού/σκουληκιού. Προληπτικά μέτρα συνήθως περιλαμβάνουν τη λειτουργία ενός Συστήματος Ανίχνευσης / Πρόληψης Εισβολέων (Intrusion Detection / Prevention System (IDS/IPS) για να εντοπισθεί και να μετριαστεί η επίθεση (Dadoun 2002:17) [4].
- **κατανάλωσης πόρων – resource starvation,**

  Οι επιθέσεις τύπου εξάντλησης πόρων (resource starvation) κατακλύζουν μία συσκευή αντί για το μέσο. Ο επιτιθέμενος συνήθως δημιουργεί αιτήσεις σε μία

συσκευή με πολύ γρήγορο και εξαντλητικό ρυθμό ώστε να εξαντληθούν όλοι οι διαθέσιμοι πόροι της συσκευής και να σταματήσει η διαθεσιμότητα υπηρεσιών σε άλλους χρήστες (Dadoun 2002:18) [4].

- **επιθέσεις στην δρομολόγηση πακέτων – routing attack**
  Οι επιθέσεις δρομολόγησης αφορούν στο χειρισμό των πληροφοριών δρομολόγησης ή των πρωτοκόλλων αυτής ώστε να υποκλαπεί ή να διακοπεί η διακίνηση της πληροφορίας. Συνήθως επιτυγχάνεται με τη πλαστογράφηση στοιχείων στα πακέτα ή εισάγοντας ολόκληρα πακέτα με πλαστές πληροφορίες. Τα προληπτικά μέτρα αντιμετώπισης των επιθέσεων αυτών περιλαμβάνουν τη προστασία των συσκευών μέσω φυσικής ασφάλειας και κρυπτογράφησης.

- **προγραμματιστικά ελαττώματα – programming flaws.**
  Οι προγραμματιστικές αστοχίες-ατέλειες είναι ακούσια σφάλματα στο λογισμικό που μπορεί να τα ανακαλύψει και να τα εκμεταλλευθεί ένας κακόβουλος χρήστης προκειμένου να αποκτήσει πρόσβαση σε ένα σύστημα. Αυτές οι προγραμματιστικές ρωγμές συνήθως συμβαίνουν λόγω χαμηλών-χαλαρών προτύπων κατά τη διαδικασία δημιουργίας της εφαρμογής, ως αποτέλεσμα της βιασύνης "να μπει στην αγορά". Τα προληπτικά μέτρα είναι συνήθως "πυροσβεστικά" εκδίδοντας δηλαδή εκ των υστέρων επιδιορθώσεις (patches) (Dadoun 2002:19) [4].

2.2.3.1 Επίπεδο Πρόσβασης Δικτύου

Στο επίπεδο αυτό, οι επιθέσεις που αφορούν στη διαθεσιμότητα χωρίζονται σε δύο κατηγορίες: στις επιθέσεις φυσικής ασφάλειας και στις επιθέσεις που αφορούν στις λειτουργίες του 2ου επιπέδου του OSI. Η φυσική ασφάλεια προστατεύει τις συσκευές του δικτύου, περιορίζοντας την πρόσβαση. Με αυτό το τρόπο αποτρέπεται η πρόκληση φυσικής καταστροφής σε μία συσκευή όπως να κοπεί ένα καλώδιο σύνδεσης ή να διακοπεί η τροφοδοσία της σε ρεύμα. Τα προγράμματα ασφαλείας δεν μπορούν να κάνουν πολλά για τον εντοπισμό και την αποτροπή τέτοιων πράξεων.

Η κυριότερη επίθεση στο 2ο επίπεδο του OSI αφορά στο ARP και εκδηλώνεται με επιθέσεις spoofing, poisoning, flooding και cache (McKeag 2004:326, Spangler 2003:1) [10],[11] με βασικότερη την επίθεση υπερχείλισης όπου αποστέλλονται σε μία συσκευή

πλαστογραφημένες απαντήσεις ARP ώστε να υπερχειλίσει η προσωρινή μνήμη cache. Σε μία επίθεση αυτού του είδους τα αποτελέσματα ποικίλλουν από την κατανάλωση του εύρους ζώνης ως και τη κατάρρευση του συστήματος, με την εξάντληση των πόρων του επεξεργαστή. Η μεγαλύτερη αδυναμία του ARP απαντάται στο γεγονός ότι είναι stateless πρωτόκολλο δηλαδή, δεν περιμένει απαντήσεις σε ερωτήσεις που το ίδιο έθεσε, παρά απαντάει σε οποιαδήποτε ερώτηση χωρίς να έχει στείλει κάποια αίτηση. Ως αποτέλεσμα, ένας εισβολέας μπορεί να χειριστεί την ARP cache μιας συσκευής, στέλνοντας πλαστές ARP απαντήσεις οι οποίες δεν ζητήθηκαν ποτέ. Ο εισβολέας συνήθως εισάγει μια ανύπαρκτη διεύθυνση προκειμένου να δημιουργήσει μία "μαύρη τρύπα" όπου η κυκλοφορία θα κατευθύνεται εκεί ή την εισαγωγή συγκεκριμένων διευθύνσεων για να ανακατευθύνει την επικοινωνία(Kuhn 2005:24) [3].

2.2.3.2 Επίπεδο Δικτύου

Η διαθεσιμότητα και η εμπιστευτικότητα επηρεάζονται στο επίπεδο δικτύου κυρίως με τη πλαστογράφηση (spoofing) των IP διευθύνσεων των συσκευών του δικτύου, προκειμένου να υποκλαπεί ή/και να ανακατευθυνθεί η κίνηση. Παρόμοια με την επίθεση ARP cache είναι και η επίθεση που εστιάζει στη μάσκα υποδικτύου (netmask) των IP τηλεφώνων, με την οποία μπορεί να αλλαχτεί η διεύθυνση IP και η μάσκα ενός δρομολογητή (router) ώστε να ανακατευθυνθούν τα πακέτα σε άλλη συσκευή. Δύο πολύ κοινές επιθέσεις αλλοίωσης των πακέτων δεδομένων είναι η επίθεση κατακερματισμού IP (IP fragmentation) όπου δημιουργούνται μεγάλα πακέτα τα οποία πρέπει να διασπαστούν πριν αποσταλούν. Κατά την συναρμολόγηση αυτών, στον παραλήπτη, τα δεδομένα μπορεί να αληλλοκαλύπτονται και να δυσχεραίνει η διαδικασία και να εμφανιστούν προβλήματα έλλειψης πόρων ή και κατάρρευσης του συστήματος (Kaufman 2003:5) [12]. Η δεύτερη επίθεση είναι η επίθεση jolt, που ανήκει στις DoS (Denial of Service) επιθέσεις. Σε αυτήν, το IP datagram πακέτο διασπάται σε μικρά τμήματα με αποτέλεσμα ο επεξεργαστής να εξαντλεί μέχρι και το 100% της ισχύος του για την συναρμολόγηση αυτών.

2.2.3.3 Επίπεδο Μεταφοράς

Η διαθεσιμότητα, στο επίπεδο μεταφοράς, επηρεάζεται από επιθέσεις υπερχείλισης ή κατακερματισμού τόσο σε TCP όσο και σε UDP πρωτόκολλο. Στην επίθεση υπερχείλισης TCP SYN, κατευθύνονται κακόβουλες TCP αιτήσεις σύνδεσης στον server. Αυτές οι αιτήσεις δεν είναι σε θέση να ολοκληρώσουν την τριμερή χειραψία (three-way

handshake) και έτσι η σύνδεση δεν ολοκληρώνεται. Μετέπειτα πλαστά αιτήματα δημιουργούν περισσότερες "μισάνοιχτες" συνδέσεις και έτσι ο server εξαντλεί τα αποθέματα μνήμης του (Angelo 2009:3) [13]. Ομοίως, στην UDP υπερχείλιση, τα πακέτα κατευθύνονται σε μία τυχαία πόρτα στη συσκευή. Μόλις ο δέκτης καταλάβει ότι η θύρα δεν υπάρχει, απαντάει ότι ο ICMP προορισμός δεν είναι προσβάσιμος. Ο επιτιθέμενος όμως συνεχίζει να στέλνει κακόβουλα πακέτα δημιουργώντας έτσι ένα ατελείωτο βρόγχο και τελικά αφού καταναλωθεί η διαθέσιμη μνήμη καταρρέει το σύστημα (Angelo 2009:4) [13].

2.2.3.4 Επίπεδο Εφαρμογής

Η διαθεσιμότητα επηρεάζεται κατά κύριο λόγο σε αυτό το επίπεδο με το SIP να κατέχει τα πρωτεία σε ευπάθειες. Μία συνηθισμένη μορφή επίθεσης είναι μέσω DoS όπου ο επιτιθέμενος πλαστογραφεί την IP διεύθυνση και κατόπιν αλλάζει την επικεφαλίδα (header) της αίτησης εισάγοντας πληροφορίες που ανήκουν στο "θύμα" και κατόπιν στέλνει την αίτηση σε πολλούς SIP UA's ή σε proxies δημιουργώντας κίνηση προς το θύμα. Στην επίθεση στο SIP γνωστή ως Cancel/Bye, ο επιτιθέμενος μπορεί να τερματίσει μία κλήση στέλνοντας ένα πλαστογραφημένο μύνημα " Cancel " ή " Bye " σε μία συσκευή (Wu 2004:436) [7]. Με παρόμοιο τρόπο μπορεί να επιτευχθεί και μία επίθεση τύπου "απρόσιτης ICMP πόρτας (ICMP port unreachable) όπου ο επιτιθέμενος μπορεί να τερματίσει μία κλήση στέλνοντας ένα πλαστογραφημένο μήνυμα "ICMP Port Unreachable" σε μία συσκευή (Tech whitepaper 2004:2) [14]. Επίσης, ο επιτιθέμενος μπορεί να προσποιηθεί ότι είναι κάποιος UA, κατά τη διαδικασία καταχώρησης (registration) στο SIP η κατά τη διαδικασία της ανακατεύθυνσης (REDIRECT) ώστε να αλλάξει τις πληροφορίες του UA (Wu 2004:437) [7]. Μία πετυχημένη επίθεση μπορεί να αφαιρέσει τις επαφές (contacts) ενός URL και να τις αντικαταστήσει με τα δικά του στοιχεία επικοινωνίας προκειμένου να υποκλέψει ή να ανακατευθύνει τις κλήσεις. Αυτού του είδους η επίθεση είναι γνωστή και ως call hijacking (Si DF 2004:577) [8].

Ευπάθειες στο πρωτόκολλο SDP και στο RTP μπορούν επίσης να επηρεάσουν το τομέα της διαθεσιμότητας. Για παράδειγμα, μπορεί ένας εισβολέας να τροποποιήσει τα SDP δεδομένα ώστε οι RTP ροές δεδομένων να ανακατευθύνονται σε μία συσκευή υποκλοπής συνομιλιών. Κατόπιν τα συλλεγμένα δεδομένα RTP μπορούν να χρησιμοποιηθούν για ένα πλήθος επιθέσεων όπως η man-in-the-middle ή επίθεση ανακατεύθυνσης (redirect) ή επανάληψης (replay). Επίσης, σε μία επίθεση γνωστή ως

“RTP payload”, ο επιτιθέμενος μπορεί να στείλει πακέτα γεμάτα από τυχαία bytes τόσο στο header όσο και στο κυρίως σώμα, ώστε να δημιουργήσει προβλήματα στο jitter buffer μιας συσκευής. Αποτελέσματα αυτής της επίθεσης είναι διαλείψεις στην επικοινωνία ή κατάρρευση του συστήματος (Wu 2004:438) [7].

Άλλες επιθέσεις στο πρωτόκολλο RTP περιλαμβάνουν εισαγωγή και αλλοίωση του μηνύματος ή αλλαγή της αρίθμησης των πακέτων (Wu 2004:439) [7]. Στο RTP, τα πακέτα δεν αριθμούνται για να συναρμολογηθεί το αρχικό μήνυμα αλλά για να ειδοποιηθούν τα ανώτερα επίπεδα για πιθανή ύπαρξη προβλήματος. Από τη στιγμή λοιπόν που το RTP δεν ορίζει τη σειρά άφιξης των πακέτων, μπορεί ο επιτιθέμενος να εισάγει κακόβουλα πακέτα δίδοντας τα μία λογική αρίθμηση από πακέτα που μόλις έχει συλλέξει και έτσι να ξεκινήσει μία Denial of Service επίθεση.

Σε μία επίθεση ICMP (Ping) Flood ο επιτιθέμενος καθοδηγεί αρκετές αιτήσεις ICMP echo σε μία συσκευή (συνήθως κάνοντας broadcast σε άλλες συσκευές) (Angelo 2009:4) [13]. Οι απαντήσεις στις αιτήσεις αυτές κατευθύνονται στο θύμα τελικά καταναλώνεται όλη η επεξεργαστική ισχύς ή καταναλώνεται τόσο εύρος ζώνης που δημιουργείται μία DoS κατάσταση. Μία άλλη επίθεση γνωστή ως εξάντληση (starvation) των DHCP IP Διευθύνσεων γίνεται όταν ο επιτιθέμενος στέλνει αρκετές DHCP αιτήσεις ώστε να δεσμεύσει όλες τις διαθέσιμες διευθύνσεις IP ενός DHCP server και έτσι δεν αφήνει διαθέσιμη διεύθυνση για τις “νόμιμες” αιτήσεις (Dubrawsky 2004:12) [15].

Μία επίθεση που ομοιάζει με την SPAM είναι γνωστή ως SPIT (Spam over Internet Telephony). Σε αυτή αποστέλλονται εκατομμύρια διαφημίσεων στο voice mail πιθανών πελατών. Εξαιτίας του μεγαλύτερου όγκου από τα SPAM μηνύματα email, τα αποθηκευμένα αρχεία ήχου καταναλώνουν πολύ περισσότερους πόρους δικτύου. Στην επίθεση κλειδώματος λογαριασμού (account lockout attack), η επίθεση μπορεί να δημιουργήσει συνθήκες DoS στέλνοντας λανθασμένα διαπιστευτήρια (credentials) χρήστη για να κλειδώσει το σύστημα. Έτσι ο “νόμιμος” χρήστης δεν μπορεί να “μπει” και να χρησιμοποιήσει τη συσκευή. Για να αποφευχθεί η επίθεση βασιζόμενη σε λεξικό (dictionary attack), χρησιμοποιούνται μετρητές που καταμετρούν τις λανθασμένες απόπειρες εισαγωγής κωδικού. Ο επιτιθέμενος, απλώς, προσπαθεί να εισέλθει δίνοντας λανθασμένα στοιχεία μέχρι να κλειδωθεί ο “νόμιμος” χρήστης εκτός (Kuhn 2005:27) [3].

2.3 Εργαλεία ελέγχου ευπαθειών στο VoIP

Καθώς τα συστήματα VoIP διεισδύουν σε όλο αι περισσότερα σημεία, αναπτύσσονται συνεχώς καινούρια εργαλεία με τα οποία μπορούν οι μηχανικοί δικτύων, οι αναλυτές ασφαλείας κ.λπ. να μελετήσουν το βαθμό προστασίας των συστημάτων που αναπτύσσουν και να προλάβουν ή να διορθώσουν τυχόν ευπάθειες που μπορεί να προκύψουν.

Υπάρχει πληθώρα προγραμμάτων τόσο εμπορικά (θα συμβολίζονται με (€)) όσο και open source τα οποία ισχυρίζονται ότι είναι χρήσιμα στην ασφάλιση των συστημάτων VoIP.

Τα εργαλεία χωρίζονται σε κατηγορίες ανάλογα με το είδος των επιθέσεων που μπορούν να διεξάγουν σε (voipsa.org 2016) [16]:

- **VoIP sniffing** όπως τα:
    - **AuthTool** – εργαλείο που προσπαθεί να βρει τον κωδικό του χρήστη αναλύοντας την SIP κίνηση (traffic)
    - **Cain & Abel** – πολύ-εργαλείο με την ικανότητα επανασύνθεσης κλήσεων RTP.
    - **CommView** VoIP Analyzer (€) – προσθήκη στο πασίγνωστο CommView (της Κυπριακών συμφερόντων TamoSoft που κρατάει από την προ-Windows εποχή του DOS - 1998), η οποία μπορεί να συλλαμβάνει και να αναλύει σε πραγματικό χρόνο γεγονότα στο VoIP όπως τη ροή της κλήσης, signaling sessions, registrations, media streams, σφάλματα, κ.α.
    - **Etherpeek** (€) – γενικευμένος VoIP & Ethernet sniffer.
    - ILTY ("I'm Listening To You") - Open-source (python), πολυκάναλος (multi-channel) SKINNY sniffer.
    - **NetDude** – Ένα framework για την επιθεώρηση (inspection), την ανάλυση and και τον χειρισμό (manipulation) των tcpdump αρχείων καταγραφής.
    - **Oreka** – Αρθρωτή (modular) cross-platform εφαρμογή καταγραφής και ανάκτησης των audio streams.

- **PSIPDump** – εργαλείο καταγραφής των συνεδριών SIP και του RTP, αν είναι διαθέσιμο, παρόμοιο με το "tcpdump -w".
- **rtpBreak** – Ανιχνεύει, ανασυνθέτει και αναλύει οποιαδήποτε RTP συνεδρία που γίνεται σε πρωτόκολλο UDP χρησιμοποιώντας ευρετική (heuristic) μέθοδο. Λειτουργεί καλά με SIP, H.323, SCCP και οποιοδήποτε άλλο πρωτόκολλο σηματοδοσίας. Δεν απαιτεί πακέτα RTCP.
- **SIPomatic** - SIP listener που είναι τμήμα του Linphone
- **SIPv6 Analyzer** – Αναλυτής SIP και IPv6.
- **UCSniff** – εργαλείο αξιολόγησης που επιτρέπει το γρήγορο έλεγχο απειλών υποκλοπών VoIP. Υποστηρίζει SIP και Skinny σηματοδοσία, G.711-ulaw και G.722 codecs, καθώς και μία λειτουργία MITM ARP Poisoning.
- **VoiPong** - Βοηθητικό πρόγραμμα το οποίο ανιχνεύει όλες τις Voice Over IP κλήσεις, και για όσες έχουν κωδικοποίηση G711, καταγράφει τη πραγματική συζήτηση σε ξεχωριστά αρχεία ήχου (wav). Υποστηρίζει SIP, H323, Skinny, RTP και RTCP.
- **VoIPong** ISO Bootable – Εκκινήσιμη (bootable) "Live-CD" έκδοση του VoIPong.
- **VOMIT** – Utility που μετατρέπει μία τηλεφωνική συνομιλία από Cisco IP τηλέφωνο σε αρχείο ήχου.
- **Wireshark** – Γνωστό παλαιότερα ως Ethereal, είναι το κορυφαίο πρόγραμμα ανάλυσης της κίνησης ενός δικτύου.
- **WIST** - Web Interface for SIP Trace - ένα PHP Web Interface που επιτρέπει να συνδεθεί κάποιος σε απομακρυσμένο υπολογιστή/θύρα και να συλλάβει/φιλτράρει μία SIP συνεδρία.

- **VoIP Scanning and Enumeration** όπως τα κάτωθι:
  - **EnableSecurity VoIPPack for CANVAS (€) -** ένα πλήθος εργαλείων σχεδιασμένο να λειτουργεί με τη πλατφόρμα Immunity CANVAS. Τα εργαλεία μπορούν να σκανάρουν, να συλλέξουν πληροφορίες (enumeration), και να ανιχνεύσουν κωδικούς.
  - **enumIAX –** Login enumerator για το IAX2 (Asterisk) πρωτόκολλο.
  - **iaxscan –** Scanner γραμμένος σε Python ανίχνευσης IAX/2 hosts από τους οποίους συλλέγει πληροφορίες χρηστών με τη χρήση bruteforce τεχνικών.

- **iWar -** Wardialer σε IAX2 protocol
- **Nessus –** Ο παλαιότερος δωρεάν ανιχνευτής ευπαθειών.
- **nmap -** Ο παλαιότερος open source, port scanner δικτύων.
- **Passive Vulnerability Scanner (€) –** Παθητικός σαρωτής ευπάθειας που μπορεί να ανακαλύψει τι συμβαίνει στο δίκτυό χωρίς ενεργή σάρωση. Ανιχνεύει το πραγματικό πρωτόκολλο, διάφορες συνδέσεις διαχειριστών και σαρωτές VoIP. Επί του παρόντος, περιλαμβάνει πάνω από 40 ελέγχους VoIP.
- **SCTPScan –** Εργαλείο που συλλέγει πληροφορίες ανοιχτών SCTP θυρών χωρίς να εγκαθιστά πλήρη SCTP συσχέτιση με τον απομακρυσμένο host. Επίσης μπορεί να σκανάρει ολόκληρο δίκτυο για να βρει συσκευές που επικοινωνούν με SCTP.
- **SIP Forum Test Framework (SFTF)** – Δημιουργία του SIP Forum που δημιουργήθηκε για να ελέγχουν οι κατασκευαστές τις SIP συσκευές τους για σφάλματα.
- **SIP-Scan –** Πολύ γρήγορος ανιχνευτής δικτύων SIP.
- **SIPcrack –** Cracker της login διαδικασίας του SIP. Αποτελείται από δύο προγράμματα, το SIPdump που ανιχνεύει τα logins σε ένα δίκτυο και το SIPcrack με το οποίο μέσω bruteforce ψάχνει για passwords.
- **Sipflanker –** Ανιχνεύει πιθανώς ευάλωτα Web GUIs σε ένα δίκτυο.
- **SIPSCAN -** SIP enumerator των usernames που χρησιμοποιεί μεθόδους INVITE, REGISTER και OPTIONS.
- **SIPVicious Tool Suite –** Σουίτα με τα svmap, svwar, svcrack. Το svmap είναι ανιχνευτής SIP. Εμφανίζει όλες τις SIP συσκευές μέσα σε ένα δοθέν IP range. Το svwar προσδιορίζει τις ενεργές επεκτάσεις σε ένα PBX τηλεφωνικό κέντρο. Το svcrack είναι ένας online password cracker για τα SIP PBX.
- **SiVuS –** Ανιχνευτής ευπαθειών στο SIP.
- **SMAP -** SIP Stack Fingerprinting Scanner
- **VLANping** – Pinging utility που λειτουργεί σε VLAN.
- **Viproy** - VoIP και Exploitation Kit. Παρέχει προσθήκες (modules) για το metasploit για Penetration Testing χρησιμοποιώντας βιβλιοθήκες Skinny, SIP και MSRP.

- **Viproxy** (MITM Proxy and Testing Tool) - Αυτόνομο module για το Metasploit που επιτρέπει την παρακολούθηση της TCP/TLS κυκλοφορίας και την εκτέλεση κάποιων επιθέσεων σε thick clients εφαρμογές, mobile εφαρμογές και πελάτες VoIP. (v3)
- **VoIPAudit (€) –** Ειδικευμένος ανιχνευτής ευπαθειών VoIP.

- **VoIP Packet Creation and Flooding** με χαρακτηριστικούς εκπροσώπους τα:
  - **IAXFlooder** – Υπερχειλιστής πακέτων (packet flooder) που δημιουργεί πακέτα IAX.
  - **INVITE Flooder –** Αποστέλλει καταιγιστικά μηνύματα SIP INVITE σε ένα τηλέφωνο ή proxy.
  - **iThinkTest FlowCoder: SiPBlast (€) –** Δοκιμάζει τις ικανότητες των υποδομών να διαχειριστούν επιθέσεις υπερχείλισης/χωρητικότητας SIP εξομοιώνοντας δημιουργώντας μαζικές κλήσεις CPE.
  - **kphone-ddos –** Χρησιμοποιεί το KPhone για επιθέσεις υπερχείλισης πλαστογραφημένων (spoofed) πακέτων SIP.
  - **NSAUDITOR - SIP UDP Traffic Generator - Flooder (€) -** Γεννήτρια κυκλοφορίας/υπερχείλισης SIP UDP για εξαντλητική δοκιμή συστημάτων VoIP, προγραμμάτων SIP και εφαρμογών κάτω από μεγάλο φόρτο του δικτύου. Είναι ένα πολύ απλό και γρήγορο πρόγραμμα που μπορεί να προσομοιώσει πελάτη SIP όπως και τη δραστηριότητα κλήσης.
  - **RTP Flooder –** Δημιουργεί "καλοσχηματισμένα" πακέτα RTP που μπορούν να υπερχειλίσουν ένα τηλέφωνο ή ένα proxy.
  - **Scapy –** Το Scapy είναι ένα ισχυρό διαδραστικό πρόγραμμα χειραγώγησης πακέτων. Μπορεί να χειριστεί εύκολα πιο κλασικές ενέργειες, όπως η σάρωση, το tracerouting, το probing, τον έλεγχο μονάδων, και επιθέσεις σε ένα δίκτυο.
  - **Seagull** - μια γεννήτρια κίνησης πολλαπλών πρωτοκόλλων με ιδιαίτερη στόχευση στο IMS.
  - **SIPBomber –** Ελεγκτής πρωτοκόλλου SIP για Linux.
  - **SIPNess –** Ελέγχει εφαρμογές SIP.
  - **SIPp –** δωρεάν open source εργαλείο ελέγχου και δημιουργίας κίνησης για το πρωτόκολλο SIP.
  - **SIPsak –** Πολυεργαλείο για το SIP.

- **VoIP Fuzzing**
    - **Asteroid** - Ένα σύνολο παραποιημένων SIP μεθόδων (INVITE, CANCEL, BYE, κ.λπ.) που μπορεί να σταλεί σε οποιοδήποτε τηλέφωνο ή διακομιστή μεσολάβησης proxy.
    - **Fuzzy Packet -** Εργαλείο για το χειρισμό μηνυμάτων μέσω εισαγωγής, σύλληψης, λήψης ή αποστολή πακέτων που δημιουργούνται σε ένα δίκτυο.
    - **Interstate Fuzzer -** VoIP Fuzzer
    - **Mu Dynamics VoIP, IPTV, IMS Fuzzing Platform(€) -** Fuzzing εφαρμογή για πρωτόκολλα SIP, Diameter, H.323 και MGCP.
    - **ohrwurm –** απλό και εύχρηστο RTP fuzzer.
    - **PROTOS H.323 Fuzzer -** ένα εργαλείο Java που στέλνει ένα σύνολο ακατάλληλων μηνυμάτων H.323. Σχεδιάστηκε από το Πανεπιστήμιο Oulu στη Φινλανδία.
    - **PROTOS SIP Fuzzer -** ένα εργαλείο Java που στέλνει ένα σύνολο ακατάλληλων μηνυμάτων SIP. Σχεδιάστηκε από το Πανεπιστήμιο Oulu στη Φινλανδία.
    - **Sip-Proxy -** Ενεργεί ως ενδιάμεσος μεταξύ ενός VoIP UA και ενός VoIP PBX. Τα μηνύματα SIP που ανταλλάσσονται περνάνε μέσα από την εφαρμογή και μπορούν να καταγραφούν, να χειραγωγηθούν, ή να σταλούν με fuzzy τεχνική.
    - **Spirent ThreatEx (€) -** εμπορικό fuzzer πρωτόκολλο και ελεγκτής αντοχής-στιβαρότητας (robustness).
    - **VoIPER -** ένα σύνολο εργαλείων ασφαλείας που έχει ως στόχο να επιτρέψει στους προγραμματιστές και ερευνητές ασφαλείας την εύκολη, εκτενή και με αυτόματο τρόπο τον έλεγχο συσκευών VoIP για ευπάθειες ασφάλειας.

- **VoIP Signaling Manipulation**
    - **BYE Teardown -** Αυτό το εργαλείο προσπαθεί να αποσυνδέσει μια ενεργή συνομιλία VoIP πλαστογραφώντας του μηνύματος SIP BYE από τον παραλήπτη.

- **Check Sync Phone Rebooter -** Μεταδίδει ένα ειδικό NOTIFY SIP μήνυμα το οποίο θα επανεκκινήσει ορισμένα τηλέφωνα.
- **H225regregject -** εργαλείο που χρησιμοποιείται για την αποσύνδεση κλήσεων H.323. Παρακολουθεί πρώτα το δίκτυο, προκειμένου να καθορίσει εάν μια κλήση λαμβάνει χώρα. Μόλις εντοπιστεί μια κλήση, τότε εγχέει ένα πακέτο Εγγραφής Απόρριψης (Registration Reject) στην κλήση.
- **IAXAuthJack** - χρησιμοποιείται για να εκτελέσει ενεργά μια επίθεση υποβάθμισης ταυτότητας και να αναγκάσει ένα τελικό σημείο να αποκαλύψει τον κωδικό του σε απλό κείμενο μέσω του δικτύου.
- **IAXHangup -** εργαλείο που χρησιμοποιείται για να αποσυνδέει κλήσεις IAX. Παρακολουθεί πρώτα το δίκτυο, προκειμένου να καθορίσει εάν γίνεται μια κλήση. Μόλις εντοπιστεί η κλήση, εγχέει ένα HANGUP πλαίσιο.
- **iThinkTest FlowCoder: SiPCPE (€)** - Αξιολογούν SIP υποδομές παρεμβάλλοντας μηνύματα SIP.
- **RedirectPoison** - το εργαλείο αυτό λειτουργεί σε περιβάλλον σηματοδόσίας SIP όπου παρακολουθεί για αιτήματα INVITE και ανταποκρίνεται με απάντηση ανακατεύθυνσης, προκαλώντας το σύστημα να εκδώσει νέο INVITE σε άλλη τοποθεσία.
- **Registration Adder** – Η εφαρμογή προσπαθεί να συνδέσει μια άλλη διεύθυνση SIP με το στόχο, δημιουργώντας ουσιαστικά ένα call ring σε δύο μέρη (του νόμιμο χρήστη και του επιτιθέμενου)
- **Registration Eraser -** Εργαλείο που προκαλεί DoS, στέλνοντας ένα πλαστογραφημένο SIP REGISTER μήνυμα για να πείσει τον proxy ότι ένα τηλέφωνο / χρήστης δεν είναι διαθέσιμος.
- **Registration Hijacker -** αυτό το εργαλείο προσπαθεί να πλαστογραφήσει τα SIP REGISTER μηνύματα ώστε να δρομολογήσει όλες τις εισερχόμενες κλήσεις προς τον επιτιθέμενο.
- **SIP-Kill -** Sniffing των SIP-INVITEs με σκοπό να τερματίσει την κλήση.
- **SIP-Proxy-Kill –** Τερματίζει ένα SIP-Session στον τελευταίο proxy πριν αυτό καταφέρει να φτάσει στον τελικό προορισμό του.
- **SIP-RedirectRTP -** Χειρίζεται SDP επικεφαλίδες, έτσι ώστε τα πακέτα RTP να ανακατευθύνονται σε ένα RTP-proxy.

- **SipRogue -** ένας πολυλειτουργικός SIP proxy που μπορεί να εισαχθεί μεταξύ δύο συνομιλούντων.
- **vnak -** VoIP Network Attack Toolkit - συνδυασμός μιας σειράς από επιθέσεις εναντίον πολλαπλών πρωτοκόλλων σε ένα εύκολο στη χρήση εργαλείο. Στόχος του είναι να είναι το μοναδικό εργαλείο που χρειάζεται ένας χρήστης για να επιτεθεί σε πολλαπλά πρωτόκολλα VoIP.
- **VoIPHopper** - εργαλείο επικύρωσης ασφαλείας που ελέγχει αν ένας υπολογιστής μπορεί να μιμηθεί τη συμπεριφορά ενός τηλεφώνου IP. Αυτοματοποιεί γρήγορα ένα VLAN Hop σε VLAN φωνής.

- **VoIP Media Manipulation**
  - **RTP InsertSound** - παίρνει τα περιεχόμενα ενός αρχείου ήχου. wav ή ενός αρχείου καταγραφής του tcpdump και **εισάγει** τον ήχο σε μια ενεργή συνομιλία.
  - **RTP MixSound** - παίρνει τα περιεχόμενα ενός αρχείου ήχου. wav ή ενός αρχείου καταγραφής του tcpdump και **αναμιγνύει** τον ήχο σε μια ενεργή συνομιλία.
  - **RTPInject -** εγχέει αυθαίρετο ήχο σε εγκατεστημένες συνδέσεις RTP. Το εργαλείο εντοπίζει ενεργές συζητήσεις, αναγνωρίζει τον codec της συνομιλίας και επιτρέπει την έγχυση ενός αυθαίρετου αρχείου ήχου.
  - **RTPProxy** – Περιμένει για εισερχόμενα πακέτα RTP και τα στέλνει σε άλλη διαδρομή (που σηματοδοτείται από ένα ελαφρύ πρωτόκολλο).
  - **SteganRTP** - ένα εργαλείο στεγανογραφίας το οποία δημιουργεί ένα αμφίδρομο στεγανογραφικό πρωτόκολλο μεταφοράς δεδομένων χρησιμοποιώντας πακέτα Real-time Tranfer Protocol (RTP) ως μέσο κάλυψης. Το εργαλείο παρέχει chat, μεταφορά αρχείων, και απομακρυσμένο shell.
  - **Vo²IP** - Με το Vo²IP, μπορούμε να δημιουργήσουμε μια κρυφή συνομιλία ενσωματώνοντας περαιτέρω συμπιεσμένα δεδομένα φωνής σε κανονική PCM-based κυκλοφορία φωνής (δηλαδή G.711 codec).

- **Διάφορα άλλα εργαλεία**
  - **IAX.Brute** - παθητικό εργαλείο για επιθέσεις λεξικού (dictionary attacks) στην μέθοδο ελέγχου ταυτότητας πρόκλησης/απόκρισης (challenge/

response authentication) του IAX. Η επίθεση αυτή επιτρέπει στους κακόβουλους χρήστες να κλέψουν τους κωδικούς πρόσβασης και τις ταυτότητες των συνομιλούντων.

- **SIP-Send-Fun** – μικρό script γραμμής εντολών το οποίο εκμεταλλεύεται συγκεκριμένες ευπάθειες.
- **SIP Tastic** - παθητικό εργαλείο επιθέσεων λεξικού στην μέθοδο ελέγχου ταυτότητας του SIP. Η επίθεση αυτή επιτρέπει στους κακόβουλους χρήστες να κλέψουν τους κωδικούς πρόσβασης και τις ταυτότητες των συνομιλούντων.
- **Spitter** - Ένα σύνολο εργαλείων για Asterisk για εκτέλεση δοκιμών spam στο VoIP.
- **VoIP Security Audit Program (VSAP)** - αυτοματοποιημένο εργαλείο ερωταπαντήσεων για τον έλεγχο της ασφάλειας των δικτύων VoIP (SIP / H.323 / RTP). Παρέχει θέματα ασφάλειας και ερωτήματα ελέγχου για τον τελικό χρήστη. Μόλις όλα τα ερωτήματα απαντηθούν, το VSAP παρέχει την τελική βαθμολογία.
- **XTest** - Ένα απλό, πρακτικό και δωρεάν, εργαλείο ελέγχου ασφαλείας ενσύρματου 802.1x που εφαρμόζει την RFC 3847 EAP-MD5 authentication μέθοδο.

Κεφάλαιο 3
Linphone - Πρωτόκολλα & ροή δεδομένων

Το Linphone είναι η πρώτη open-source GNU/Linux εφαρμογή που χρησιμοποίησε το SIP. Ξεκίνησε το 2001 από τον Simon Morlat και είναι διαθέσιμο σε εκδόσεις τόσο για κινητές και embedded συσκευές (iOS, Android, Windows Phone, BlackBerry, Linux/arm, Linux/blackfin) όσο και για υπολογιστές (GNU/Linux, Windows Desktop, MAC OSX) καθώς και για web browsers.
Η παρούσα διατριβή εστιάζει στο Linphone λόγω του γεγονότος ότι, μέχρι σήμερα, δεν έχει μελετηθεί σε μεγάλο βαθμό ως προς τα χαρακτηριστικά ασφαλείας του.

3.1 Linphone - Επισκόπηση

Η κατασκευή του Linphone διαχωρίζει τη διεπαφή χρήστη (user interface) από το πυρήνα (core), γεγονός που επιτρέπει τη δημιουργία ανεξάρτητων user interfaces διατηρώντας όλες του τις λειτουργίες. Αναλυτικότερα περιλαμβάνει τα κάτωθι frontends user interface:

- Gtk+ interface για windows, mac και GNU/Linux
- Interface κονσόλας (Linphonec, Linphonecsh)
- Εφαρμογή iPhone application γραμμένη σε objective C
- Εφαρμογή Android γραμμένη σε java
- Εφαρμογή Windows Phone γραμμένη σε C#

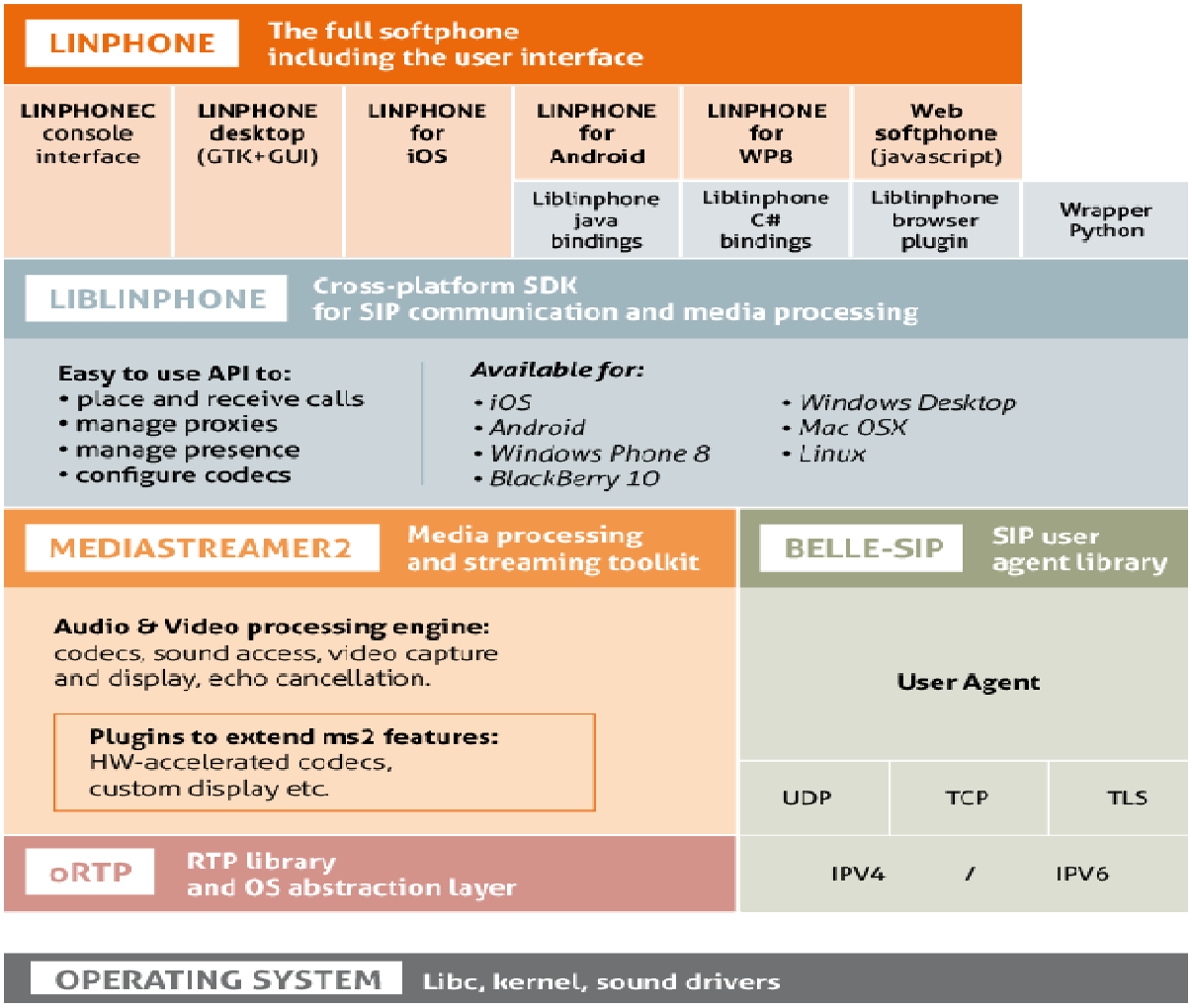
Ο πυρήνας, το **Lib**Linphone, υλοποιεί όλες τις λειτουργίες του Linphone. Θεωρείται αρκετά ισχυρό SIP VoIP video SDK που μπορεί να χρησιμοποιηθεί για να προσθέσει δυνατότητες ήχου ή video-κλήσης σε μια εφαρμογή. Παρέχει API με το οποίο μπορεί να ελεγχθεί η έναρξη, η λήψη και ο τερματισμός κλήσεων ήχου & βίντεο. Η βιβλιοθήκη

LibLinphone είναι εξ' ολοκλήρου γραμμένη σε C και βασίζεται στις κάτωθι επιμέρους βιβλιοθήκες:

- Mediastreamer2, ένα multimedia SDK που υλοποιεί και διαχειρίζεται audio/video streaming.
- oRTP, μία RTP βιβλιοθήκη.
- belle-sip, η βιβλιοθήκη SIP.

Υπάρχει επίσης ένα interface γραμμής εντολών, το Linphonec που μπορεί να χρησιμοποιεί τη readline βιβλιοθήκη στο GNU/Linux όπως το bash για να συλλέγει την ολοκλήρωση της επικοινωνίας και το ιστορικό της.

Το Linphonecsh είναι άλλο ένα εργαλείο γραμμής εντολών για να ελέγχει ένα Linphonec daemon εξ' αποστάσεως. Σε αντίθεση με το Linphonec, το Linphonecsh τερματίζει αμέσως μόλις εκτελεστεί η εντολή. Η αρχιτεκτονική του Linphone φαίνεται στον παρακάτω πίνακα:



Πίνακας 2. Αρχιτεκτονική του Linphone.
Πηγή: http://www.Linphone.org/technical-corner/Linphone/overview

3.2 Το πρωτόκολλο SIP

Το Session Initiation Protocol (SIP) καθορίστηκε αρχικά στο RFC 2543 και αργότερα στο RFC 3261, από την ομάδα εργασίας MMUSIC (Multiparty Multimedia Session Control) της IETF (Internet Engineering Task Force) και είναι ένα πρωτόκολλο σηματοδοσίας υπεύθυνο για την αρχικοποίηση, την διαχείριση και τον τερματισμό συνόδων, μέσω του Διαδικτύου, όπως για παράδειγμα η αρχικοποίηση, ο έλεγχος και ο τερματισμός VoIP κλήσεων. Η έννοια της συνόδου εισήχθη αρχικά στο RFC 2327 (Session Description Protocol) ως ένα σύνολο από data streams που περιέχουν πολλαπλούς τύπους δεδομένων media μεταξύ αποστολέων και δεκτών. Μια σύνοδος μπορεί να είναι ένα τηλεφώνημα, μια τηλεδιάσκεψη, ο διαμοιρασμός δεδομένων μεταξύ δύο χρηστών, το chatting, η ανταλλαγή instant messaging (Internet Society 2002:5) [17].

Το SIP είναι εμπνευσμένο από το HTTP και το SMTP πρωτόκολλο, χρησιμοποιεί τόσο το UDP όσο και το TCP και τα μηνύματα που ανταλλάσσει μπορούμε να τα συγκρίνουμε με αυτά των CB's. Είναι text-based πρωτόκολλο βασισμένο στο μοντέλο πελατών-εξυπηρετητών (client-server model), που παρέχει τις ακόλουθες βασικές απαιτήσεις των σημερινών μέσων επικοινωνιών (sipforum.org 2003:3) [18]:

- User Location: Προσδιορισμός παραμέτρων όπως διεύθυνση IP και port number, οι οποίες απαιτούνται για την επαφή με τον τελικό χρήστη.
- User Availability: Προσδιορισμός της ικανότητας επίτευξης επικοινωνίας με έναν τελικό χρήστη.
- Endpoint Capabilities: Προσδιορισμός των media δυνατοτήτων (π.χ. codecs) των τελικών χρηστών.
- Session Setup: Εγκατάσταση συνόδων επικοινωνίας (sessions) μεταξύ endpoint users.
- Session Management: Διαχείριση των συνόδων επικοινωνίας.

Ένα σημαντικό χαρακτηριστικό γνώρισμα του πρωτοκόλλου SIP είναι ότι δεν είναι σε θέση να καθορίζει τον τύπο της επικοινωνίας που εγκαθίσταται, αλλά μόνο να την διαχειρίζεται. Αυτό έχει ως συνέπεια να μπορεί να χρησιμοποιηθεί σε έναν τεράστιο αριθμό εφαρμογών και υπηρεσιών, όπως interactive games, μουσική, φωνή και βίντεο.

Μερικά χαρακτηριστικά του SIP, που το κάνουν να ξεχωρίζει από τα υπόλοιπα πρωτόκολλα σηματοδοσίας είναι τα εξής (telecomspace.com 2016:1) [19]:

- Τα μηνύματα SIP είναι text-based και ως εκ τούτου είναι εύκολο να διαβαστούν και να διορθωθούν. Επομένως, η υλοποίηση νέων υπηρεσιών γίνεται πιο εύκολη για τους σχεδιαστές-προγραμματιστές.
- Επαναχρησιμοποιεί περιγραφή τύπου MIME με τον ίδιο τρόπο όπως και οι email clients και έτσι οι εφαρμογές που σχετίζονται με συνεδρίες μπορούν να εκκινήσουν αυτομάτως
- Το SIP επαναχρησιμοποιεί διάφορες υπάρχουσες και δοκιμασμένες υπηρεσίες και πρωτόκολλα του Διαδικτύου όπως DNS, RTP, RSVP κ.λπ.
- Μπορούν εύκολα να ορισθούν νέα SIP extensions, επιτρέποντας στους οργανισμούς παροχής VoIP επικοινωνιών να προσθέτουν στα προϊόντα τους νέες υπηρεσίες χωρίς το φόβο ζημιάς στο υπάρχον δίκτυο τους.
- Το πρωτόκολλο SIP είναι ανεξάρτητο από το στρώμα μεταφοράς του δικτύου IP. Μάλιστα, χρησιμοποιεί είτε το User Datagram Protocol (UDP) ή το Transmission Control Protocol (TCP).
- Υποστηρίζει διαπραγμάτευση multi-device. Εάν μια υπηρεσία ή συνεδρία ξεκινά βίντεο και φωνή, η φωνή μπορεί ακόμα να μεταδοθεί σε συσκευές χωρίς ενεργοποιημένο το βίντεο, ή μπορούν να χρησιμοποιηθούν άλλα χαρακτηριστικά της συσκευής όπως μονόδρομο video streaming.

3.2.1 SIP Elements

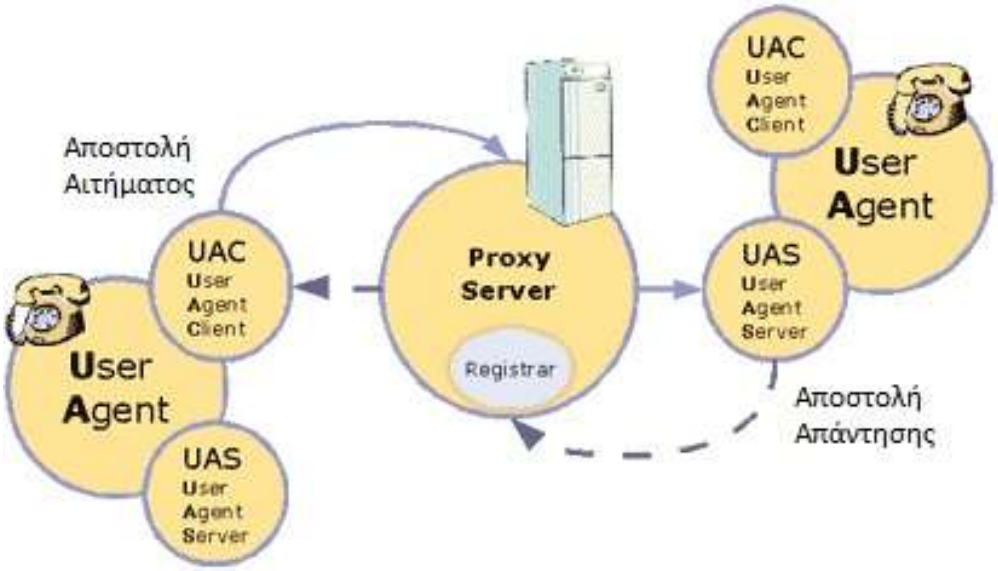
Το SIP αποτελείται από ένα σύνολο στοιχείων που αλληλοεπιδρούν μεταξύ τους. Τα βασικότερα είναι:

- User Agents (UAs)
- Proxy servers
- Registar server
- Redirect server
- Location server

Ο **User Agent** είναι ένα από τα σημαντικότερα στοιχεία σε ένα δίκτυο SIP και θεωρείται η πλέον “έξυπνη” συσκευή σε αυτό. Μπορεί να είναι ένα softphone, ένα κινητό τηλέφωνο, ένα λάπτοπ κ.λπ. Ο UA Μπορεί να αρχίσει, να μεταβάλλει και να τερματίσει

μια συνεδρία χωρίς να απαιτείται η χρήση αφοσιωμένου (dedicated) server. Αυτό στηρίζεται στο χωρισμό του SIP σε δύο μέρη: στον user-agent-server (UAS) και στον user-agent client (UAC) (tutorialspoint.com 2016:1) [20].

Ο UA **Client αποστέλλει** μία αίτηση και **λαμβάνει** μία απάντηση.
Ο UA **Server λαμβάνει** μία αίτηση και **αποστέλλει** μία απάντηση.



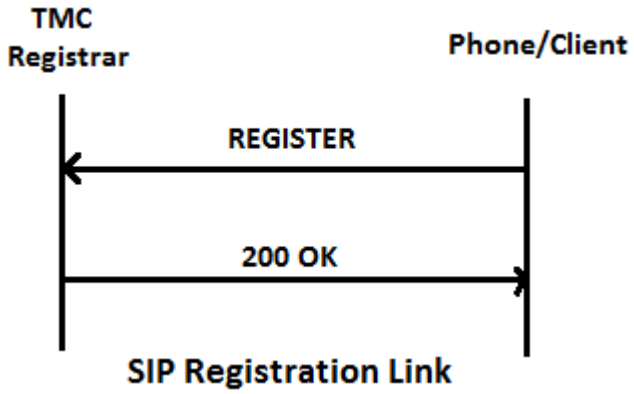
Σχήμα 1: Διάγραμμα βασικής λειτουργίας Client/Server στο πρωτόκολλο SIP.
Πηγή: http://voip.html.it/articolo06.aspx

Το πρωτόκολλο SIP βασίζεται στην αρχιτεκτονική client-server όπου η συσκευή του καλούντος λειτουργεί ως client και ξεκινά τη κλήση και η συσκευή του καλούμενου λειτουργεί ως server και απαντάει στη κλήση (tutorialspoint.com 2016:2) [20].

Ο **Proxy Server** είναι το στοιχείο εκείνο του δικτύου το οποίο παίρνει μία αίτηση από ένα UA και τη προωθεί σε άλλο χρήστη. Βρίσκεται ανάμεσα σε δύο UA's και η λειτουργία του έχει πολλές ομοιότητες με εκείνες ενός router. Διαθέτει αρκετή "ευφυΐα" για να καταλάβει μια αίτηση SIP και να την προωθήσει με τη βοήθεια του URI.
Υπάρχουν δύο τύποι proxy servers:

- **Stateless Proxy Server**, ο οποίος απλώς προωθεί το μήνυμα που έλαβε χωρίς να αποθηκεύσει καμία πληροφορία σχετική με τη κλήση ή το μήνυμα και
- ο **Stateful Proxy Server**, ο οποίος παρακολουθεί κάθε αίτηση και απάντηση που μεταδόθηκε και μπορεί να τις χρησιμοποιήσει στο μέλλον, εάν απαιτείται. Μπορεί να αναμεταδώσει το αίτημα, αν δεν υπάρξει ανταπόκριση από την άλλη πλευρά.

Ο **Registar Server** δέχεται αιτήσεις εγγραφής (registrations) από τους UA's. Βοηθά τους χρήστες σχετικά με τον έλεγχο ταυτότητας τους (authentication) μέσα στο δίκτυο. Αποθηκεύει το URI και τη θέση των χρηστών σε μια βάση δεδομένων για να βοηθήσει άλλους SIP servers μέσα στο ίδιο domain.



Σχήμα 2: Τυπική SIP καταχώριση (registration).

Πηγή: https://www.tutorialspoint.com/session_initiation_protocol/session_initiation_protocol_network_elements.htm

Στο σχ. 2 ο καλλών θέλει να κάνει εγγραφή (registration) στο domain TMC. Έτσι, στέλνει ένα αίτημα REGISTER στον Registar server του TMC και ο server του επιστρέφει μία απάντηση 200 OK, δεδομένου ότι έκανε δεκτή την αίτηση του πελάτη (tutorialspoint.com 2016:3) [20].

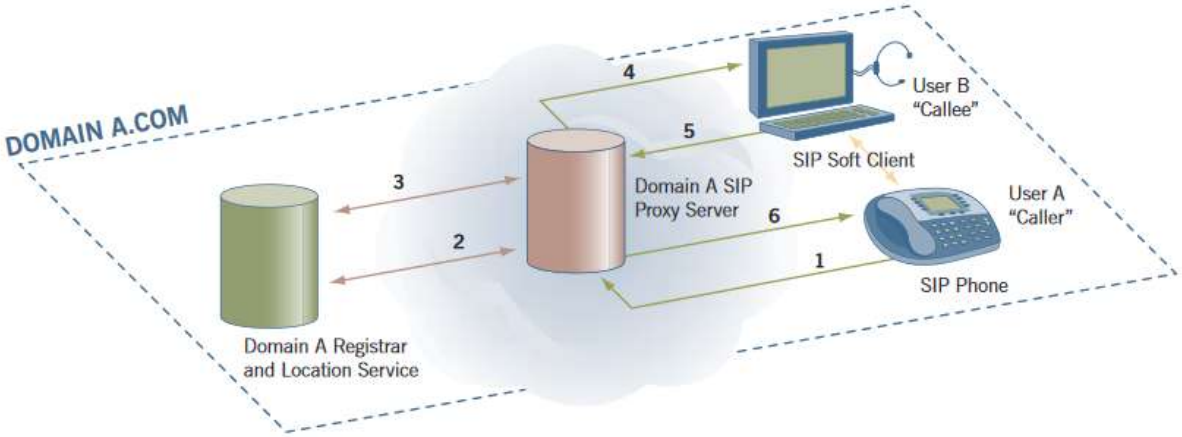
Ο **Redirect Server** δέχεται αιτήματα και αναζητά τον αποδέκτη της αίτησης στη βάση δεδομένων που δημιούργησε ο Registar server. Τη βάση αυτή την χρησιμοποιεί για να πάρει πληροφορίες για την τοποθεσία και απαντά με 3xx (Redirect response) στον χρήστη.

Ο **Location Server** παρέχει πληροφορίες σχετικά με τις πιθανές θέσεις του καλούντος στους Redirect και Proxy servers. Μόνο ένας Proxy server ή ένας Redirect server μπορεί να επικοινωνήσει με ένα Location Server.

Το σχήμα 3 απεικονίζει τους ρόλους που διαδραματίζουν οι κάθε ένα από τα στοιχεία του δικτύου για τη δημιουργία μιας συνεδρίας.

Σχήμα 3: Τα στοιχεία που απαρτίζουν μία SIP συνεδρία. Πηγή: https://www.tutorialspoint.com/session_initiation_protocol/session_initiation_protocol_network_elements.htm

Στα δύο σχήματα που ακολουθούν φαίνεται η εγκαθίδρυση της επικοινωνίας όταν οι χρήστες είναι στον ίδιο SIP server (σχ. 4) και όταν είναι σε διαφορετικό (σχ. 5) (sipforum.org 2003:4) [18]



1. Call User B
2. Query "Where is User B?"
3. Response "User B SIP Address"
4. 'Proxied' Call
5. Response
6. Response
7. Multimedia Chanel Establised

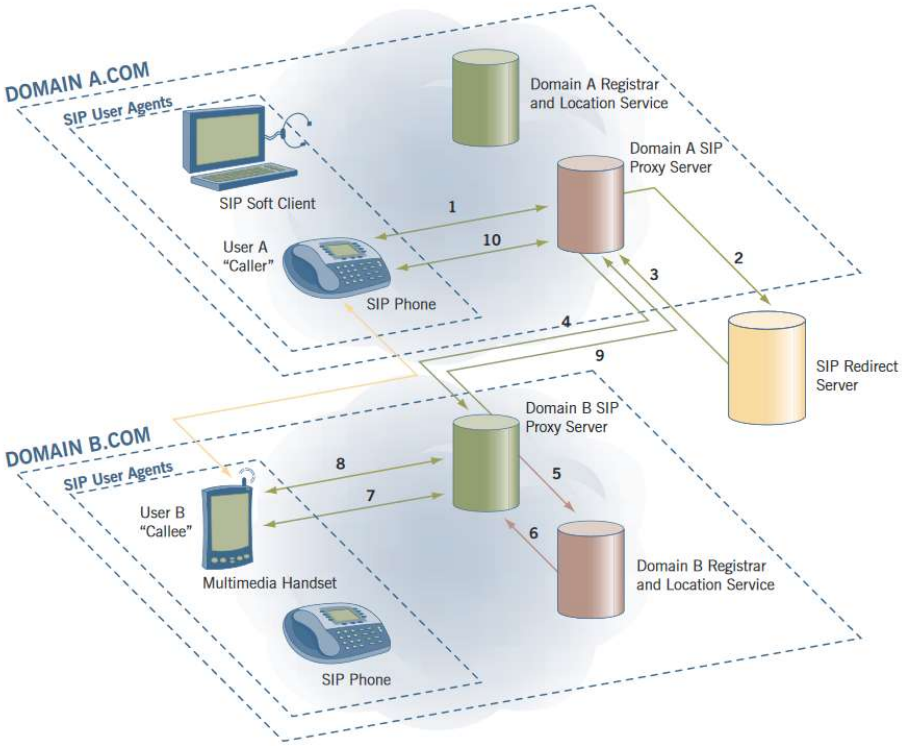
Non-SIP Queried (i.e. Database Lookup)

SIP Signaling

RTP

Σχήμα 4. Συνεδρία SIP όπου και οι δύο χρήστες είναι στον ίδιο SIP server

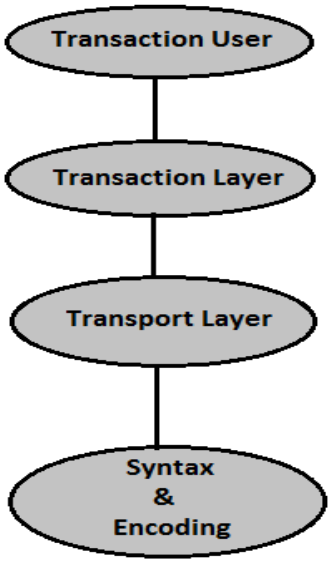
1. Call User B
2. Query “How do I get to User B, Domain B?”
3. Response “Address of Proxy Controller for Domain”
4. Call ‘Proxied’ to SIP Proxy for Domain B
5. Query “Where is User B?”
6. User B’s Address
7. Proxied Call
8. Response
9. Response
10. Response
11. Multimedia Channel Established



Σχήμα 5. Συνεδρία SIP όπου ο κάθε χρήστης είναι σε διαφορετικό SIP server

3.2.2 SIP – Αρχιτεκτονική Συστήματος

Το πρωτόκολλο SIP έχει διαστρωματωμένη αρχιτεκτονική, γεγονός που φανερώνει ότι η συμπεριφορά του περιγράφεται με όρους ανεξάρτητων σταδίων επεξεργασίας έχοντας μόνο μία “χαλαρή” σύνδεση μεταξύ αυτών των σταδίων-επιπέδων.



Σχήμα 6: Η SIP διαστρωματωμένη αρχιτεκτονική.

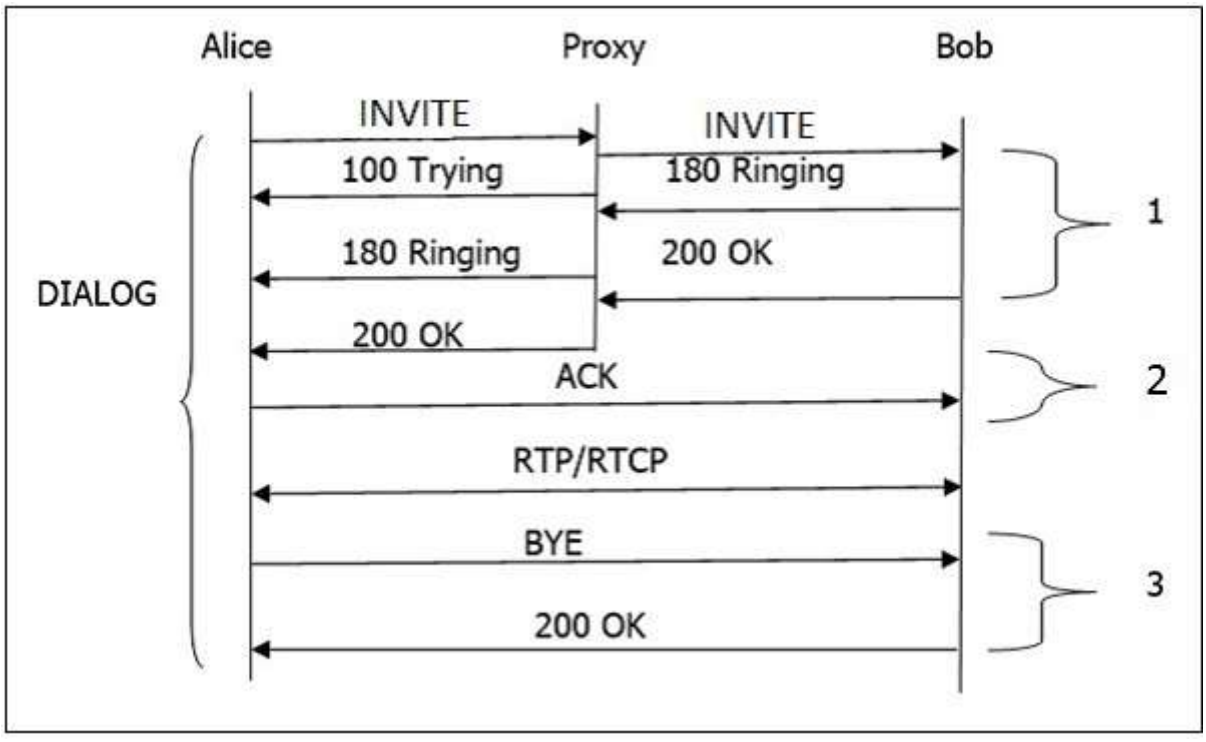
Πηγή: https://www.tutorialspoint.com/session_initiation_protocol/session_initiation_protocol_network_elements.htm

Το χαμηλότερο στρώμα SIP είναι η σύνταξη και κωδικοποίηση του. Η κωδικοποίηση του καθορίζεται με τη χρήση μιας επαυξημένης γραμματικής Backus-Naur Form (BNF).

Το δεύτερο επίπεδο είναι το επίπεδο μεταφοράς. Καθορίζει πώς ένας πελάτης στέλνει αιτήσεις και λαμβάνει απαντήσεις και πώς ένας διακομιστής δέχεται αιτήματα και στέλνει απαντήσεις μέσω του δικτύου. Όλα τα δομικά στοιχεία του SIP περιλαμβάνουν ένα στρώμα μεταφοράς.

Ακολουθεί το επίπεδο συναλλαγής. Μια συναλλαγή (transaction) είναι ένα αίτημα που αποστέλλονται από ένα Client (χρησιμοποιώντας το στρώμα μεταφοράς) σε ένα Server, μαζί με όλες τις απαντήσεις στο αίτημα αυτό από τον server πίσω στον πελάτη. Κάθε διεργασία που ολοκληρώνει ένας client UAC, πραγματοποιείται χρησιμοποιώντας μια ακολουθία συναλλαγών. Οι Stateless proxies δεν περιέχουν ένα επίπεδο συναλλαγής.
Το στρώμα πάνω από το transaction layer ονομάζεται χρήστης συναλλαγής. Κάθε μία από τις οντότητες του SIP, εκτός από τους Stateless proxies, είναι και ένας χρήστης συναλλαγής.

3.2.3 Βασική ροή κλήσεων στο SIP

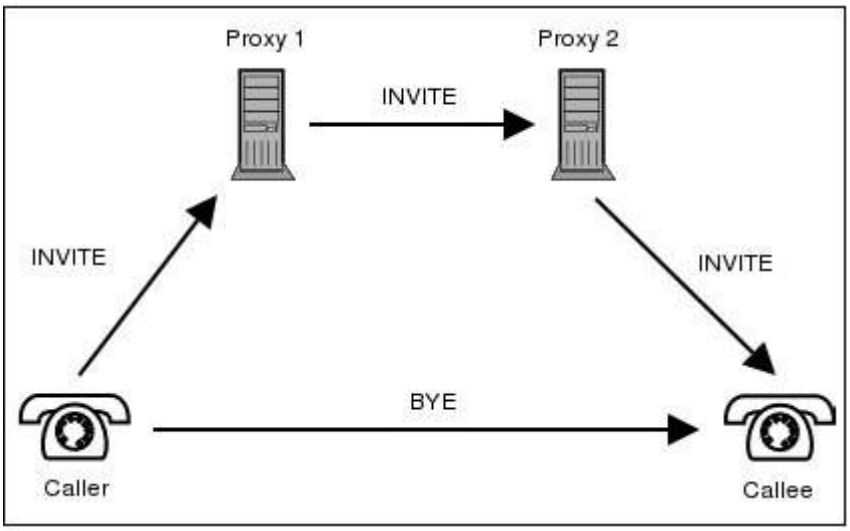


Σχήμα 7: Η βασική ροή κλήσεως στο SIP.

Στο σχ. 5 παρουσιάζεται η βασική ροή μιας κλήσης στο SIP όπου:

- κατ' αρχήν αποστέλλεται μία αίτηση INVITE στον proxy server ώστε να ξεκινήσει η σύνοδος επικοινωνίας
- ο proxy server στέλνει αμέσως μία απάντηση **100 Trying** στον καλών (Alice) για να σταματήσουν οι εκ νέου μεταδόσεις της αίτησης INVITE.
- ο proxy server αναζητά τη διεύθυνση του καλούμενου (Bob) στον location server. Αφού λάβει τη διεύθυνση, προωθεί την INVITE αίτηση περαιτέρω.
- στη συνέχεια, μία προσωρινή απάντηση **180 Ringing** που παράγεται από τον Bob επιστρέφει πίσω στην Alice.
- μία **200 OK** απάντηση παράγεται αμέσως μετά που ο Bob σηκώνει το τηλέφωνο.
- Μόλις η Alice λάβει **200 OK** αποστέλλει ένα **ACK** στον Bob.
- Την ίδια στιγμή, η σύνοδος έχει εγκατασταθεί και πακέτα RTP (συνομιλίες) αρχίζουν να μετακινούνται από τα δύο άκρα.
- Μετά τη συζήτηση, κάθε συμμετέχων (η Alice ή ο Bob) μπορεί να στείλει ένα αίτημα **BYE** για να τερματίσει τη σύνοδο.
- Το αίτημα **BYE** φτάνει απευθείας από την Alice στον Bob παρακάμπτοντας τον proxy server
- Τέλος, ο Bob στέλνει μία **200 OK** απάντηση για να επιβεβαιώσει την BYE αίτηση και τερματίζεται η συνεδρία.
- Στη παραπάνω βασική ροή κλήσεων, υπάρχουν στην ουσία τρεις συναλλαγές οι οποίες σημειώνονται ως 1, 2, 3.

3.2.4 SIP Trapezoid



Σχήμα 8: SIP Trapezoid. Πως βοηθά ο proxy να συνδεθεί ένας χρήστης με έναν άλλον.

Η διαδικασία που ακολουθεί ο proxy για να συνδέσει δύο χρήστες είναι η παρακάτω:

- Όταν ο καλών ξεκινά μια κλήση, ένα μήνυμα INVITE αποστέλλεται στον proxy server. Με την παραλαβή του INVITE, ο proxy server προσπαθεί να επιλύσει τη διεύθυνση του καλούμενου, με τη βοήθεια του διακομιστή DNS.
- Αφού λάβει την επόμενη διαδρομή, ο proxy server του καλούντος (ο proxy 1, είναι γνωστός και ως εξερχόμενος (outbound) διακομιστής μεσολάβησης) προωθεί την αίτηση INVITE στο διακομιστή μεσολάβησης του καλούμενου, ο οποίος δρα ως εισερχόμενος (Inbound) διακομιστή μεσολάβησης (proxy 2) για τον καλούμενο.
- Ο inbound proxy server συνδέεται με τον location server για να πάρει πληροφορίες σχετικά με τη διεύθυνση του καλούμενου.
- Αφού πάρει πληροφορίες από τον location server, προωθεί την κλήση στον προορισμό της.
- Μόλις οι UA's γνωρίσουν τη διεύθυνσή τους, μπορούν να παρακάμψουν την κλήση, δηλαδή, οι συνομιλίες να περνούν άμεσα.

3.2.5 SIP Messaging

Υπάρχουν δύο ειδών μηνύματα που διακινούνται στο SIP: οι αιτήσεις (**reguests**) και οι απαντήσεις (**responses**). Η διαδικασία που ακολουθεί ο proxy για να συνδέσει δύο χρήστες είναι η παρακάτω:

- Η αίτηση ξεκινάει με μία μέθοδο η οποία ορίζει την αίτηση και ένα Reguest-URI το οποίο καθορίζει που πρέπει να σταλεί η αίτηση.
- Παρομοίως, η απάντηση περιέχει έναν κωδικό απόκρισης.

Μέθοδοι αιτήσεων

Οι αιτήσεις SIP είναι οι κωδικοί που χρησιμοποιούνται για τη δημιουργία μιας επικοινωνίας. Για να ολοκληρωθούν, υπάρχουν απαντήσεις SIP που δείχνουν εάν η αίτηση πέτυχε ή απέτυχε. Αυτές οι αιτήσεις SIP που είναι γνωστές ως **METHODS** καθιστούν λειτουργικό το SIP μήνυμα. Οι ΜΕΘΟΔΟΙ μπορεί να θεωρηθούν ως SIP αιτήσεις, καθώς ζητούν από κάποιον UA ή server να λάβει κάποια συγκεκριμένη δράση.

Οι ΜΕΘΟΔΟΙ διακρίνονται σε δύο κατηγορίες:

Core Methods (Μέθοδοι πυρήνα) και

Extension Methods (Μέθοδοι επέκτασης)

3.2.5.1 Core Methods

- Πρώτη μέθοδος είναι η **INVITE** και χρειάζεται για να ξεκινήσει μια συνεδρία με έναν UA. Με άλλα λόγια, μια μέθοδος INVITE χρησιμοποιείται για να δημιουργήσει μια σύνοδο επικοινωνίας μεταξύ των UA's και μπορεί να περιέχει και πληροφορίες του μέσου (media) του καλούντος. Μία συνεδρία θεωρείται ότι δημιουργήθηκε όταν η INVITE λάβει απάντηση επιτυχίας (2xx) ή ACK.
-

Σχήμα 9: INVITE method.

Μία επιτυχημένη αίτηση INVITE ξεκινάει το διάλογο μεταξύ δύο UA ο οποίος συνεχίζεται μέχρι να αποσταλεί **BYE** και να τερματιστεί η σύνοδος.

Όταν αποσταλεί INVITE ενώ υπάρχει ενεργή σύνοδος τότε ονομάζεται re-INVITE. Η αίτηση Re-Invite χρησιμοποιείται για την αλλαγή των χαρακτηριστικών της συνεδρίας ή για να ανανεωθεί η συνεδρία. Στον πίνακα 3 φαίνεται ένα παράδειγμα χρήσης της μεθόδου.

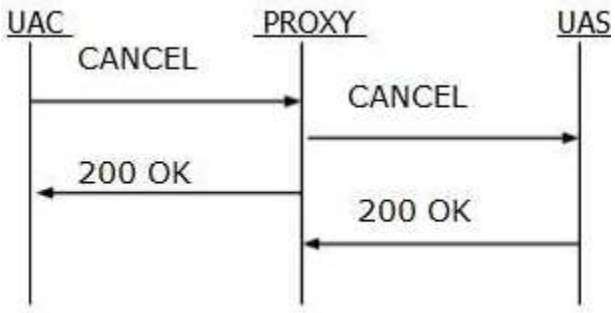
```
INVITE sips:Bob@TMC.com SIP/2.0
        Via: SIP/2.0/TLS client.ANC.com:5061;branch = z9hG4bK74bf9
        Max-Forwards: 70
        From: Alice<sips:Alice@TTP.com>;tag = 1234567
        To: Bob<sips:Bob@TMC.com>
        Call-ID: 12345601@192.168.2.1
        CSeq: 1 INVITE
        Contact: <sips:Alice@client.ANC.com>
        Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY
        Supported: replaces
        Content-Type: application/sdp
        Content-Length:. ..

        v = 0
        o = Alice 2890844526 2890844526 IN IP4 client.ANC.com
        s = Session SDP
        c = IN IP4 client.ANC.com
        t = 3034423619 0
        m = audio 49170 RTP/AVP 0
        a = rtpmap:0 PCMU/8000
```

Πίνακας 3. Παράδειγμα χρήσης της μεθόδου INVITE.

- η μέθοδος **BYE** χρησιμοποιείται για να τερματίσει μια συνεδρία. Είναι μία αίτηση SIP που μπορεί να αποσταλεί είτε από τον καλούντα είτε από τον καλούμενο για να τερματίσει μια συνεδρία αλλά όχι από τον διακομιστή μεσολάβησης ο οποίος συνήθως παρακάμπτεται κατά την αποστολή της. Η αίτηση αυτή δεν μπορεί να σταλεί όταν υπάρχει αίτηση INVITE σε εκκρεμότητα ή συνεδρία που δεν έχει εγκαθιδρυθεί ακόμη.

- η **REGISTER** καταχωρεί ένα UA. Αυτή η αίτηση αποστέλλεται από έναν UA σε ένα registar server. Η αίτηση μπορεί να προωθείται μέχρι να φτάσει σε έγκυρο registar του συγκεκριμένου domain. Στην επικεφαλίδα (header) και συγκεκριμένα στο σημείο **To** μεταφέρει το AOR (Address of Record) του χρήστη που έχει καταχωρηθεί. Ένας UA μπορεί να στείλει ένα αίτημα **REGISTER** για λογαριασμό άλλου UA. Αυτό είναι γνωστό ως καταχώριση τρίτου (third-party registration). Εδώ, η ετικέτα **From** περιέχει το URI του χρήστη που υποβάλει την καταχώριση για λογαριασμό του χρήστη που προσδιορίζεται στο **To** header.

- Η μέθοδος **CANCEL** χρησιμοποιείται για να τερματίσει μία σύνοδο όταν δεν έχει ακόμη ολοκληρωθεί. Οι UA's χρησιμοποιούν αυτή την αίτηση για να ακυρώσουν μία κλήση που βρίσκεται σε εκκρεμότητα και δεν έχει ολοκληρωθεί ακόμη. CANCEL μπορούν να στείλουν τόσο οι UA's όσο και οι proxy servers. Η μέθοδος αυτή είναι hop by hop δηλαδή περνάει από στοιχείο σε στοιχείο και παίρνει την απάντηση που αποστέλλεται από το επόμενο stateful στοιχείο.

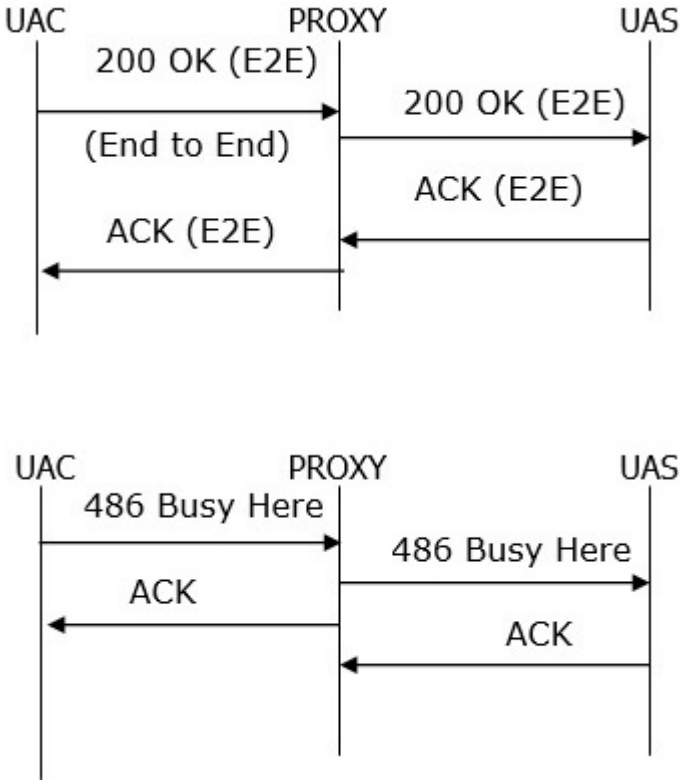


Σχήμα 10: Μέθοδος CANCEL – Hop by hop reguest.

- Η **ACK** χρησιμοποιείται για να αναγνωρίσει τις τελικές απαντήσεις σε μια μέθοδο INVITE. Η ACK πηγαίνει πάντα προς την κατεύθυνση της INVITE και μπορεί να περιέχει SDP μηνύματα (media characteristics), εάν δεν είναι διαθέσιμα στην INVITE.

Σχήμα 11. Ανταλλαγή SDP στην μέθοδο ACK.

Η ACK δεν μπορεί να χρησιμοποιηθεί για να τροποποιήσει την περιγραφή του μέσου (media) που έχει ήδη αποσταλεί στην αρχική INVITE.



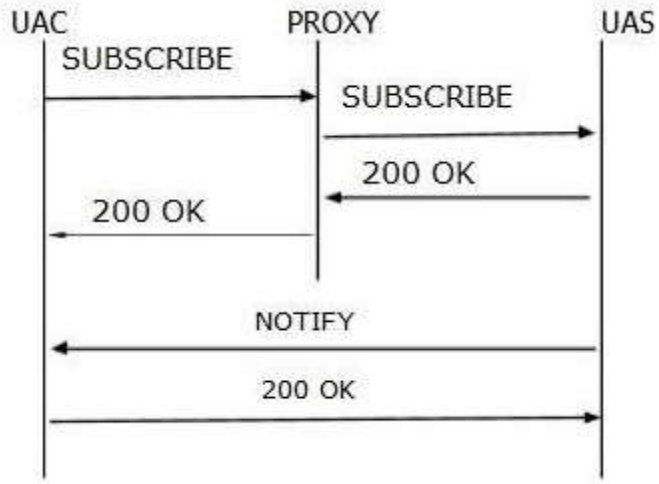
Σχήμα 12. Η μέθοδος ACK Hop by Hop.

Όταν ένας stateful proxy λάβει μία ACK, πρέπει να καθορίσει εάν η ACK πρέπει να προωθηθεί σε άλλο proxy ή UA. Για τις **2xx** απαντήσεις, η ACK θεωρείται ότι είναι από άκρο σε άκρο, ενώ για όλες τις άλλες τελικές απαντήσεις, λειτουργεί ως hop by hop όταν εμπλέκονται stateful proxies.

- Η μέθοδος **OPTIONS** χρησιμοποιείται για να ρωτήσει έναν UA ή ένα proxy server για τις δυνατότητές του και να ανακαλύψει την τρέχουσα διαθεσιμότητα του. Η απάντηση στο αίτημα, απαριθμεί τις δυνατότητες του UA ή του διακομιστή. Ένας proxy δεν δημιουργεί ποτέ μια αίτηση OPTIONS.

3.2.5.2 Extension Methods

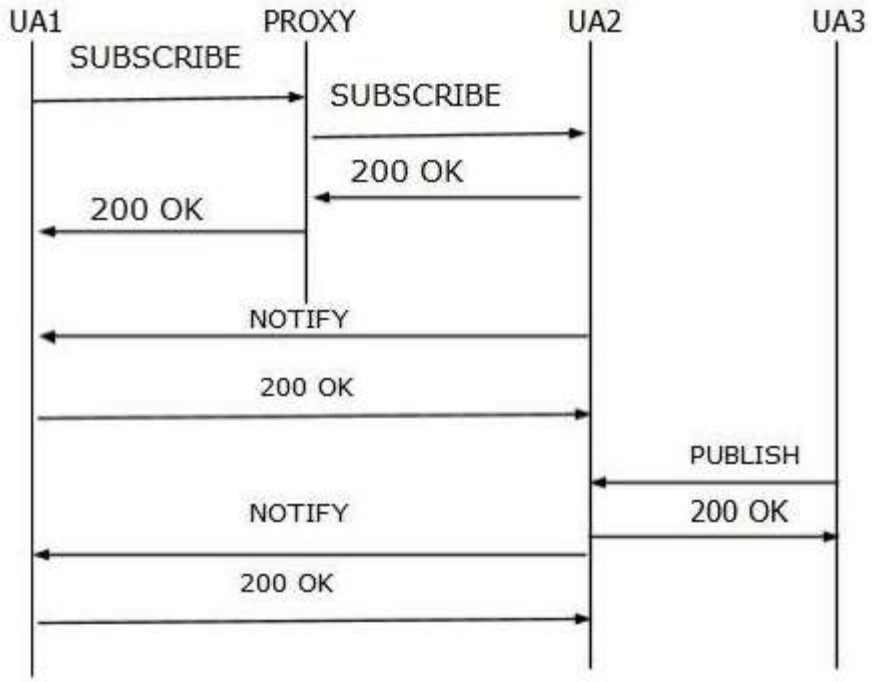
- **SUBSCRIBE.** Χρησιμοποιείται από τους UA's για να δημιουργήσει μια συνδρομή (subscription) με σκοπό να ειδοποιηθεί για ένα συγκεκριμένο γεγονός.



Σχήμα 13. Παράδειγμα της SUBSCRIBE και της NOTIFY ροής

- **NOTIFY.** Χρησιμοποιείται από τους UA's για να συλλάβει ένα συγκεκριμένο συμβάν. Συνήθως μία **NOTIFY** ενεργοποιείται μέσα σε διάλογο σε εξέλιξη.

- **PUBLISH.** Χρησιμοποιείται από τους UA's για την αποστολή πληροφοριών κατάστασης (event state) σε ένα διακομιστή.



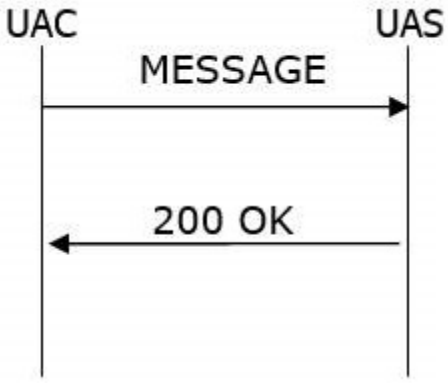
Σχήμα 14. Παράδειγμα της PUBLISH ενώ υπάρχει συνομιλία σε εξέλιξη.

- **REFER.** Χρησιμοποιείται από ένα UA για να παραπέμψει έναν άλλον UA να αποκτήσει πρόσβαση σε ένα URI για τον διάλογο.

- **INFO.** Χρησιμοποιείται από έναν UA για την αποστολή πληροφοριών σηματοδοσίας της κλήσης σε άλλον UA με τον οποίο έχει δημιουργήσει μια συνεδρία media.
- **UPDATE.** Χρησιμοποιείται για να τροποποιήσει την κατάσταση της συνόδου, αν αυτή δεν έχει ακόμη εγκατασταθεί. Ο χρήστης μπορεί να αλλάξει τον codec με αυτή.

Σχήμα 15. Παράδειγμα της UPDATE πριν εγκαθιδρυθεί η συνεδρία.

- **PRACK.** Χρησιμοποιείται για να αναγνωρίσει την παραλαβή μιας αξιόπιστης μεταφοράς των κωδικών απόκρισης (1xx).
- **MESSAGE.** Χρησιμοποιείται για να στείλουμε ένα άμεσο μήνυμα (IM- Instant message). Ένα IM αποτελείται συνήθως από σύντομα μηνύματα που ανταλλάσσονται σε πραγματικό χρόνο.



Σχήμα 16. Παράδειγμα της MESSAGE για ανταλλαγή μηνυμάτων.

3.2.6 Κωδικοί απόκρισης (Response Codes) στο SIP

Ένας κωδικός απόκρισης στο SIP είναι ένα μήνυμα που παράγεται από έναν UAS ή έναν SIP server ως απάντηση σε αίτημα ενός client. Θα μπορούσε να είναι μια επίσημη αναγνώριση για να εμποδίσει την αναμετάδοση των αιτήσεων από ένα UAC.

Υπάρχουν έξι ειδών αποκρίσεις στο SIP. Οσες ανήκουν στο εύρος **1xx** έως **5xx** είναι δανεικές από το HTTP ενώ οι **6xx** εισήχθησαν στο SIP. Οι απαντήσεις **1xx** θεωρούνται ως προσωρινές απαντήσεις ενώ οι υπόλοιπες είναι τελικές απαντήσεις.

S. No.	Λειτουργία & Περιγραφή
1	**1xx: Προσωρινές / Ενημερωτικές Αποκρίσεις**
	Ενημερωτικές απαντήσεις που χρησιμοποιούνται για να δείξουν τη πρόοδο της κλήσης και συνήθως είναι από άκρο σε άκρο (εκτός της 100 Trying).
	100 Trying
	180 Ringing
	181 Call is Being Forwarded
	182 Call Queued
	183 Session Progress
2	**2xx: Αποκρίσεις επιτυχίας**
	200 OK
	202 Accepted
3	**3xx: Redirect Responses**
	Αποκρίσεις που στέλνονται από redirect servers σε απάντηση της INVITE.
	300 Multiple Choices
	301 Moved Permanently
	302 Moved Temporarily
	305 Use Proxy
	380 Alternative Service
4	**4xx: Client Failure Responses**
	Απαντήσεις από τη πλευρά του client που δηλώνουν ότι ένα αίτημα δεν μπορεί να ικανοποιηθεί.
	402 Payment Required
	403 Forbidden
	404 Not Found
	405 Method Not Allowed
	406 Not Acceptable
	407 Proxy Authentication Required
	408 Request Timeout
	422 Session Timer Interval Too Small
	423 Interval Too Brief
	480 Temporarily Unavailable
	481 Dialog/Transaction Does Not Exist
	483 Too Many Hops
	486 Busy Here
	487 Request Terminated
5	**5xx: Server Failure Response**

	Ύπαρξη σφάλματος στο διακομιστή.
	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
6	**6xx: Global Failure Response**
	Σφάλματα που δηλώνουν ότι ο διακομιστής γνωρίζει ότι η αίτηση θα αποτύχει όσες φορές και αν επαναληφθεί. Ως αποτέλεσμα, η αίτηση δεν θα πρέπει να αποστέλλεται σε άλλες τοποθεσίες.
	600 Busy Everywhere
	603 Decline
	604 Does Not Exist Anywhere
	606 Not Acceptable

Πίνακας 4. Κωδικοί απόκρισης (Response Codes) στο SIP.

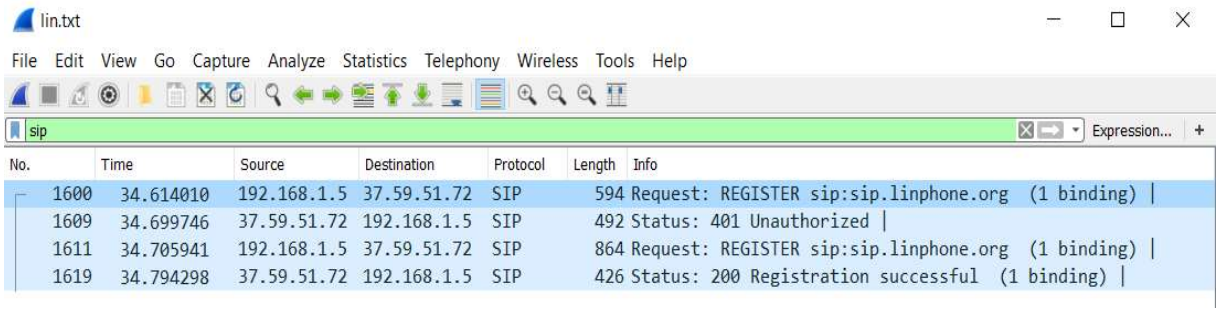
3.2.7 SIP Registration Transaction

Για να υπάρξει μια SIP επικοινωνία πρέπει όλοι οι συμμετέχοντες να έχουν ένα δημόσιο Uniform Resource Indicator (URI), μια δημόσια διεύθυνση δηλαδή, όπως είναι το exlax@sip.Linphone.org. Τα URIs περιγράφουν την τοποθεσία ενός proxy server. Στο παράδειγμα μας ο exlax πρέπει να συνδεθεί με τη συγκεκριμένη διεύθυνση. Αυτό θα γίνει με την αποστολή ενός REGISTER αιτήματος στον proxy server που είναι υπεύθυνος για την Linphone.org.

Σε μια τυπική επικοινωνία όταν ο client σηκώνει το τηλέφωνο στέλνει ένα REGISTER αίτημα στον registrar server χωρίς όμως κάποια στοιχεία αυθεντικοποίησης. Σε αυτήν την αίτηση ο server απαντάει αρνητικά και στέλνει ένα nonce ζητώντας στοιχεία αυθεντικοποίησης. Το σύστημα αυθεντικοποίησης που χρησιμοποιείται στις SIP συναλλαγές βασίζεται στο HTTP Digest Authentication και αξιοποιείται ως εξής:
Ο client στέλνει πίσω το nonce με το username του και ένα μίγμα από hashes που περιέχει το nonce, το username και ένα μυστικό κοινό password. Η σωστή αποκωδικοποίηση του hash ταυτοποιεί τον χρήστη στο server. Από τη στιγμή που ο χρήστης αυθεντικοποιηθεί στον server μπορεί να στέλνει INVITE αιτήματα μέσω του server αλλά και να παραλαμβάνει στον ταυτοποιημένο υπολογιστή του εισερχόμενα μηνύματα που θα του προωθούνται από τον server.

Εκτός από το HTTP Digest Authentication το SIP επιτρέπει στα τηλέφωνα να ταυτοποιούν τον server και μέσω Transport Layer Security (TLS) πριν αποσταλούν τα στοιχεία αυθεντικοποίησης, κάτι που προστατεύει από eavesdropping και brute force επιθέσεις του password με την κρυπτογράφησή του, καθώς επίσης αυθεντικοποιεί τον server στον SIP client μέσω των πληροφοριών που περιέχονται στο αυθεντικοποιημένο certificate.

Στο παρακάτω πίνακα φαίνεται η έναρξη της επικοινωνίας (Registration & απάντηση) χρησιμοποιώντας το wireshark



Πίνακας 5. Capturing της διαδικασία κατοχύρωσης (Register) με τη χρήση του WireShark.

Στο παράδειγμα αυτό ωθήσαμε την εφαρμογή Linphone να χρησιμοποιήσει UDP πρωτόκολλο.

```
Session Initiation Protocol (REGISTER) ---- (γραμμή 1611)
    Request-Line: REGISTER sip:sip.Linphone.org SIP/2.0
        Method: REGISTER
        Request-URI: sip:sip.Linphone.org
            Request-URI Host Part: sip.Linphone.org
        [Resent Packet: False]
    Message Header
        Via: SIP/2.0/UDP 192.168.1.5:5060;branch=z9hG4bK.wASvdNtbc;rport
            Transport: UDP
            Sent-by Address: 192.168.1.5
            Sent-by port: 5060
            Branch: z9hG4bK.wASvdNtbc
            RPort: rport
        From: <sip:dimalban@sip.Linphone.org>;tag=kjcx2OTSl
            SIP from address: sip:dimalban@sip.Linphone.org
                SIP from address User Part: dimalban
                SIP from address Host Part: sip.Linphone.org
            SIP from tag: kjcx2OTSl
        To: sip:dimalban@sip.Linphone.org
            SIP to address: sip:dimalban@sip.Linphone.org
                SIP to address User Part: dimalban
                SIP to address Host Part: sip.Linphone.org
        CSeq: 21 REGISTER
```

```
            Sequence Number: 21
            Method: REGISTER
        Call-ID: s8xzi6njuh
        Max-Forwards: 70
        Supported: replaces, outbound
        Accept: application/sdp
        Accept: text/plain
        Accept: application/vnd.gsma.rcs-ft-http+xml
        Contact:
            <sip:dimalban@192.168.1.5;transport=udp>;+sip.instance="<urn:uui
            d:282548b4-688d-45d8-be33-e5a6adb39dc5>"
          Contact URI: sip:dimalban@192.168.1.5;transport=udp
              Contact URI User Part: dimalban
              Contact URI Host Part: 192.168.1.5
              Contact URI parameter: transport=udp
              Contact parameter: +sip.instance= "<urn:uuid:282548b4-688d-
              45d8-be33-e5a6adb39dc5>"\r\n
        Expires: 3600
        User-Agent: Linphone/3.10.2 (belle-sip/1.5.0)
        [truncated]Authorization: Digest realm="sip.Linphone.org",
        nonce="nUvA2wAAAAB3Dum2AAA/4Cu/guYAAAAA",
        algorithm=MD5,opaque="+GNywA==",username="dimalban",
        uri="sip:sip.Linphone.org",response="10471efa75e3625ccd818c54caf5dda3
        ", cnonce=
        Authentication Scheme: Digest
        Realm: "sip.Linphone.org"
        Nonce Value: "nUvA2wAAAAB3Dum2AAA/4Cu/guYAAAAA"
        Algorithm: MD5, opaque="+GNywA=="
        Username: "dimalban"
        Authentication URI: "sip:sip.Linphone.org"
        Digest Authentication Response: "10471efa75e3625ccd818c54caf5dda3"
        CNonce Value: "g1QE~rueg-ELz44n"
        Nonce Count: 00000001, qop=auth
```

Και η απάντηση στη γραμμή 1619 (από τον πίνακα 2)

```
Session Initiation Protocol (200)
 Status-Line: SIP/2.0 200 Registration successful
        Status-Code: 200
        [Resent Packet: False]
        [Request Frame: 1611]
        [Response Time (ms): 88]
 Message Header
    Via: SIP/2.0/UDP 192.168.1.5:5060;branch=z9hG4bK.wASvdNtbc;rport=5060
        Transport: UDP
        Sent-by Address: 192.168.1.5
        Sent-by port: 5060
        Branch: z9hG4bK.wASvdNtbc
        RPort: 5060
    From: <sip:dimalban@sip.Linphone.org>;tag=kjcx2OTSl
        SIP from address: sip:dimalban@sip.Linphone.org
```

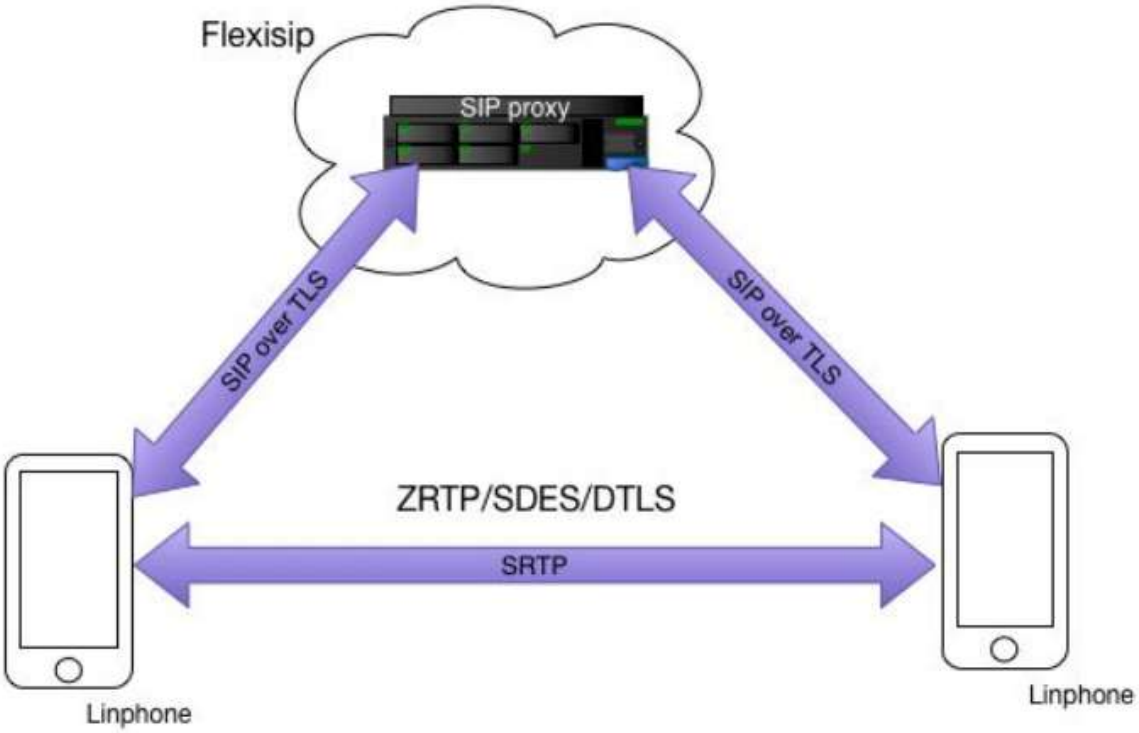
```
            SIP from address User Part: dimalban
            SIP from address Host Part: sip.Linphone.org
        SIP from tag: kjcx2OTSl
    To: <sip:dimalban@sip.Linphone.org>;tag=py0KN0S4FBSym
        SIP to address: sip:dimalban@sip.Linphone.org
            SIP to address User Part: dimalban
            SIP to address Host Part: sip.Linphone.org
        SIP to tag: py0KN0S4FBSym
    Call-ID: s8xzi6njuh
    CSeq: 21 REGISTER
        Sequence Number: 21
        Method: REGISTER
    Contact: <sip:dimalban@192.168.1.5:5060>;expires=3600;q=0.00
        Contact URI: sip:dimalban@192.168.1.5:5060
            Contact URI User Part: dimalban
            Contact URI Host Part: 192.168.1.5
            Contact URI Host Port: 5060
        Contact parameter: expires=3600
            Contact parameter: q=0.00
    Server: Flexisip/1.0.10 (sofia-sip-nta/2.0)
    Content-Length: 0
```

3.3 Πρωτόκολλα του Linphone

Όπως φαίνεται και στο πίνακα 2 (Αρχιτεκτονική του Linphone), η εφαρμογή Linphone χρησιμοποιεί μία έκδοση του πρωτοκόλλου SIP που ονομάζεται **belle-SIP** εξ'αιτίας της εταιρείας "Belledone Communications" που το αναπτύσσει. Για την επικοινωνία μπορεί να χρησιμοποιήσει **UDP**, **TCP** καθώς και **TLS** τόσο στο IPv4 όσο και στο IPv6. Όσον αφορά στη μετάδοση των media streams, αυτή είναι εφικτή με τη χρήση του πρωτοκόλλου **oRTP** το οποίο είναι η υλοποίηση του RTP (Real-time Transport Protocol) για το Linphone.

3.3.1 FlexiSIP

Στις ψηφιακές επικοινωνίες, που περιλαμβάνουν φωνή, βίντεο και μηνύματα περιέχονται ευαίσθητα δεδομένα του χρήστη που πρέπει να προστατευτούν από μη εξουσιοδοτημένη πρόσβαση. Το Linphone και το Flexisip παρέχουν ενσωματωμένες δυνατότητες ασφάλειας όπως φαίνεται στο παρακάτω σχήμα (Canavan 2001:18)[2].



Σχήμα 17. Δίκτυο Linphone / Flexisip SIP. (belledone 2016:2) [21]

Το πρώτο επίπεδο της ασφάλειας είναι να βεβαιωθούμε ότι τόσο η καταχώρηση του χρήστη (registration) όσο και η εγκατάσταση της κλήσης γίνονται με ασφαλή τρόπο. Οι ενέργειες αυτές αφορούν και το Linphone και τον Flexisip SIP proxy. Ο Linphone πελάτης δημιουργεί και διατηρεί μια SIP-TLS σύνδεση στον flexisip server και επαληθεύει την αυθεντικότητα του SIP server με βάση πιστοποιητικά x509 που ελέγχονται σε σχέση με μια λίστα των αξιόπιστων αρχών πιστοποίησης (root authorities).

Αυτό το πρώτο βήμα (αν γίνεται σωστά), εγγυάται την ακεραιότητα και την εμπιστευτικότητα όλων των πληροφοριών που ανταλλάσσονται μεταξύ του πελάτη Linphone και του διακομιστή Flexisip.

Το δεύτερο βήμα είναι να εξακριβώσει τη γνησιότητα των SIP μηνυμάτων που προέρχονται από τους πελάτες. Ο Flexisip server είναι υπεύθυνος για αυτή τη διαδικασία, χρησιμοποιώντας είτε επαλήθευση ταυτότητας (digest authentication) από μια βάση δεδομένων κωδικών πρόσβασης, ή καλύτερα χρησιμοποιώντας TLS client-based αυθεντικοποίηση: στην περίπτωση αυτή το πιστοποιητικό του πελάτη που παρουσιάζεται από τον πελάτη Linphone πρέπει να είναι έγκυρο και πρέπει να ταιριάζει με την ταυτότητα (κεφαλίδα From) των μηνυμάτων SIP. Η επιλογή μεταξύ των δύο

μεθόδων (http digest ή TLS client-based authentication) είναι θέμα διαμόρφωσης του Flexisip και του πελάτη Linphone.

Η μετάδοση φωνής και βίντεο πάνω από το RTP κρυπτογραφείται χρησιμοποιώντας AES είτε με 128 είτε με 256 bits μήκος κλειδιού με εξ' ορισμού τα 256 bits. Ο τρόπος που κρυπτογραφούνται τα RTP πακέτα περιγράφεται στο RFC3711. Για κρυπτογραφήσεις ανταλλαγής κλειδιών, το Linphone υλοποιεί 3 διαφορετικά πρότυπα IETF.

3.3.2 Πρωτόκολλο SDES - (Session Description Protocol Security Descriptions)

Αυτός είναι ο αρχικός τρόπος για την ανταλλαγή κλειδιών κρυπτογραφήσεως. Η βασική ιδέα είναι η ανταλλαγή κλειδιών να γίνεται κατά την διάρκεια εγκατάστασης της κλήσης. Στο RFC4568 περιγράφεται ένα χαρακτηριστικό του SDP που χρησιμοποιείται για να κωδικοποιήσει ένα κλειδί AES σε base64. Καθώς τα μηνύματα SDP είναι ασφαλισμένα μέσω SIP πάνω από TLS, η ανταλλαγή κλειδιών μπορεί να θεωρηθεί ασφαλής εφ' όσον είναι εγγυημένη η ακεραιότητα του δικτύου SIP.

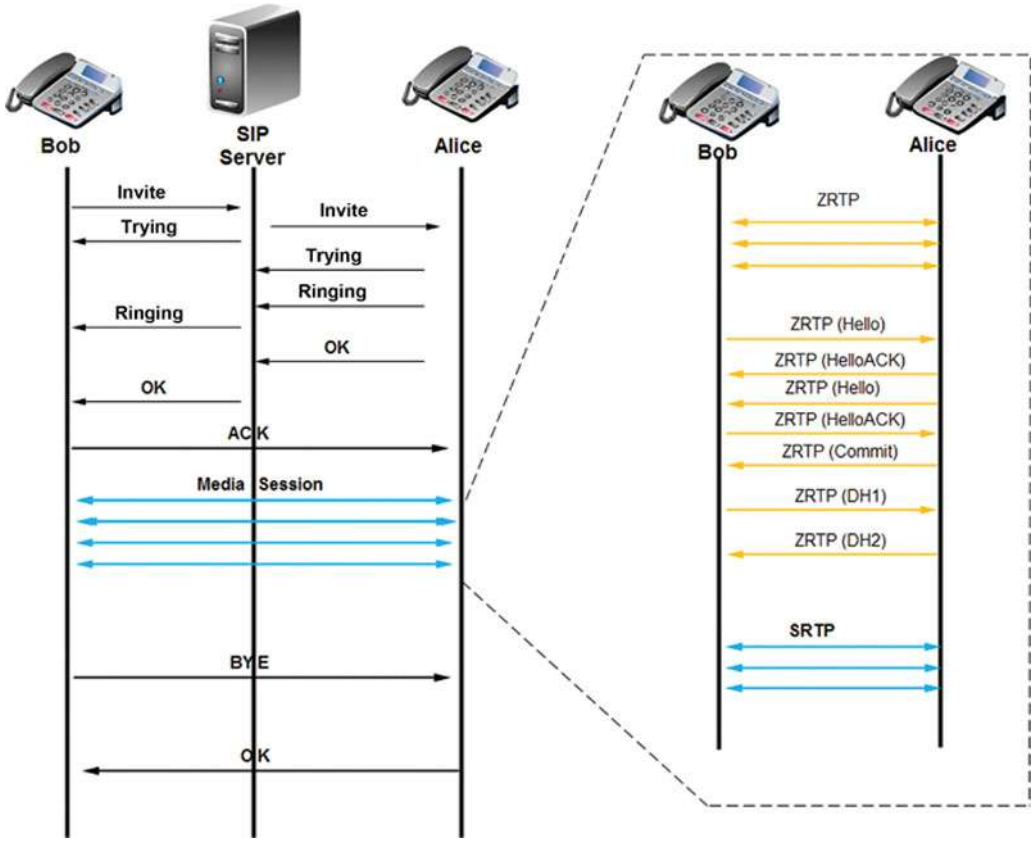
Η κυριότερη ανησυχία για το SDES είναι ότι η ασφάλεια του βασίζεται εξ ολοκλήρου στο δίκτυο SIP το οποίο αποτελείται από έναν ή περισσότερους SIP proxies. Όπως το SIP/TLS είναι μία λύση κρυπτογράφησης απ' άκρο σ' άκρο, κάθε SIP proxy έχει πρόσβαση στα κλειδιά κρυπτογράφησης σε απλό κείμενο. Το SDES απαιτεί όλοι οι SIP proxies που συμμετέχουν στην δρομολόγηση μιας κλήσης ή μηνύματος να είναι αξιόπιστοι.

3.3.3 Πρωτόκολλο ZRTP (Media path key agreement for unicast secure RTP)

Για να μειωθεί η πίεση που υπάρχει για ασφάλεια στους SIP proxies, το πρωτόκολλο ZRTP (RFC6189) προτείνει να γίνεται ανταλλαγή κρυπτογραφημένων κλειδιών από το ένα άκρο στο άλλο με τη τεχνική Diffie-Hellman, χωρίς την πολυπλοκότητα της υποδομής δημόσιου κλειδιού (PKI). Το κύριο μέλημά με τον αλγόριθμο Diffie-Hellman είναι ότι μπορεί να εκτεθεί σε επιθέσεις τύπου Man In The Middle (RSA Labs 2016:1) [25]. Για να αποφευχθεί αυτό, το ZRTP προτείνει ένα μηχανισμό που βασίζεται στη τεχνική SAS (Short Authentication String). Το σύντομο κείμενο SAS που αποστέλλεται στη τεχνική αυτή θα πρέπει να ελέγχεται από κάθε συμμετέχοντα, χρησιμοποιώντας τη φωνή κατά τη διάρκεια της κλήσης, και εγγυάται ότι δεν υπάρχει επίθεση MiTM. Το SAS

κείμενο καταστρέφεται με το πέρας της συνομιλίας κρατώντας όμως στοιχεία από αυτό για να αναμιχθούν με το επόμενο εφήμερο DH κλειδί στην επόμενη συνομιλία.

Σε αντίθεση με το SDES, η ύπαρξη αναξιόπιστου SIP proxy δεν θα έχει καμία επίδραση στην ακεραιότητα ή την εμπιστευτικότητα των πληροφοριών ήχου και εικόνας που ανταλλάσσονται μεταξύ δύο clients. Το Linphone υλοποιεί τη δική του έκδοση ZRTP μέσα στη βιβλιοθήκη bZRTP. Όσο για τη κρυπτογράφηση SIP/TLS, αυτή υλοποιείται στη βιβλιοθήκη mbedTLS (παλαιότερα νωστή ως PolarSSL).



Σχήμα 18. Μία σύνοδος SRTP μεταξύ Bob και Alice χρησιμοποιώντας πρωτόκολλο ZRTP (Regis 2014) [36].

3.3.4 Πρωτόκολλο DTLS (Datagram Transport Layer Security).

Το πρωτόκολλο αυτό αποτελεί επέκταση για το διαμοιρασμό κλειδιών για το Secure Real-time Transport Protocol (SRTP).

Όπως το ZRTP έτσι και το SRTP-DTLS (RFC5764) προσφέρει κρυπτογράφηση απ' άκρο σ' άκρο, αλλά κρυπτογραφεί τα κλειδιά που ανταλλάσσονται με τη χρήση δημόσιου/ιδιωτικού κλειδιού (public/private key) χρησιμοποιώντας πιστοποίηση X509 για authentication. Το κύριο πλεονέκτημα αυτού του πρωτοκόλλου είναι η

διαλειτουργικότητα του με το με WebRTC. Στο Linphone υλοποιείται και αυτό στη βιβλιοθήκη mbedTLS.

3.3.5 Αξιοπιστία μηνυμάτων και διαμοιρασμός αρχείων

Στο Linphone, τα μηνύματα μπορεί να ασφαλιστούν είτε από SIP-TLS, ή εάν απαιτείται από άκρο σε άκρο κρυπτογράφηση, μπορούν να χρησιμοποιηθούν κλειδιά κρυπτογράφησης που προέρχονται από την προηγούμενη συνεδρία ZRTP. Τότε τα μηνύματα κειμένου ή τα αρχεία που ανταλλάσσονται θα είναι κρυπτογραφημένα χρησιμοποιώντας κλειδιά ειδικά για κάθε χρήστη. Αυτή η τελευταία επιλογή απαιτεί να προηγηθεί μία φωνητική κλήση ZRTP πριν από την έναρξη μιας συνόδου μηνυμάτων.

3.3.6 Πρωτόκολλο TLS

Το Transport Layer Security (TLS) και ο προκάτοχός του, το Secure Sockets Layer (SSL), είναι πρωτόκολλα κρυπτογράφησης (Dierks 2008:2) [22]. Αρκετές εκδόσεις των πρωτοκόλλων βρίσκουν ευρεία χρήση σε εφαρμογές όπως η περιήγηση στο διαδίκτυο, το ηλεκτρονικό ταχυδρομείο, υπήρεσίες φαξ μέσω Διαδικτύου, instant messaging και voice-over-IP (VoIP). Πολλές ιστοσελίδες χρησιμοποιούν TLS για να ασφαλίσουν όλες τις επικοινωνίες μεταξύ των servers και των web browsers.

Το πρωτόκολλο TLS στοχεύει κυρίως στην διασφάλιση της ακεραιότητας και της ιδιωτικότητας των δεδομένων που διακινούνται. Η επικοινωνία με τη χρήση του TLS έχει μία ή περισσότερες από τις ακόλουθες ιδιότητες:

- Η σύνδεση είναι ιδιωτική (ή ασφαλής), επειδή χρησιμοποιείται συμμετρική κρυπτογραφία για την κρυπτογράφηση των δεδομένων που μεταδίδονται. Τα κλειδιά για αυτή τη συμμετρική κρυπτογράφηση παράγονται μοναδικά για κάθε σύνδεση και βασίζονται σε ένα κοινό κωδικό διαπραγματεύσιμο κατά την έναρξη της συνοδού (TLS handshake). Ο διακομιστής και ο πελάτης διαπραγματεύονται τις λεπτομέρειες για το ποιον αλγόριθμο κρυπτογράφησης και ποια κρυπτογραφημένα κλειδιά θα χρησιμοποιήσουν πριν μεταδοθεί το πρώτο byte δεδομένων. Η διαπραγμάτευση ενός κοινού κωδικού είναι επίσης ασφαλής ακόμη και σε επιθέσεις eavesdrop ή MITM και αξιόπιστη.
- Η ταυτότητα των επικοινωνούντων μερών μπορεί να πιστοποιηθεί χρησιμοποιώντας κρυπτογραφία δημόσιου κλειδιού. Αυτή η πιστοποίηση μπορεί

να γίνει προαιρετικά, αλλά γενικά απαιτείται για τουλάχιστον ένα από τα συμβαλλόμενα μέρη (συνήθως τον server).

- Η σύνδεση εξασφαλίζει την ακεραιότητα, διότι κάθε μήνυμα που μεταδίδεται περιλαμβάνει έναν έλεγχο ακεραιότητας μηνύματος χρησιμοποιώντας έναν κωδικό επαλήθευσης ταυτότητας μηνύματος για την πρόληψη απωλειών ή αλλοίωση των δεδομένων κατά τη μετάδοση.

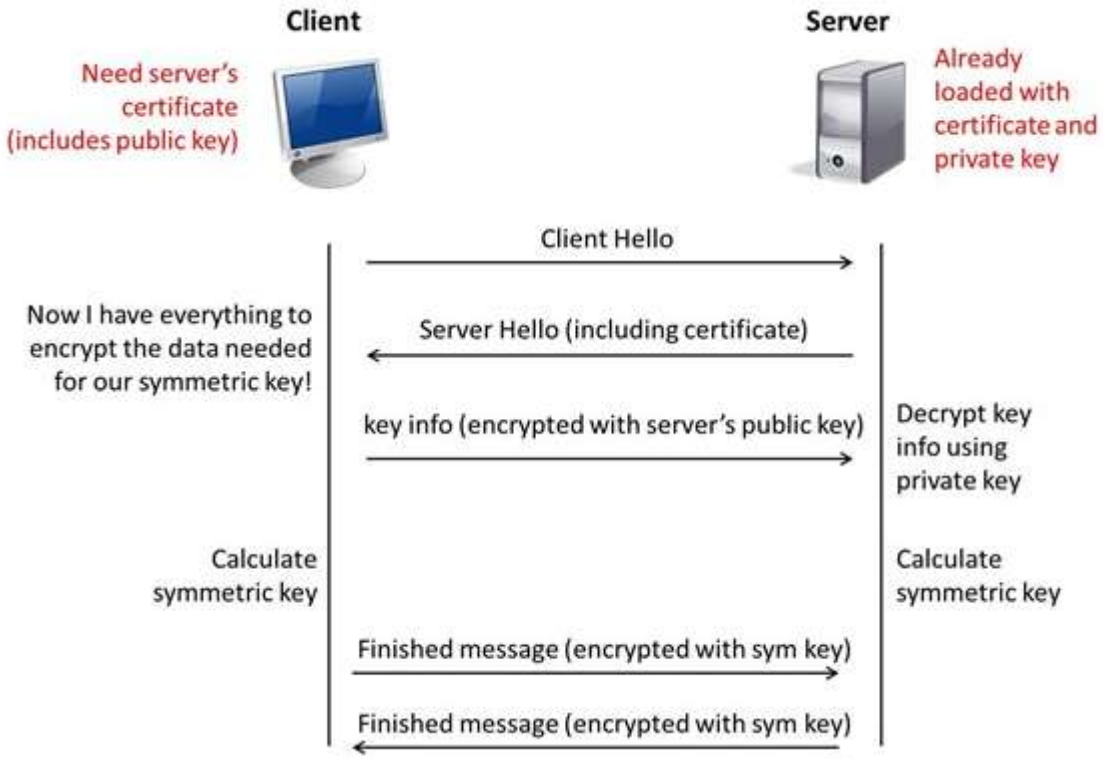
Εκτός από τις παραπάνω ιδιότητες, η προσεκτική διαμόρφωση του TLS μπορεί να παρέχει επιπλέον ιδιότητες σχετικά με την ιδιωτικότητα, όπως την forward secrecy, διασφαλίζοντας ότι κάθε μελλοντική δημοσιοποίηση των κλειδιών κρυπτογράφησης δεν μπορεί να χρησιμοποιηθεί για την αποκρυπτογράφηση οποιασδήποτε συνεδρίας TLS έχει καταγραφεί στο παρελθόν (Dierks 2008:2) [22].

Το TLS υποστηρίζει πολλές διαφορετικές μεθόδους για την ανταλλαγή κλειδιών, τη κρυπτογράφηση δεδομένων, και την επικύρωση της ακεραιότητας του μηνύματος. Ως αποτέλεσμα, η ασφαλή διαμόρφωση του TLS περιλαμβάνει πολλές ρυθμιζόμενες παραμέτρους.

Το πρωτόκολλο TLS αποτελείται από δύο στρώματα: το TLS record protocol και το TLS handshake protocol.

Το πρωτόκολλο TLS Record στοχεύει στην διατήρηση μιας ασφαλούς σύνδεσης μεταξύ δύο τελικών σημείων (παραδείγματος χάριν, client και server). Η διαπραγμάτευση των κρυπτογραφικών ιδιοτήτων όπως οι κρυπτογραφικές ακολουθίες και τα κλειδιά κρυπτογράφησης για την αντίστοιχη σύνδεση εκτελείται από το πρωτόκολλο TLS Handshake, το οποίο είναι ενθυλακωμένο μέσα στο πρωτόκολλο TLS Record.

Το πρωτόκολλο TLS Handshake χρησιμοποιείται για την αμοιβαία αυθεντικοποίηση client/server και για την διαπραγμάτευση των κρυπτογραφικών ιδιοτήτων (παραδείγματος χάριν, αλγόριθμοι κρυπτογράφησης και κλειδιά) της αντίστοιχης συνεδρίας. Το TLS Handshake πρέπει να ολοκληρωθεί επιτυχώς πριν διαβιβαστούν τα δεδομένα.



Σχήμα 19. TLS handshake μεταξύ client και server.

Το TLS έχει σχεδιαστεί με σκοπό να χρησιμοποιηθεί πάνω σε μια αξιόπιστη μεταφορά όπως το TCP ή SCTP. Αυτό εισάγει έναν περιορισμό για τις εφαρμογές που χρησιμοποιούν το UDP ως πρωτόκολλο μεταφοράς επειδή το TLS δεν μπορεί να χρησιμοποιηθεί με το UDP για να προστατεύσει τα SIP μηνύματα.

Η SIP RFC συστήνει τη χρήση του TLS για την παροχή της απαραίτητης προστασίας ενάντια σε επιθέσεις τύπου eavesdropping, message tampering, message replay, κ.λπ.. Όταν οι χρήστες θέλουν να δημιουργήσουν μια κλήση και να διατηρήσουν ένα επίπεδο απορρήτου, μπορούν να χρησιμοποιήσουν τα SIPS URI (secure SIP ή SIP over TLS) έτσι ώστε να χρησιμοποιείται τουλάχιστον μία ασφαλής, κρυπτογραφημένη μεταφορά για να προστατεύσει τα μηνύματα σηματοδοσίας μεταξύ δύο χρηστών.

3.3.7 Πρωτόκολλο oRTP

Το πρωτόκολλο αυτό είναι το γνωστό RTP (Real Time Protocol) πρωτόκολλο όπως αυτό περιγράφεται στο RFC3550 (Wagnon 2013:2) [23], είναι μία βιβλιοθήκη την οποία χρησιμοποιεί το Linphone για να μεταδώσει media streams. Μερικά από τα χαρακτηριστικά του είναι:

- Είναι γραμμένο σε C και λειτουργεί τόσο σε περιβάλλον Linux όσο και σε Windows.
- Εφαρμόζει το RFC3550 (RTP) (Schulzrinne 2003:2) [24] υλοποιώντας ένα εύχρηστο API με πρόσβαση υψηλού και χαμηλού επιπέδου.
- Διαθέτει ένα προσαρμοστικό αλγόριθμο ώστε ο δέκτης να προσαρμόζεται στο ρυθμό αποστολής δεδομένων του αποστολέα.
- Υποστηρίζει πολλαπλά προφίλ, με το προφίλ AV (RFC3551) να είναι το προεπιλεγμένο.
- Περιλαμβάνει ένα χρονοπρογραμματιστή πακέτων (packet scheduler) για την αποστολή και λήψη πακέτων "στην ώρα τους", σύμφωνα με την χρονική σήμανση τους (timestamp).
- Υποστηρίζει πολυπλεξία και έτσι εκατοντάδες RTP sessions μπορούν να συνυπάρχουν σε ένα μόνο νήμα (thread).
- Υποστηρίζει μέρος του RFC2833 (μεταφορά τηλεφωνικών τόνων - DTMF) πάνω από RTP.
- Περιλαμβάνει ένα API για να αναλύσει και να συνθέσει RTCP πακέτα, συμπεριλαμβανομένων και των AVPF RTCP πακέτων.
- Διατίθεται με άδεια Lesser Gnu Public License (LGPL).

Κεφάλαιο 4
Επιθέσεις στα ZRTP & TLS -

Στα περισσότερα VoIP περιβάλλοντα, τα δεδομένα εικόνας και ήχου διακινούνται χωρίς προστασία και μπορούν εύκολα να υποκλαπούν. Για να ασφαλιστεί η επικοινωνία δημιουργήθηκε το SRTP (Secure Realtime Transmission Protocol) (Baugher 2004:1) [26] το οποίο έτυχε ευρείας αποδοχής διότι παρέχει κρυπτογράφηση και έλεγχο ακεραιότητας. Το SRTP δεν εφαρμόζει διαδικασίες συμφωνίας κλειδιού, κάτι που σημαίνει ότι ένα μυστικό κύριο κλειδί πρέπει να παρέχεται από κάποιο άλλο μηχανισμό. Είναι πολύ σημαντικό τα δύο μέρη (ο Bob και η Alice) να έχουν ένα κοινό μυστικό κλειδί για να μπορούν να επικοινωνήσουν με ασφάλεια πάνω σε ένα μη ασφαλές δίκτυο, και αυτό το κλειδί πρέπει να μεταδοθεί με τη σειρά του πάνω από το ανασφαλές ίδιο δίκτυο. Εδώ αναλαμβάνει το ZRTP, το οποίο είναι ένα στην ουσία πρωτόκολλο συμφωνίας κλειδιού που έχει σχεδιαστεί ειδικά για τα συστήματα VoIP για να λύσει το πρόβλημα ανταλλαγής κλειδιών. Δημιουργήθηκε από τον Phil Zimmermann, ο οποίος είναι γνωστός για τη δημιουργία του λογισμικού κρυπτογράφησης PGP (Pretty Good Privacy) το οποίο επέτρεψε την ασφαλή συνομιλία μέσω ηλεκτρονικού ταχυδρομείου χωρίς την ανάγκη ενός PKI (Public Key Infrastructure).

4.1 Επιθέσεις στο ZRTP πρωτόκολλο (Petraschek 2008:1) [48]

Το ZRTP λοιπόν είναι ένα πρωτόκολλο συμφωνίας κλειδιού που καθιστά δυνατή την κρυπτογράφηση των media δεδομένων των VoIP συνδέσεων χωρίς να στηρίζονται σε ένα PKI και για να το πετύχει αυτό χρησιμοποιεί τον αλγόριθμο Diffie-Hellman (DH) (Hellman 1976:644) [27] για την ανταλλαγή κλειδιών με σκοπό τη δημιουργία ενός κοινού μυστικού. Ο DH όμως είναι γνωστό ότι είναι ευάλωτος σε MITM επιθέσεις και έτσι το ZRTP υλοποιεί διάφορους μηχανισμούς για να ανιχνεύσει μια επίθεση σε εξέλιξη. Το πιο σημαντικό αντίμετρο είναι μια τεχνική γνωστή ως Short Authentication String (SAS). Πρόκειται στην ουσία για μία κρυπτογραφική τιμή hash που υπολογίζεται πάνω από το κοινό μυστικό και εμφανίζονται στους τελικούς χρήστες μέσω μιας διεπαφής χρήστη (interface). Τα δύο μέρη, η Alice και ο Bob, πρέπει να συγκρίνουν αυτήν την τιμή

μέσω τηλεφώνου και αν οι τιμές ταιριάζουν, υπάρχει μια μεγάλη πιθανότητα ότι δεν συμβαίνει επίθεση MITM.

Εδώ μπορεί να εφαρμοστεί μία από τις πλέον σοβαρές επιθέσεις MiTM γνωστή με την ονομασία “Chess Grandmaster Attack” ή “Mafia Fraud Attack”. Η ονομασία οφείλεται στην ύπαρξη μίας τεχνικής (postal chess-ταχυδρομικό σκάκι) που επιτρέπει στον καθένα να νικήσει έναν grandmaster στο σκάκι παίζοντας ταυτόχρονα με δύο grandmasters, έχοντας τα λευκά με τον ένα και τα μαύρα με τον άλλον και αναμεταδίδοντας τις μεταξύ τους κινήσεις, ώστε στη πραγματικότητα να παίζει ο ένας grandmaster με τον άλλον.

Αντί να προσπαθήσει λοιπόν να σπάσει την κρυπτογράφηση πίσω από την ανταλλαγή κλειδιών μέσω DH, κάτι το οποίο θεωρείται ανέφικτο, η Mallory, η κακόβουλη εισβολέας, πραγματοποιεί μια Man-in-the-middle (MITM) επίθεση, όπου τόσο η Alice όσο και ο Bob συνδέονται εν αγνοία μαζί της. Αυτό οδηγεί σε δύο ξεχωριστές συνεδρίες ZRTP, μία μεταξύ της Aice και Mallory και μία μεταξύ Bob και Mallory. Η Mallory είναι το "έγκυρο" τελικό σημείο για αυτές τις δύο συνεδρίες και έτσι έχει πρόσβαση στα μη κρυπτογραφημένα δεδομένα τα οποία μπορεί να αναμεταδώσει μεταξύ των δύο συνεδριών. Φυσικά, αυτές οι δύο σύνοδοι έχουν διαφορετικά κλειδιά κρυπτογράφησης και διαφορετικές τιμές SAS. Χωρίς περαιτέρω ενέργειες, η επίθεση θα πρέπει να ανιχνεύεται αμέσως μόλις η Alice και ο Bob συγκρίνουν τις SAS τιμές τους. Ωστόσο, υπάρχουν διάφοροι τρόποι για τη Mallory να αποφύγει την ανίχνευση με επεμβαίνοντας στη SAS διαδικασία. όπως θα δούμε παρακάτω.

4.1.1. Σηματοδοσία και μεταφορά δεδομένων.

Ενώ τα δεδομένα ήχου και εικόνας μιας συνόδου επικοινωνίας ανταλλάσσονται με τη χρήση SRTP, η πραγματική ρύθμιση-εγκατάσταση της συνεδρίας (Rosenberg 2002:1) [28] γίνεται με τη χρήση του Πρωτοκόλλου Έναρξης Συνόδου (Session Initiation Protocol SIP). Το SIP χρησιμοποιείται για τη δημιουργία, τον τερματισμό, την ανακατεύθυνση και τον έλεγχο της VoIP σύνδεσης. Παρόμοια με το HTTP, είναι ένα πρωτόκολλο καθαρού κειμένου που περιέχει μια κεφαλίδα που αποτελείται από ένα αριθμό πεδίων, και ένα προαιρετικό σώμα (body). Οι πληροφορίες που αφορούν στις συνόδους περιέχονται μέσα σ’ αυτό το σώμα και περιγράφονται από το Session Description Protocol (SDP) (Handley 2006:2) [29]. Το σώμα SDP χρησιμοποιείται για ανταλλαγή πληροφοριών

όπως: τα χρησιμοποιούμενα πρωτόκολλα μεταφοράς πολυμέσων, διευθύνσεις IP, τις πόρτες (ports), και τα codecs μεταξύ των συνομιλούντων.

Μόλις καθιερωθεί μια συνεδρία, τα δεδομένα μεταφέρονται μεταξύ των UA μέσω SRTP ή μέσω της μη ασφαλούς έκδοσης του RTP. Σε αντίθεση με τα SIP μηνύματα που συνήθως δρομολογούνται μέσω διαδοχικών SIP proxies, τα δεδομένα SRTP θα μπορούσαν να μεταφέρονται απευθείας μεταξύ των UA's. Ως εκ τούτου, είναι σύνηθες να μιλάμε για δύο διαφορετικά επίπεδα: Τη σηματοδοσία, η οποία περιλαμβάνει κυκλοφορία SIP και SDP και χρησιμοποιείται για τη διαχείριση κλήσεων και να ρύθμισης των παραμέτρων της συνεδρίας. Από την άλλη πλευρά, το επίπεδο μεταφοράς των δεδομένων δηλώνει το κανάλι που χρησιμοποιείται για τη μεταφορά δεδομένων, όπως φωνή ή/και βίντεο.

4.1.2. Ανταλλαγή κλειδιών κατά Diffie-Helman.

Ο Diffie-Hellman είναι ένας βασικός αλγόριθμος που χρησιμοποιείται από δύο μέρη για να συμφωνήσουν σε ένα κοινό μυστικό πάνω από ένα μη ασφαλές κανάλι. Λειτουργεί ως εξής:

1. Η Alice και ο Bob συμφωνούν σε δύο μεγάλους πρώτους αριθμούς g (generator – γεννήτορας) και p οι οποίοι δεν χρειάζεται να κρατηθούν μυστικοί.
2. Η Alice επιλέγει ένα τυχαίο αριθμό a (χωρίς να τον ανακοινώνει) και υπολογίζει τη δημόσια τιμή του $A = g^a \bmod p$ την οποία και στέλνει στον Bob.
1. Ο Bob επιλέγει ένα τυχαίο αριθμό b (χωρίς να τον ανακοινώνει) και υπολογίζει τη δημόσια τιμή του $B = g^b \bmod p$ την οποία και στέλνει στην Alice.
3. Η Alice υπολογίζει το κοινό κλειδί $K = B^a \bmod p = g^{ab} \bmod p$
4. Ο Bob υπολογίζει το κοινό κλειδί $K = A^b \bmod p = g^{ba} \bmod p$
5. Η Alice και ο Bob έχουν πλέον το ίδιο κοινό κλειδί.

Πρέπει να σημειωθεί ότι ο Bob ο οποίος στέλνει τη δημόσια τιμή του B αφού έλαβε πρώτα τη δημόσια τιμή A της Alice, μπορεί να επηρεάσει το προκύπτον κλειδί K με την επιλογή του κατάλληλου ιδιωτικού αριθμού b, κάτι το οποίο θα οδηγήσει σε κινδύνους για την ασφάλεια όπως θα δούμε παρακάτω.

Ως εκ τούτου, το ZRTP χρησιμοποιεί το κλειδί ανταλλαγής Diffie-Hellman με τη χρήση hash, όπου η Alice στέλνει ένα κρυπτογραφημένο hash του A αντί του ίδιου του A στο – βήμα 2, δεσμεύοντας έτσι τον εαυτό της για τη δημόσια τιμή της χωρίς να τη στείλει.

Μόλις η Alice λάβει B από τον Bob μεταδίδει A. Ετσι, ούτε η Alice ούτε ο Bob έχουν τη δυνατότητα να μεταβάλουν το προκύπτον κοινόχρηστο κλειδί K.
Ο Diffie-Hellman έχει μια πολύ σημαντική ιδιότητα που ονομάζεται perfect forward secrecy. Αυτό σημαίνει ότι η αποκάλυψη ενός κλειδιού συνόδου δεν θα θέσει σε κίνδυνο τα προηγούμενα και τα μετέπειτα κλειδιά καθώς τα νέο-δημιουργηθέντα κλειδιά συνεδρίας δεν έχουν σχέση με τα προηγούμενη κλειδιά.

4.1.3. Short Authentication String (SAS).

Η ανταλλαγή κλειδιών DH δεν παρέχει έλεγχο ταυτότητας και επομένως είναι επιρρεπής σε επιθέσεις MITM. Το ZRTP χρησιμοποιεί ένα μηχανισμό που ονομάζεται Short Authentication String (SAS) για την ανίχνευση τέτοιων επιθέσεων. Η χρήση της τεχνικής SAS είναι ένα κοινός μηχανισμός για την πρόληψη επιθέσεων MITM σε κρυπτογραφημένες επικοινωνίες φωνής. Στο ZRTP, η SAS υπολογίζεται ως το HMAC του κοινό μυστικό s0 (το αποτέλεσμα της διαδικασίας συμφωνίας κλειδιού) και εμφανίζεται στην Alice και ο Bob μέσω κάποιας διεπαφής χρήστη. Η σύγκριση SAS γίνεται με την ανάγνωση της αξία SAS από το τηλέφωνο. Αν ταιριάζουν οι δύο τιμές, υπάρχει μεγάλη πιθανότητα να μην συμβαίνει επίθεση MITM. Το ZRTP παρέχει δύο διαφορετικές μεθόδους για την παραγωγή της αξίας SAS η οποία αποτελεί αντικείμενο διαπραγμάτευσης κατά τη διάρκεια της ZRTP χειραψίας (Handshake): Εάν χρησιμοποιείται κωδικοποίηση base32, τα 20 αριστερότερα bits της τιμής SAS παρέχονται σε μια μορφή που ονομάζεται z-base-32 (O' Whielacronx 2002:1) [30], η οποία έχει σχεδιαστεί για να είναι όσο το δυνατόν πιο εύχρηστη και κατανοητή από ανθρώπους. Δημιουργούνται λοιπόν 4 χαρακτήρες και εμφανίζονται στο χρήστη. Αν το handshake χρησιμοποιεί base256 τότε τα 16 αριστερότερα bits της τιμής SAS δημιουργούνται χρησιμοποιώντας τη λίστα λέξεων PGP (PGP Wordlist) (Juola 1996:1, Wikipedia 2016) [31][32]) η οποία είναι μια λίστα 512 διακριτών λέξεων που αντιστοιχούν bytes δεδομένων σε λέξεις ώστε να μπορούν να προφερθούν πάνω από ένα κανάλι φωνής. Η λειτουργία του είναι παρόμοια με το φωνητικό αλφάβητο του NATO (Wikipedia 2016) [33].

Λόγω της δέσμευσης χρήσης αλγορίθμων κατακερματισμού (hash) στην ανταλλαγή κλειδιού Diffie-Hellman, ούτε η Alice ούτε ο Bob μπορούν να επηρεάσουν την τιμή της συμβολοσειράς SAS. Το να μαντέψουν δε τη σωστή τιμή SAS έχει ένα ποσοστό επιτυχίας μόνο $1/2^{16}$, αν χρησιμοποιείται κωδικοποίηση base256.

4.1.4. Σπουδαιότητα

Το ZRTP θεωρείται το μελλοντικό πρότυπο για ανταλλαγή κλειδιών σε περιβάλλοντα VoIP. Κυρίως λόγω ζητημάτων πνευματικής ιδιοκτησίας, το ZRTP δεν είναι η προτιμητέα λύση για την IETF. Παρ 'όλα αυτά, είναι σημαντικό να εξεταστούν τα χαρακτηριστικά ασφαλείας του. Παρακάτω θα δούμε ότι μια επίθεση MITM στο ZRTP είναι αρκετά εύκολη αν οι τελικοί χρήστες δεν είναι προσεκτικοί και δεν συγκρίνουν τις SAS τιμές. Αν όμως προβούν στη σύγκριση τότε η επίθεση είναι θεωρητικά δυνατή, αλλά στην πράξη είναι πολύ πιθανό να ανιχνευθεί. Ειδικά όταν μιλάμε με κάποιον με γνώριμη φωνή (συμπεριλαμβανομένων των φίλων, των συναδέλφων και συγγενών) μία MITM είναι εξαιρετικά απίθανο να πετύχει.

4.2. Πώς λειτουργεί η επίθεση

Θα δούμε παρακάτω πώς μπορεί να υλοποιηθεί μία επίθεση MITM, καθώς και οι δυνατότητες για το χειρισμό της διαδικασίας SAS.

4.2.1. Μπαίνοντας στο δρόμο της σηματοδοσίας

Η πρώτη πρόκληση για τη Mallory είναι να μπει στο μονοπάτι σηματοδοσίας μεταξύ της Alice και του Μπομπ και υπάρχουν διάφοροι τρόποι για να επιτευχθεί αυτό. Στο επιλεγμένο περιβάλλον δικτύου (που απεικονίζεται στο σχήμα 20), η Mallory μπορεί να δημιουργήσει μία spoofing ARP (γνωστή και ως ARP request poisoning) (Whalen 2001:8) [34] επίθεση μεταξύ της Alice και του SIP proxy. Όλα τα πακέτα IP που θα έπρεπε κανονικά να μετακινούνται απ' ευθείας μεταξύ των δύο συνομιλούντων, κατευθύνονται στη Mallory που τα στέλνει στη συνέχεια στον αποδέκτη. Εάν η επικοινωνία δεν έχει προστασία ακεραιότητας μπορεί η Mallory να τροποποιήσει τα πακέτα κατά τη μεταφορά.

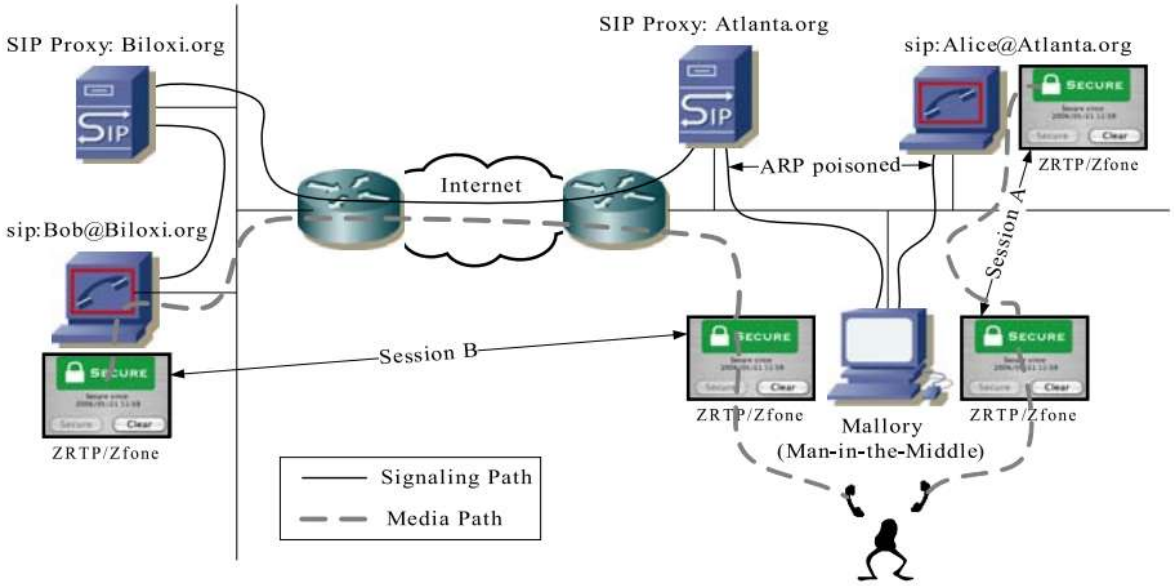
Η επίθεση μπορεί να συμβεί ακόμη και αν ο SIP proxy δεν βρίσκεται εντός του δικτύου LAN. Η ARP poisoning IP επίθεση λοιπόν πρέπει να συμβεί μεταξύ της Αλίκης και της προεπιλεγμένη πύλη ή του router που είναι υπεύθυνος για το δίκτυο του SIP proxy. Σε κάθε περίπτωση απαιτείται ο εισβολέας να έχει πρόσβαση στο LAN και πιο συγκεκριμένα στην εκπομπή τομέα (broadcast domain) της Αλίκης, αλλιώς δεν μπορεί

να λάβει χώρα. Σε άλλο σενάριο θα μπορούσε να μελετηθεί η χρήση η εγκατάσταση WLAN σημείου πρόσβασης (access point) όχι και τόσο νόμιμου (rogue) για να μπει στη διαδρομή σηματοδότησης. Επιπλέον, δεν χρειάζεται κατ' ανάγκη να εισάγουμε ένα ψεύτικο στοιχείο στη σηματοδότηση, είναι επίσης πιθανό να τεθεί σε κίνδυνο ένα υπάρχον στοιχείο του δικτύου, όπως ο SIP proxy server ή ένας δρομολογητής εκμεταλλευόμενοι μια υπάρχουσα ευπάθεια ή να συνεργαστούμε με τον χειριστή του SIP proxy.

Για τη διεξαγωγή της ARP poisoning attack, χρησιμοποιούμε το πρόγραμμα ανοιχτού κώδικα Ettercap που επιτρέπει να μπούμε στην IP επικοινωνία μεταξύ δύο (ή περισσοτέρων) hosts με έναν πολύ απλό τρόπο, εκτελώντας την εντολή

ettercap -i eth1 -T -q -M arp:remote /IPX/ /IPY/

και να "*ακούσουμε*" όλη τη συζήτηση μεταξύ των hosts με IP διευθύνσεις **IPX** και **IPY**



Σχήμα 20. Σηματοδοσία SIP.

4.2.2. Μπαίνοντας στο δρόμο των δεδομένων

Από τη στιγμή που η Mallory έχει μπει στο μονοπάτι σηματοδοσίας είναι σε θέση να τροποποιήσει τα πακέτα δεδομένων που μετακινούνται. Μπορεί λοιπόν να τροποποιήσει τα μηνύματα SIP έτσι ώστε τα δεδομένα να εκτρέπονται προς αυτή. Στην πράξη, ένα μήνυμα SIP αποτελείται από κεφαλίδες πολλών πρωτοκόλλων

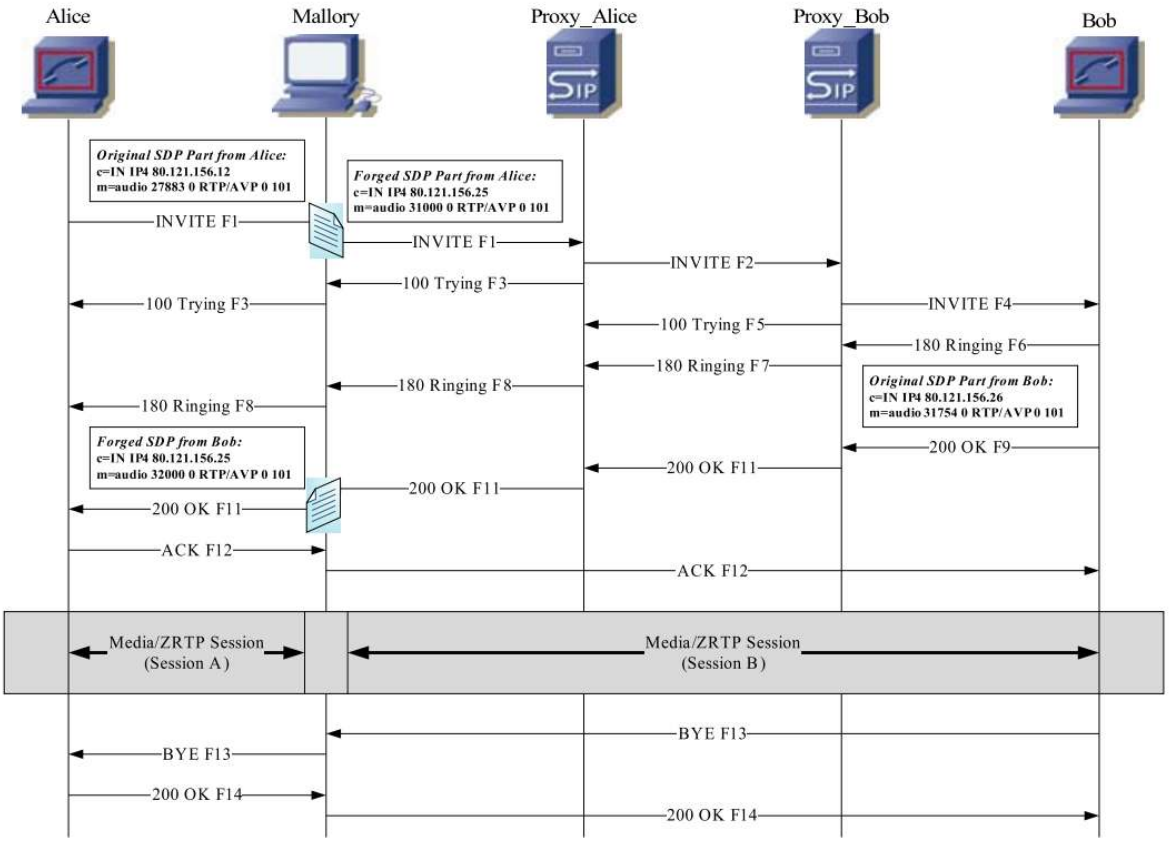
καθώς και διάφορα σώματα (body) – φορτία (payload). Κατά τη διάρκεια της εγκατάστασης της SIP συνεδρίας, ως payload χρησιμοποιείται το Session Description Protocol (SDP) το οποίο περιέχει χαρακτηριστικά που σηματοδοτούν στον συνομιλητή το που αναμένει να λάβει δεδομένα ο host και είναι αυτά ακριβώς τα χαρακτηριστικά που τροποποιεί η Mallory κατά την μετακίνηση τους.

c = IN IP4 IP_Adress_of_MitM_Host
m = audio media_port_of_MitM_Host O RTP/AVP 0 101

Όπου με c συμβολίζονται οι πληροφορίες σύνδεσης και με m τα media name transport address.

Αυτή η αντικατάσταση γίνεται χρησιμοποιώντας φίλτρα στο Ettercap, τα οποία επιτρέπουν τη τροποποίηση των πακέτων κατά τη μεταφορά τους ενώ περνούν από τον MitM host. Τα φίλτρα αυτά επιτρέπουν την αναζήτηση πακέτων που ταιριάζουν σε συγκεκριμένα πεδία στις κεφαλίδες και να προσθέσουν, να αντικαταστήσουν ή να αφαιρέσουν πληροφορίες μέσα σ' αυτά τα πακέτα χρησιμοποιώντας πολύ απλή γλώσσα προγραμματισμού.

Ως αποτέλεσμα, η Alice και ο Bob στέλνουν δεδομένα εν αγνοία τους άμεσα στην εισβολέα Mallory. Πρέπει να σημειωθεί ότι και τα δύο SDP μηνύματα (το αρχικό που περιέχεται στο - SIP – INVITE ή «200 OK», καθώς και η απάντηση στο μετέπειτα «200 OK» ή ACK μήνυμα) πρέπει να τροποποιηθούν προκειμένου να λάβουν τη ροή ήχου από την Alice στον Bob, καθώς και από τον Bob στην Alice.



Σχήμα 21. Σηματοδοσία SIP.

4.2.3 Ξεπερνώντας το Short Authentication String (SAS)

Εξ' αιτίας της αλλοίωσης της σηματοδοσίας, όλα τα δεδομένα από την Alice και από τον Bob αποστέλλονται στη Mallory η οποία, εάν δεν χρησιμοποιείται κρυπτογράφηση, απλώς προωθεί τα RTP πακέτα μεταξύ της Alice και του Bob αφού πρώτα λάβει γνώση της συνομιλίας η ίδια. Στη περίπτωση τώρα που η Alice θέλει να ξεκινήσει μια συνεδρία ZRTP, η Mallory θα λάβει το αρχικό ZRTP μήνυμα και αντί να το διαβιβάσει στον Bob, θα απαντήσει στην Αλίκη η ίδια και ταυτόχρονα θα δημιουργήσει μια ZRTP συνεδρία ως SAlice (αναφέρεται ως Συνεδρία Α). Ταυτόχρονα, η Mallory θα ξεκινήσει μια δεύτερη σύνοδο ZRTP ως SBob (Συνεδρία Β) με τον Bob. Στο τέλος θα υπάρχουν δύο συνεδρίες ZRTP (SAlice και SBob) με την Mallory να κάθεται στη μέση και να έχει πρόσβαση σε μη κρυπτογραφημένο ήχο. Τα κλειδιά συνόδου KAlice και KBob είναι διαφορετικά, δεδομένου ότι είναι το αποτέλεσμα δύο ανεξάρτητων ανταλλαγών κλειδιού Diffie-Hellman. Υπάρχουν τώρα αρκετές δυνατότητες-πιθανότητες για το πως η Mallory θα αναμεταδώσει τα δεδομένα μεταξύ SAlice και SBob και θα τις δούμε παρακάτω.

4.2.3.1 Άμεση αναμετάδοση (Direct Relay)

Η απλούστερη μέθοδος υποκλοπής είναι να πάρουμε το μη αποκρυπτογραφημένο μήνυμα από τη μία πλευρά και να το δώσουμε στην άλλη όπου θα κρυπτογραφηθεί και πάλι. Κατά τη διάρκεια της παράδοσης μεταξύ της SAlice και του SBob, η Mallory μπορεί εύκολα να υποκλέψει τα μη κρυπτογραφημένα δεδομένα

Πλεονεκτήματα:

- Μικρή καθυστέρηση
- Σταθερή ποιότητα αναμετάδοσης
- Η επίθεση μπορεί να αυτοματοποιηθεί πλήρως χωρίς να απαιτείται ανθρώπινη παρέμβαση
- Το κάθε μέρος ακούει την αυθεντική φωνή του άλλου χωρίς αλλοιώσεις-παρεμβάσεις

Μειονέκτημα:

- Εάν χρησιμοποιείται η σύγκριση SAS, η επίθεση θα αποκαλυφθεί. Οσο η Alice και ο Bob δεν συγκρίνουν την τιμή SAS δεν είναι σε θέση να αντιληφθούν ότι επιχειρείται επίθεση. Ωστόσο, δεδομένου ότι οι δύο συνεδρίες χρησιμοποιούν διαφορετικά κλειδιά συνόδου και η SAS προέρχεται από τα κλειδιά συνόδου, μια σύγκριση της SAS μεταξύ της Alice και του Bob θα αποτύχει και θα αποκαλυφθεί η MiTM επίθεση.

4.2.3.2 Άμεση αναμετάδοση & Μίμηση τιμής SAS

Η μέθοδος είναι παρόμοια με την προηγούμενη, αλλά όταν η σύγκριση SAS λαμβάνει χώρα κατά τη διάρκεια της συνομιλίας, ο επιτιθέμενος αντικαθιστά τον προφορική τιμή με μία κατάλληλη για τη συγκεκριμένη σύνοδο. Βέβαια δεν πρόκειται για εύκολη υπόθεση δεδομένου ότι πρέπει να μιμηθεί πραγματική φωνή. Μια διαφορετική φωνή κατά τη διάρκεια της επαλήθευσης SAS πιθανότατα θα δημιουργήσει καχυποψία. Επιπλέον, η εισαγωγή της φωνής πρέπει να είναι σε πραγματικό χρόνο και να επηρεάζει μόνο την αξία της SAS. Αυτό μπορεί να γίνει με διάφορους τρόπους ως εξής:

- Ένας άνθρωπος προσπαθεί να ακούγεται όσο το δυνατόν πιο αυθεντικός, κάτι που απαιτεί από τον εισβολέα να έχει δεξιότητες στη μίμηση της φωνής άλλων ανθρώπων. Μία φάση εκπαίδευσης είναι απαραίτητη για τον εισβολέα ώστε να εξοικειωθεί με τη φωνή του θύματός του ή να ψιθυρίσει την SAS τιμή, προσποιούμενος ότι το κάνει για λόγους ασφαλείας. Ωστόσο, αυτή η επίθεση

είναι πολύ επικίνδυνη και πολύ πιθανό να ανιχνευθεί από έναν προσεκτικό χρήστη.

- Έχοντας έναν υπολογιστή για να συνθέσει την SAS τιμή. Ένα τέτοιο σύστημα θα πρέπει πιθανότατα να είναι εκπαιδευμένο εκ των προτέρων π.χ. κάνοντας ψευδείς κλήσεις προς το θύμα και συλλέγοντας έτσι, δείγματα ήχου.
- Συλλέγοντας δείγματα ήχου όλων των τιμών SAS από το θύμα και αναπαράγοντας τα με τη σωστή σειρά (Rescorla 2007:4) [35]. Εάν χρησιμοποιείται z-base-32 κωδικοποίηση, ο εισβολέας πρέπει να συλλέξει 32 διαφορετικά δείγματα ήχου (24 γράμματα και 8 αριθμούς) για να έχει ένα πλήρες σύνολο χαρακτήρων (O' Whielacronx 2002:1) [30]. Αν χρησιμοποιείται ο κατάλογος λέξεων PGP για να γίνει η SAS, τότε πρέπει να αποκτηθούν 512 ξεχωριστά ηχητικά δείγματα.

Πλεονεκτήματα:

- Μικρή καθυστέρηση
- Κανονική ποιότητα αλληλεπίδρασης
- Οι συνομιλούντες ακούν τη πραγματική φωνή του άλλου.

Μειονεκτήματα:

- Ανεξάρτητα από την εφαρμοζόμενη μέθοδο, απαιτείται χειροκίνητη παρέμβαση.
- Η εισαγωγή των πλαστών τιμών SAS πρέπει να προγραμματιστεί με μεγάλη ακρίβεια
- Σε περίπτωση που η Mallory συλλέγει δείγματα ήχου SAS, πρέπει να το κάνει προκαταβολικά.

4.2.3.3 Άμεση αναμετάδοση & παράκαμψη SAS

Μετά το INVITE η Mallory προσπαθεί να είναι η πρώτη που θα ζητήσει σύγκριση τιμών SAS από την Alice και τον Bob, ενεργώντας ως Bob και Alice αντίστοιχα και έτσι μπορεί να αποφύγει να πει την τιμή SAS η ίδια. Το μόνο που έχει να κάνει είναι να επιβεβαιώσει την «ορθότητα» των τιμών SAS των άλλων. Η επίθεση αυτή είναι παρόμοια με την προηγούμενη, δεδομένου ότι η Mallory πρέπει να μιμηθεί τη φωνή της Alice και του Bob, όταν ζητούν σύγκριση τιμών SAS. Ωστόσο, είναι πολύ πιο εύκολο να ληφθεί ένα ηχητικό δείγμα που περιέχει ένα αίτημα για μια σύγκριση SAS από ένα δείγμα με όλες τις πιθανές τιμές της SAS. Υπάρχει βέβαια και εδώ το πρόβλημα του χρονισμού. Κατά τη διάρκεια της σύγκρισης SAS, οι δύο συνεδρίες ήχου πρέπει να απομονωθούν: Οι απαντήσεις SAS που προέρχονται από την Alice ή τον Bob δεν πρέπει να

αναμεταδίδονται στον απέναντι. Μόλις η Mallory επιβεβαιώσει την ορθότητα της SAS τόσο στην Alice όσο και στον Bob, πρέπει να γυρίσει σε λειτουργία "Direct Relay". Αν η Alice και ο Bob χρειάζονται διαφορετικό χρονικό διάστημα για να πούνε τις τιμές SAS, θα δημιουργηθεί κενό στη συζήτηση που μπορεί να αυξήσει τις υποψίες. Επίσης η κατάσταση αμέσως μετά που η Mallory έχει επιβεβαιώσει τις τιμές της SAS θα μπορούσε να προκαλέσει μια παράξενη συμπεριφορά αλληλεπίδρασης καθώς, πιθανών, και τα δύο θύματα περιμένουν ο ένας τον άλλον να ξεκινήσει την επικοινωνία.

Πλεονεκτήματα:

- Μικρή καθυστέρηση
- Κανονική ποιότητα αλληλεπίδρασης
- Οι συνομιλούντες ακούν τη πραγματική φωνή του άλλου.
- Η Mallory δεν χρειάζεται καμία τιμή SAS (απλώς επιβεβαιώνει αυτή που δέχεται)
- Δεν απαιτούνται δείγματα ήχου για το σύνολο των πιθανών τιμών SAS.

Μειονεκτήματα:

- Η παρέμβαση πρέπει να προγραμματιστεί με μεγάλη ακρίβεια
- Για τα δύο θύματα απαιτούνται δύο ηχητικά δείγματα όπου ζητούν την SAS τιμή, καθώς και ένα είδος επιβεβαίωσης.

4.2.3.4 Μασκάρεμα - Masquerade

Στην περίπτωση αυτή, η Mallory ακούει την Αλίκη ως S_{Alice} και σχεδόν ταυτόχρονα επαναλαμβάνει ό, τι άκουσε για τον Bob ως S_{Bob}. Επιπλέον, η Mallory μπορεί να ελέγξει ακόμη και τη συνομιλία με αυθαίρετο τρόπο, καθώς είναι σε θέση να τροποποιήσει, να παραλείψει ή να εισάγει λέξεις. Μόλις μία πλευρά προφέρει τη SAS τιμή, η Mallory πρέπει να αντιδράσει και να παράσχει την κατάλληλη τιμή SAS για κάθε μία κλήση από τις δύο που έχει κάνει αντί να τις επαναλάβει. Λογικά, αυτή η προσαρμογή πρέπει να γίνει και για τις δύο κατευθύνσεις.

Πλεονεκτήματα:

- Η σύγκριση SAS δεν έχει διαρρηχθεί
- Η φωνή που εκφωνεί τη SAS τιμή και η φωνή στο υπόλοιπο της συνομιλίας είναι η ίδια

Μειονεκτήματα:

- Λειτουργεί μόνο εάν οι επικοινωνούντες δεν γνωρίζουν ο ένας τη φωνή του άλλου

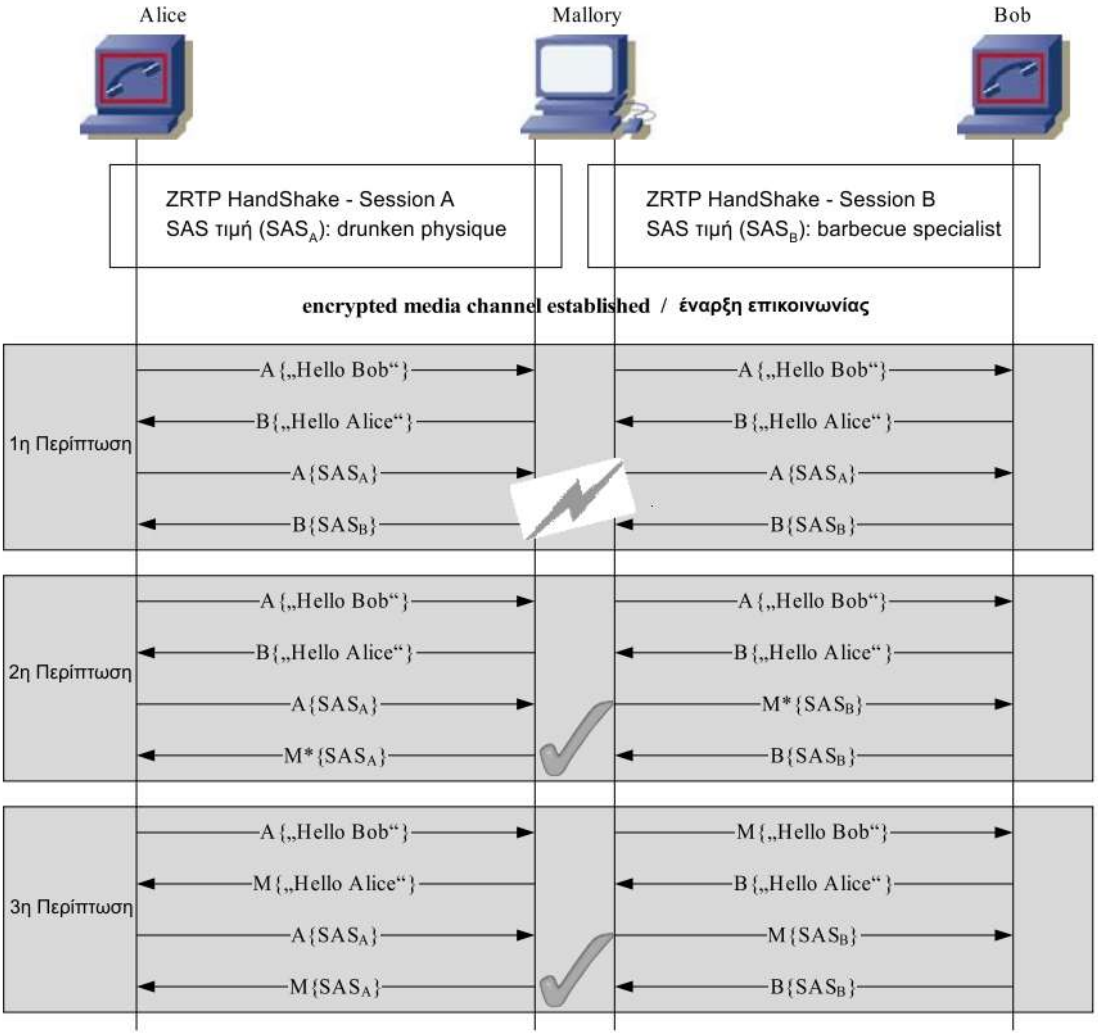
- Υπάρχει μάλλον μεγάλη καθυστέρηση, που οφείλεται στην επανάληψη των λέξεων που εκφωνούνται
- Κακή ποιότητα αλληλεπίδρασης ως αποτέλεσμα των παραπάνω:
- Η Mallory πρέπει να ακούσει και να μιλήσει την ίδια στιγμή, κάτι το οποίο είναι αρκετά δύσκολο έργο, ειδικά όταν πρέπει να το κάνει ιδίου στόματος. Ωστόσο, αν η Mallory έχει έναν συνεργάτη, θα μπορούσε αυτός να χειριστεί την μία από τις δύο κατευθύνσεις προκειμένου να βελτιωθεί η ανταπόκριση.

Δεδομένου ότι η επίθεση αυτή λειτουργεί μόνο σε σενάρια όπου οι επικοινωνούντες δεν γνωρίζουν ο ένας τη φωνή του άλλου, η χρήση του είναι πολύ περιορισμένη. Ένα σημαντικό σενάριο όπου αυτή η επίθεση θα μπορούσε να είναι εφικτή είναι το phone banking. Δεδομένου ότι η φωνή του ενός είναι πιθανότατα άγνωστη στον άλλον, ο περιορισμός της επίθεσης δεν είναι πλέον πρόβλημα.

4.2.4. Προϋπόθεση για να "σπάσει" το ZRTP: Νέο ZID

Κάθε ZRTP-aware λογισμικό ή συσκευή έχει ένα μοναδικό, τυχαίο 96-bit ZRTP-ID (ZID), το οποίο δημιουργείται κατά την εγκατάσταση. Χρησιμοποιείται ως αναφορά για την αποθήκευση ενός αριθμού τιμών (όπως ένα κρυπτογραφικό hash του χρησιμοποιημένου κοινόχρηστου μυστικού) σε μια τοπική βάση δεδομένων. Στην επόμενη κλήση από/προς το ίδιο ZID, αυτά τα προσωρινά αποθηκευμένα μυστικά ενσωματώνονται στον υπολογισμό του κοινού μυστικού που οδηγεί σε μια μορφή συνέχειας του κλειδιού. Η Mallory η οποία δεν γνωρίζει αυτά τα κοινόχρηστα μυστικά, δεν είναι σε θέση να αποκτήσει γνώση πάνω σ' αυτά. Οι αναλυτές-εμπνευστές του ZRTP υποστηρίζουν ότι ο μηχανισμός αυτός θα πρέπει να δυσκολέψει ακόμη περισσότερο τη Mallory να εξαπολύσει μια επίθεση:

Η συνέχεια των κοινόχρηστων αποθηκευμένων μυστικών, καθιστούν δυνατή την ανίχνευση μίας MiTM όταν εισχωρεί ή και όταν αποχωρεί από μία συνομιλία. Επίσης, εάν ο επιτιθέμενος προσπαθήσει να μείνει σε πολλές κλήσεις, αλλά αποτύχει να εκτελέσει μια επίθεση DH MITM για μία και μόνο κλήση τότε θα εξαιρεθεί μόνιμα και δεν θα μπορεί πλέον να συγχρονιστεί με την αλυσίδα της προσωρινά αποθηκευμένων κοινόχρηστων μυστικών.



Σχήμα 22. Συγκριτικό επιθέσεων SAS.

Για να προσπεράσει αυτή την ασφάλεια, η Mallory δημιουργεί ένα νέο ZID για κάθε μία σύνδεση και έτσι κάθε κλήση θα αντιμετωπίζεται ως «πρώτη κλήση". Δεδομένου ότι το ZRTP δεν προβλέπει καμία αντιστοίχιση μεταξύ SIP-URI και ZID, η κλήση από ένα ήδη γνωστό SIP-URI που περιέχει μια διαφορετική ZID δεν θα προκαλούσε υποψίες. Στην πραγματικότητα, υπάρχουν αρκετές περιπτώσεις όπου η σχέση μεταξύ ένα SIP-URI και ZID μπορεί να αλλάξει όπως στην επανεγκατάσταση του λογισμικού ή όταν γίνει κλήση από διαφορετική συσκευή.

Έτσι η Mallory μπορεί να αποφύγει να χρησιμοποιηθεί υλικό από προηγούμενες συνομιλίες μεταξύ της Alice και του Bob για τον υπολογισμό του κοινού μυστικού. Αυτό σημαίνει ότι με κάθε νέα κλήση μεταξύ της Alice και του Bob η επικοινωνία επανέρχεται στην αρχική κατάσταση (reset) κάτι που επιτρέπει στη Mallory να πραγματοποιήσει μια επιτυχημένη ZRTP χειραψία.

Ωστόσο, αυτή η αλλοίωση μπορεί να ανιχνευθεί από έναν προσεκτικό χρήστη σε ορισμένες περιπτώσεις. Αν η Alice πραγματοποιήσει μια επιτυχημένη SAS διαδικασία και θέσει την τιμή “verified” στην εφαρμογή κάποια στιγμή που η Mallory δεν ήταν παρούσα, αυτή θα μπορούσε να υποψιαστεί καθώς το πλαίσιο ελέγχου θα πήγαινε πίσω στην “unverified” κατάσταση. Αυτό προκαλείται από το γεγονός ότι για το νέο ZID που παρουσιάζει η Mallory, η σημαία “verified” δεν έχει τεθεί. Ωστόσο, αυτό θα μπορούσε να συμβεί και πραγματικά για τους λόγους που αναφέρθηκαν παραπάνω. Συμπληρωματικά, θα ήταν κατανοητό ότι η Mallory αποθηκεύει τα διαπιστευτήρια ZRTP για την SAlice και τον SBob ώστε να μπορέσει να συνεχίσει τις υποκλοπές με υλικό σχετικό με κλειδιά που έχουν ήδη εφαρμοστεί.

4.3 Επιθέσεις στο TLS πρωτόκολλο

Κατά τη διάρκεια των τελευταίων ετών, έχουν υπάρξει πολλές μεγάλες επιθέσεις στο TLS (Dierks 2008:12) [22], συμπεριλαμβανομένων των επιθέσεων στους πιο συχνά χρησιμοποιούμενους αλγόριθμους κρυπτογράφησης και τους τρόπους λειτουργίας. Τόσο ο AES-CBC όσο και ο RC4, που μαζί αποτελούν το μεγαλύτερο σε χρήση συνδυασμό, έχουν δεχτεί σοβαρές επιθέσεις στο πλαίσιο του TLS.

Η κατάσταση αυτή υπήρξε ένα από τα κίνητρα για τη δημιουργία της ομάδας εργασίας UTA, η οποία ήταν επιφορτισμένη με τη δημιουργία των γενικών και συγκεκριμένων συστάσεων πρωτόκολλο για τη χρήση του TLS και του DTLS (RFC6347). Υπάρχει ένα παλιό ρητό που αποδίδεται, αρκετά ειρωνικά, στην αμερικανική εθνική υπηρεσία ασφαλείας (NSA): "Οι επιθέσεις πάντα καλυτερεύουν, δεν χειροτερεύουν ποτέ" Δυστυχώς, η φράση αυτή αποδεικνύεται αληθής, έτσι ώστε οποιαδήποτε περιγραφή επιθέσεων ασφαλείας αποτελεί ένα μόνο στιγμιότυπο στο χρόνο (snapshot). Έτσι, οποιαδήποτε αναφορά σε επιθέσεις ισχύουν “προς το παρόν”.
Το SIPS μήνυμα είναι παρόμοιο με ένα SIP (unencrypted) μήνυμα που μεταφέρεται άνω των UDP, TCP, ή STCP. Οι μεγαλύτερες διαφορές είναι οι ακόλουθες:

- Η σύνταξη του URI ορίζεται ως sips: alice@domainb.com.
- Η μεταφορά γίνεται μέσω του TLS, αντί του UDP ή TCP.
- Η SIPS πόρτα είναι η 5061, αντί της 5060, η οποία είναι δεσμευμένη για τα UDP και TCP.

Το TLS παρέχει τα μέσα για αμοιβαία αυθεντικοποίηση χρησιμοποιώντας πιστοποιητικά για την προστασία από τις «man-in-the-middle» επιθέσεις. Η συσκευή μπορεί να επικυρώσει τον εαυτό της στο δίκτυο, αλλά μπορεί επίσης να ελέγξει την αυθεντικότητα του SIP proxy (ή του SIP registrar). Το προτεινόμενο πρότυπο κρυπτογράφησης για να χρησιμοποιηθεί στο SIPS είναι το AES (Advanced Encryption Standard), χρησιμοποιώντας ένα 128-bits ή 256-bits κλειδί στην CBC (Cipher Block Chaining) μέθοδο και ο προτεινόμενος κώδικας επικύρωσης μηνυμάτων είναι ο SHA-1 για την παροχή της ακεραιότητας.

Ένα επιπρόσθετο όφελος στην χρήση του SIPS είναι η δυνατότητα να ανταλλαχθούν τα κλειδιά κρυπτογράφησης με σκοπό να κρυπτογραφηθεί το ρεύμα δεδομένων χρησιμοποιώντας το ZRTP. Παραδείγματος χάριν, το SDescriptions μπορεί να χρησιμοποιηθεί μέσα σε ένα SIPS INVITE αίτημα για να ανταλλάξει το κύριο κλειδί μεταξύ δύο συμμετεχόντων. Το κλειδί κρυπτογράφησης παρέχεται στο SDP τμήμα του SIPS INVITE στην a=crypto ιδιότητα.

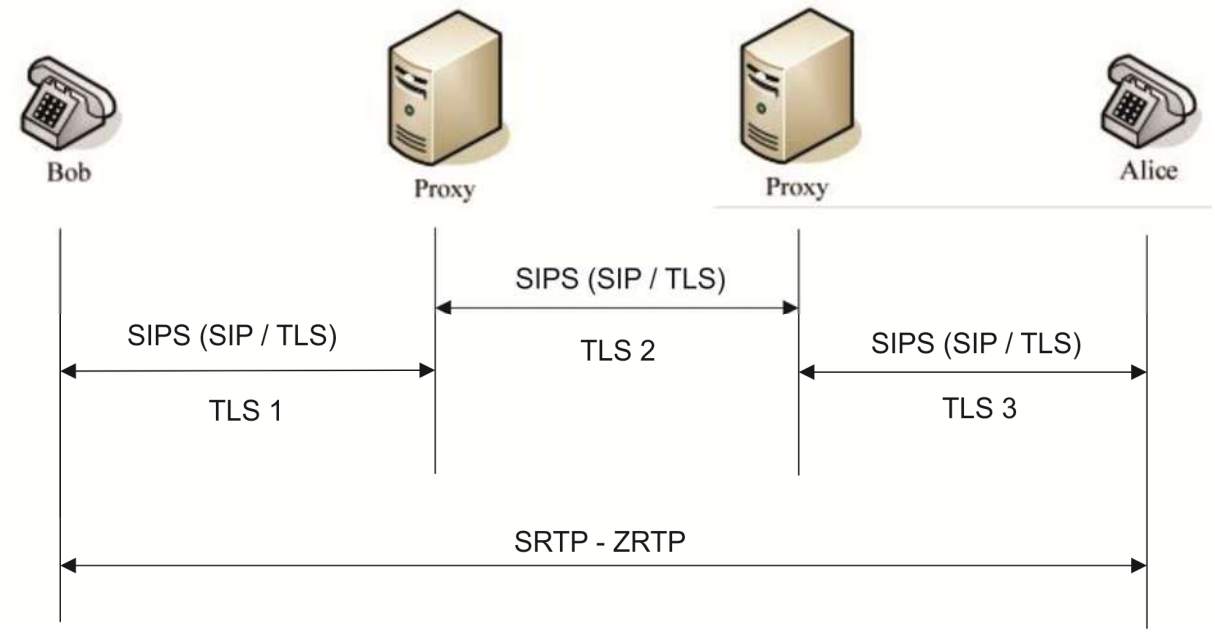
```
INVITE sip:hackme@sip.linphone.org SIP/2.0
Via: SIP/2.0/TLS 192.168.1.69:12797;branch=z9hG4bK.GYq4dRf7d;rport
From: <sip:dimalban@sip.linphone.org>;tag=M2GGrVdIn
To: "hackme" <sip:hackme@sip.linphone.org>
CSeq: 20 INVITE
Call-ID: 5nbMe8VmyJ
Max-Forwards: 70
Supported: replaces, outbound
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO, UPDATE
Content-Type: application/sdp
Content-Length: 877
Contact: <sip:dimalban@79.129.114.191:12797;transport=tls>;+sip.instance="<urn:uuid:6968ddbd-
3072-476f-bfb4-8223740a4052>"
User-Agent: Linphone/3.10.2 (belle-sip/1.5.0)
```

Πίνακας 6. Το SIP τμήμα ενός SIPS μηνύματος

```
v=0
o=dimalban 2843 1451 IN IP4 79.129.114.191
s=Talk
c=IN IP4 79.129.114.191
t=0 0
a=rtcp-xr:rcvr-rtt=all:10000 stat-summary=loss,dup,jitt,TTL voip-metrics
a=zrtp-hash:1.10 ccc433800098fdb4533fa1751170b81bc4235fc89968dc3ecfb5122ec0007a0a
m=audio 7078 RTP/AVPF 96 97 98 0 8 101 99 100
a=crypto:1 AES_CM_128_HMAC_SHA1_80 inline: ZuaD5fPsk1w3K4+x1aRzivUoL2MbkUrXG20xz2mr
```

Πίνακας 7. Το SDP τμήμα ενός SIPS μηνύματος με τις SDescriptions crypto ιδιότητες.

Όπως είπαμε στο προηγούμενο κεφάλαιο, αν και το TLS παρέχει την εμπιστευτικότητα μεταξύ δύο τελικών σημείων (σχέση client/server), δεν υποστηρίζει την άμεση «από-άκρο-σε-άκρο» (end-to-end) εμπιστευτικότητα μεταξύ των δύο μερών που θέλουν να επικοινωνήσουν και συνδέονται μέσω ενδιάμεσων SIP proxies. Για κάθε τμήμα, μια ευδιάκριτη TLS σύνδεση πρέπει να εγκατασταθεί. Έτσι, μπορεί να επικοινωνήσει ο χρήστης με τον proxy server, και ο ο proxy με άλλον proxy, χρησιμοποιώντας TLS, αλλά ο ένας χρήστης με τον άλλον δεν μπορούν να χρησιμοποιήσουν TLS και εδώ έρχονται τα ZRTP και SRTP πρωτόκολλα να καλύψουν τα κενά.



Σχήμα 23. SIP μέσω TLS και SRTP, ZRTP

Κάθε ενδιάμεσος SIP proxy πρέπει να αναλύσει τις SIP ετικέτες για να κατευθύνει ανάλογα το μήνυμα, και επομένως η SSL σύνδεση ολοκληρώνεται και επανιδρύεται μεταξύ των hops. Για κάθε hop (παραδείγματος χάριν, μια σύνδεση μεταξύ του Bob και του proxy της Α περιοχής του), υπάρχει μια ευδιάκριτη SSL σύνδεση που εγκαθίσταται (SSL1). Αυτή η σύνδεση μπορεί να διατηρήσει μια διαφορετική πολιτική ασφαλείας όπως ισχυρότερες ή πιο αδύναμες κρυπτογραφικές ακολουθίες από το επόμενο hop μεταξύ του proxy της περιοχής Α και του proxy της περιοχής Β (SSL2). Το ίδιο μπορεί να ειπωθεί για τη σύνδεση SSL3.

Σε πολλές περιπτώσεις (ή και σε όλες), δεν είναι γνωστό εάν ένας ενδιάμεσος SIP proxy που βρίσκεται πέρα από την περιοχή του χρήστη μπορεί να υποστηρίξει παρόμοια

ήισχυρότερη πολιτική ασφαλείας ή ακόμα και να υποστηρίζει TLS. Αν ισχύει κάτι τέτοιο δημιουργείται ισχυρότατο κενό ασφαλείας διότι:

- Η προσπάθεια να εγκατασταθεί η κλήση μπορεί να είναι ανεπιτυχής ανάλογα με το πώς επιβάλλεται η πολιτική ασφαλείας του χρήστη.
- Η σύνδεση μπορεί να εγκατασταθεί με μια πιο αδύναμη ισχύ των κρυπτογραφικών ακολουθιών από αυτή που καθορίζεται στην πολιτική του χρήστη.
- Η σύνδεση μπορεί να εγκατασταθεί χωρίς προστασία μεταξύ των δύο proxy στο συγκεκριμένο τμήμα δικτύου.
- Η σύνδεση μπορεί να εγκατασταθεί χωρίς καθόλου προστασία.

Ο,τιδήποτε και αν ισχύει, ο τελικός χρήστης πιθανότατα δεν γνωρίζει τις ασυνέπειες που εμφανίζονται και διατηρεί μια ψεύτικη αίσθηση ασφαλείας. Παραδείγματος χάριν, κατά την δημιουργία μιας διεθνούς κλήσης, δεν υπάρχει εγγύηση ότι όλοι οι ενδιάμεσοι φορείς παροχής υπηρεσιών θα υποστηρίξουν το SIPS είτε λόγω των τεχνολογικών περιορισμών είτε των ρυθμιστικών περιορισμών. Αυτή η έλλειψη της από άκρο σε άκρο εμπιστευτικότητας θα εκθέσει τα κλειδιά κρυπτογράφησης που χρησιμοποιούνται με το ZRTP εάν η διαπραγμάτευση των κλειδιών πραγματοποιείται χρησιμοποιώντας το SIP (UDP, TCP) και όχι το SIPS. Από την άλλη μεριά, τα δεδομένα δεν χρειάζεται να περάσουν μέσω των ενδιάμεσων συστατικών όπως κάνει το SIP, εκτός αν έχει συντονιστεί για να κάνει έτσι. Αντ' αυτού, μια peer-to-peer σύνδεση εγκαθίσταται, όπως απεικονίζεται στο σχήμα 18, για την οποία η εμπιστευτικότητα επιτυγχάνεται χρησιμοποιώντας το SRTP, ZRTP.

Το TLS έχει μειονεκτήματα τα οποία μπορεί να εκμεταλλευτεί κάποιος

- Απαιτεί μια PKI υποδομή για να επιβάλει την αμοιβαία αυθεντικοποίηση στο SSL στρώμα.
- Δεν παρέχει άμεση από άκρο σε άκρο εμπιστευτικότητα. Απαιτεί τη λήξη και τη δημιουργία μιας νέας συνεδρίας σε κάθε hop (παραδείγματος χάριν, μεταξύ των SIP proxy ή των ελεγκτών συνόρων συνεδρίας).
- Μπορεί να χρησιμοποιηθεί με τα TCP και SCTP αλλά όχι με το UDP, το οποίο προσκρούει στις SIP εφαρμογές που χρησιμοποιούν το UDP.
- Πολλές SIP εφαρμογές στα δίκτυα επιχειρήσεων και μεταφορέων χρησιμοποιούν αποκλειστικά SIP πάνω σε UDP.

- Ευάλωτο στις DOS επιθέσεις όπως TCP floods και τις RSTs (επαναρύθμιση σύνδεσης). Μια επίθεση TCP flood στοχεύει να καταναλώσει τους πόρους του συστήματος π.χ. κύκλοι CPU, εκτελώντας RSA αποκρυπτογράφηση. Επίσης, ένας επιτιθέμενος μπορεί να παράγει μεταμφιεσμένα RST πακέτα (spoofed) ή TLS αρχεία για να τερματίσει μια σύνδεση πρόωρα.

4.3.1 Γνωστές επιθέσεις στο SSL/TLS (Sheffer 2015:1) [37]

- **Self-signed certificate attack**

  Το πιο απλό σενάριο επίθεσης ενδιάμεσου σε μια σύνδεση TLS από είναι αυτό της εγκαθίδρυσης μιας TLS σύνδεσης ανάμεσα στον πελάτη και τον ενδιάμεσο και άλλης μίας ανάμεσα στον ενδιάμεσο και το εξυπηρετητή, με τη χρήση ενός πιστοποιητικού που είναι υπογεγραμμένο από τον ίδιο τον ενδιάμεσο αντί για μια αναγνωρισμένη Αρχή Πιστοποίησης. Ο ενδιάμεσος δηλαδή λειτουργεί σαν εξυπηρετητής όταν επικοινωνεί με τον πελάτη και σαν client όταν επικοινωνεί με τον server. Στην περίπτωση αυτή, οι περισσότερες εφαρμογές πελάτη εμφανίζουν κάποιο μήνυμα λάθους, αλλά δίνουν τη δυνατότητα στο χρήστη να αποδεχτεί το πιστοποιητικό και να προχωρήσει με τη σύνδεση. Λόγω μη εξοικείωσης της πλειοψηφίας των χρηστών με ζητήματα ασφαλείας και με το πως λειτουργεί η υποδομή δημοσίου κλειδιού του διαδικτύου, συνηθίζουν να αγνοούν το σφάλμα και να αποδέχονται το πιστοποιητικό.

- **Compelled certificate creation attack**

  Ακόμα πιο αποτελεσματικές είναι οι επιθέσεις με πλαστά πιστοποιητικά, που έχουν όμως υπογραφεί από μία από τις πολλές Αρχές Πιστοποίησης (CAs) που εμπιστεύονται οι εφαρμογές πελάτη (π.χ. web browsers). Οποιοσδήποτε έχει υπό τον έλεγχό του το ιδιωτικό κλειδί μιας CA, μπορεί να κατασκευάσει και να υπογράψει πιστοποιητικά για οποιοδήποτε domain name τα οποία θα γίνονται αποδεκτά από όλες τις εφαρμογές πελάτη που εμπιστεύονται αυτή την αρχή. Με το πιστοποιητικό αυτό, μπορεί κανείς να πραγματοποιήσει επιθέσεις ενδιάμεσου με τον τρόπο που περιγράφεται στην παραπάνω ενότητα, μόνο που στην περίπτωση αυτή η εφαρμογή πελάτη δεν προειδοποιεί καν τον τελικό χρήστη ότι υπάρχει κάποιο πρόβλημα, αφού τα πιστοποιητικά επαληθεύονται επιτυχώς και

δεν υπάρχει κανένας απλός τρόπος για να εξακριβώσει ο χρήστης ότι η σύνδεσή του δέχεται επίθεση.

Πρόσβαση στα πιστοποιητικά αναγνωρισμένων αρχών πιστοποίησης μπορούν να έχουν:

α) Εργαζόμενοι στις αρχές αυτές, που ενδέχεται να καταχραστούν τα πιστοποιητικά είτε αυτοβούλως, είτε στα πλαίσια της επίσημης πολιτικής της εταιρίας.

β) Επιτιθέμενοι hackers/crackers που εκμεταλλεύονται λάθη και παραλείψεις ασφαλείας της αρχής για να αποκτήσουν πρόσβαση.

γ) Κυβερνήσεις κρατών ή άλλες ισχυρές οντότητες που μπορούν με νόμιμα ή μη μέσα να υποχρεώσουν τις αρχές πιστοποίησης να τους δώσουν την πρόσβαση που επιθυμούν. Μάλιστα, κάποιες αρχές πιστοποίησης προμηθεύουν ανοιχτά κυβερνήσεις κρατών με υπηρεσίες και προϊόντα που διευκολύνουν τις "νόμιμες υποκλοπές" επικοινωνιών των πολίτων τους.

Επιθέσεις αυτού του τύπου είναι γνωστές και φαίνεται πως πραγματοποιούνται εδώ και περισσότερο από μια δεκαετία αλλά δεν είχαν πάρει ιδιαίτερη δημοσιότητα μέχρι τα τέλη του 2009 όταν ο Dan Kaminsky παρουσίασε το "Black Ops of PKI" στο 26ο Chaos Communication Congress, και λίγους μήνες αργότερα οι Christopher Soghoian και Sid Stamm εξέδωσαν το άρθρο "Certified Lies: Detecting and Defeating Government Interception Attacks Against SSL". Πλέον, όλο και περισσότεροι ερευνητές ασφαλείας τονίζουν τα προβλήματα της υποδομής δημοσίου κλειδιού του διαδικτύου και τη δυνατότητα πραγματοποίησης επιθέσεων ενδιάμεσου που δίνει σε οποιονδήποτε έχει την εξουσία να εξαναγκάσει ή τα χρήματα για να εξαγοράσει οποιαδήποτε αναγνωρισμένη Αρχή Πιστοποίησης.

Ο Dan Kaminsky περιγράφει αυτή την πλευρά της αλυσίδας εμπιστοσύνης των Αρχών Πιστοποίησης ως εξής: "*Μπορείς να πας σε μια Αρχή Πιστοποίησης, να τους δώσεις ένα μάτσο χρήματα και να πάρεις ένα root certificate, αρκεί να υποσχεθείς ότι θα είσαι καλός. Δεν είναι ακριβό, δεν είναι δύσκολο και υπάρχει ένας άγνωστος αριθμός εταιριών εκεί έξω - όχι μόνο οι αρχές πιστοποίησης, αλλά και όλες οι*

εταιρίες που έχουν ενδιάμεσα πιστοποιητικά - όλοι αυτοί μπορούν να εκδώσουν πιστοποιητικά για το domain σου."

Υπάρχουν αρκετά περιστατικά διαρροής ή κατάχρησης πιστοποιητικών CA που έγιναν δημοσίως γνωστά όπως:

- Το Μάρτιο του 2011, η Comodo, η δεύτερη μεγαλύτερή Αρχή Πιστοποίησης που έχει πιστοποιήσει ¼ με ⅕ όλων των domain names στο διαδίκτυο, παραβιάστηκε από έναν Ιρανό hacker 21 ετών σύμφωνα με το κείμενο που δημοσίευσε ο ίδιος. Σαν αποτέλεσμα παρήχθησαν 9 πλαστά πιστοποιητικά για τα domains mail.google.com, www.google.com, login.yahoo.com, login.skype.com, login.live.com & addons.mozilla.com. Μέσα στον ίδιο μήνα έγιναν άλλες δύο παραβιάσεις στην Comodo και ένα μήνα μετά άλλη μία.
- Τον Απρίλη του 2011, ο ίδιος hacker παραβίασε την Diginotar, μια ολλανδική Αρχή Πιστοποίησης. Κατασκευάστηκε άγνωστος αριθμός πιστοποιητικών για σειρά από domain names, ανάμεσα στα οποία και το google.com. Σύμφωνα με έρευνα της Trend Micro, τα πιστοποιητικά εντοπίστηκαν σε περισσότερα από 40 δίκτυα Ιρανικών ISP's και χρησιμοποιήθηκαν για πραγματοποίηση επιθέσεων ενδιάμεσου σε Ιρανούς πολίτες, πιθανότατα από την Ιρανική κυβέρνηση. Παρεμπιπτόντως, το Ιράν είναι μία από τις λίγες χώρες που δεν έχει υπό τον έλεγχό της κάποια αναγνωρισμένη Αρχή Πιστοποίησης.
- Τον Μάιο του 2011, το EFF αποκάλυψε αναφορές μιας επίθεσης ενδιάμεσου στο Facebook.com από την κυβέρνηση της Συρίας, εν μέσω κοινωνικών αναταραχών. Η επίθεση χρησιμοποιούσε αυτο-υπογεγγραμένο πιστοποιητικό οπότε έγινε εύκολα αντιληπτή.
- Τον Σεπτέμβριο του 2011, ο hacker της Comodo ανέλαβε την ευθύνη για παραβίαση και της Αρχής Πιστοποίησης GlobalSign. Η εταιρεία επιβεβαίωσε ότι υπήρξε επίθεση αλλά αρνήθηκε ότι διέρρευσαν πιστοποιητικά.
- Τον Φεβρουάριο του 2012, η Trustwave παραδέχτηκε ότι εξέδωσε πιστοποιητικό για εταιρεία που της επιτρέπει να πραγματοποιεί επιθέσεις ενδιάμεσου σε συνδέσεις TLS τρίτων. Μετά από επικρίσεις το πιστοποιητικό ανεκλήθη και η Trustwave εξέδωσε ανακοίνωση ότι θα σταματήσει να εκδίδει αντίστοιχα πιστοποιητικά.

- **Short Chosen-Prefix collisions for MD5 and the creation of a rogue CA Certificate**

  Τον Δεκέμβριο του 2008, οι Alexander Sotirov, Marc Stevens, Jacob Appelbaum, Arjen Lenstra, David Molnar, Dag Arne Osvik, Benne de Weger δημοσίευσαν μία επίθεση (Sotirov 2008:4) [38] που αξιοποιεί αδυναμία του MD5 και μέσω της οποίας κατάφεραν να δημιουργήσουν ένα πιστοποιητικό μιας πλαστής αρχής πιστοποίησης, μέσω του οποίου μπορούσαν να πραγματοποιήσουν επιθέσεις ενδιάμεσου.

  Ανακάλυψαν ότι ένα από τα πιστοποιητικά της RapidSSL (έχει εξαγοραστεί πλέον από τη Verisign) χρησιμοποιούσε τον αλγόριθμο κατακερματισμού MD5. Νωρίτερα την ίδια χρονιά οι Stevens & Lenstra δημοσίευσαν μία Chosen Prefix Collision Attack για το MD5 η οποία δίνει τη δυνατότητα κατασκευής μιας συμβολοσειράς X τέτοια ώστε, δεδομένων δυο γνωστών προθεμάτων A & B, να ισχύει ότι MD5(A+X) == MD5(B+X). Αυτή την επίθεση χρησιμοποίησαν οι Sotirov, Stevens κ.α. προκειμένου να δημιουργήσουν δυο πιστοποιητικά για διαφορετικά ονόματα και δημόσια κλειδιά υποκειμένου, αλλά με ολόιδια υπογραφή MD5. Με τον τρόπο αυτό κατάφεραν να πείσουν την RapidSSL να υπογράψει ουσιαστικά και τα δυο πιστοποιητικά, ένα από τα οποία είχε φαινομενικά έγκυρα στοιχεία εισόδου, ενώ το δεύτερο είχε ένα πλαστό όνομα και το πεδίο Basic Constraints να ορίζει ότι το συγκεκριμένο πιστοποιητικό ανήκει σε CA. Με υπογεγραμμένο πλέον το πιστοποιητικό αυτό, οι συντάκτες μπορούσαν να υπογράψουν οποιοδήποτε πιστοποιητικό και να πραγματοποιήσουν επιθέσεις ενδιάμεσου σε συνδέσεις TLS.

- **OSCP attack**

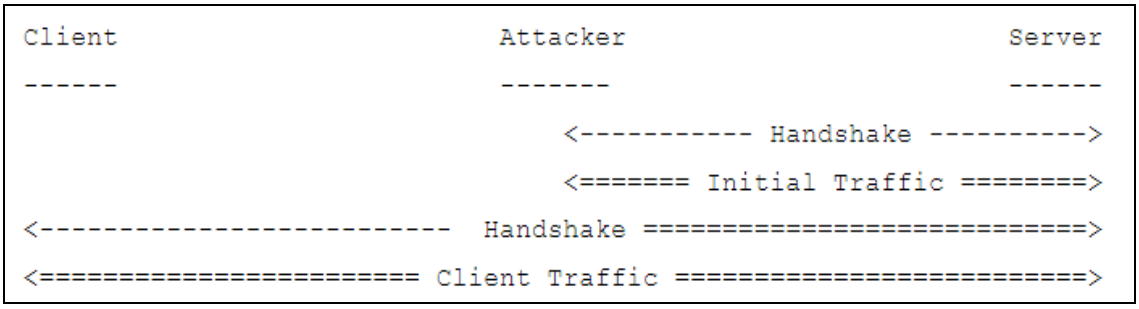
  Το πρωτόκολλο OSCP αναπτύχθηκε για να υποστηριχθεί καλυτέρα η ανάγκη ανάκλησης πιστοποιητικών. Καθώς όλο και περισσότερα πιστοποιητικά πρέπει να ανακληθούν, δεν είναι πλέον αποτελεσματικές οι παλιότερες μέθοδοι της διατήρησης μεγάλων λιστών ανάκλησης πιστοποιητικών που πρέπει να ενημερώνονται χειροκίνητα για κάθε Αρχή Πιστοποίησης. Το OSCP υποστηρίζει αιτήσεις από εφαρμογές πελάτη που θέλουν να ελέγξουν την εγκυρότητα κάποιου πιστοποιητικού. Ο Mozilla Firefox για παράδειγμα, όταν του παρουσιαστεί ένα πιστοποιητικό για το www.ebay.com που δεν έχει δει πρόσφατα, πριν το αποδεχτεί θα κάνει μια γρήγορη σύνδεση στον εκδότη του

πιστοποιητικού για να ρωτήσει αν το πιστοποιητικό θα πρέπει να θεωρείται ακόμα έγκυρο. Ο Moxie Marlinspike έδειξε ότι αν καταρρίψουμε την αίτηση αυτή και επιστρέψουμε μια απάντηση λάθους στα πλαίσια του πρωτοκόλλου OSCP με την τιμή 3 που αντιστοιχεί στο σήμα tryLater, οι περισσότερες υλοποιήσεις δεν θα εμφανίσουν κάποια ένδειξη σφάλματος στο χρήστη και θα προχωρήσουν με τη σύνδεση.

- **Renegotiation attack**

  Το TLS επιτρέπει την επαναδιαπραγμάτευση των κλειδιών κρυπτογράφησης σε ήδη κρυπτογραφημένες συνδέσεις. Οι Ray & Dispensa (Ray 2010:1) [39] εντόπισαν μια αδυναμία σε υλοποιήσεις του TLS που επιτρέπουν σε έναν επιτιθέμενο να εισάγει τα δικά του δεδομένα στην κρυπτογραφημένη σύνδεση κατά τη διάρκεια μιας τέτοιας επαναδιαπραγμάτευσης.

  Ο επιτιθέμενος δεν μπορεί να αποκρυπτογραφήσει ούτε τις αιτήσεις του πελάτη, ούτε τις απαντήσεις του εξυπηρετητή, μπορεί όμως να παρεμβάλλει τα δικά του δεδομένα προς τον εξυπηρετητή. Η επαναδιαπραγμάτευση γίνεται στα πλαίσια υπάρχουσας σύνδεσης TLS, με τα πακέτα χειραψίας να κρυπτογραφούνται μαζί με τα πακέτα εφαρμογής, χωρίς όμως να συνδέονται με άλλο τρόπο με το συγκεκριμένο κανάλι. Το γεγονός αυτό δίνει ένα παράθυρο στον επιτιθέμενο.



Σχήμα 24. Η απλούστερη μορφή επίθεσης Renegotiation

Ο επιτιθέμενος συνδέεται καταρχήν με τον εξυπηρετητή όπου μπορεί να στείλει έναν αυθαίρετο αριθμό αιτήσεων. Η κίνηση αυτή είναι κρυπτογραφημένη και σημειώνεται ως == στο παραπάνω σχήμα. Όταν είναι έτοιμος, παρεμβάλλεται στη σύνδεση μπροστά από τον πελάτη και προωθεί τα πακέτα στον εξυπηρετητή. Ο πελάτης πραγματοποιεί χειραψία με τον εξυπηρετητή και από το σημείο αυτό επικοινωνούν απευθείας. Η αρχική χειραψία φτάνει στον

επιτιθέμενο χωρίς κρυπτογράφηση, αλλά η δεύτερη χειραψία είναι κρυπτογραφημένη και περνάει από το κανάλι του επιτιθέμενου. Ο πελάτης δεν γνωρίζει ότι γίνεται επαναδιαπραγμάτευση, ενώ ο εξυπηρετητής θεωρεί πως η αρχική επικοινωνία με τον επιτιθέμενο προέρχεται επίσης από τον πελάτη. Για την επίθεση αυτή έχει κυκλοφορήσει κώδικας σε Python που την υλοποιεί (redteam 2016:1) [40].

Η απλούστερη λύση αποφυγής τέτοιων επιθέσεων είναι η απενεργοποίηση της υποστήριξης επαναδιαπραγμάτευσης στο επίπεδο του TLS στον εξυπηρετητή, πράγμα που δεν αποτελεί πρόβλημα για την πλειοψηφία των εφαρμογών. Ο πελάτης από την άλλη μεριά δεν μπορεί να πάρει κάποιο μέσο προστασίας, πέρα απ' το να ελέγχει πριν συνδεθεί το κατά πόσο ο εξυπηρετητής επιτρέπει επαναδιαπραγμάτευση, και στην περίπτωση που το επιτρέπει να μην προχωρά με τη σύνδεση. Μια καλύτερη λύση στο επίπεδο του πρωτοκόλλου που δεν προϋποθέτει την απενεργοποίηση της δυνατότητας επαναδιαπραγμάτευσης, περιγράφεται στο RFC 5746. Ουσιαστικά προβλέπει ότι κάθε χειραψία επαναδιαπραγμάτευσης θα πρέπει να περιλαμβάνει πληροφορίες από την προηγουμένη χειραψία.

- **Compression Attacks: CRIME, POODLE, and BREACH**

  Η επίθεση CRIME (CVE-2012-4929) επιτρέπει στον επιτιθέμενο να αποκρυπτογραφήσει κείμενο (συνήθως cookies) όταν το TLS χρησιμοποιείται με TLS-level συμπίεση.

  Η ευπάθεια του SSL 3.0 την οποία εκμεταλλεύεται η επίθεση POODLE πηγάζει από τον τρόπο που τα μπλοκ δεδομένων κρυπτογραφούνται κάτω από ένα συγκεκριμένο τύπο αλγορίθμου κρυπτογράφησης μέσα στο πρωτόκολλο SSL. Η επίθεση POODLE εκμεταλλεύεται την δυνατότητα διαπραγμάτευσης του πρωτοκόλλου σε SSL / TLS για να αναγκάσει τη χρήση του SSL 3.0 και στη συνέχεια αξιοποιεί αυτή τη νέα ευπάθεια για να αποκρυπτογραφήσει επιλεγμένο περιεχόμενο εντός της συνεδρίας SSL. Η αποκρυπτογράφηση γίνεται byte με byte και δημιουργεί ένα μεγάλο αριθμό συνδέσεων μεταξύ του πελάτη και του διακομιστή.

Ενώ το SSL 3.0 είναι ένα παλιό πρότυπο κρυπτογράφησης και γενικά έχει αντικατασταθεί από το TLS, οι περισσότερες SSL / TLS υλοποιήσεις παραμένουν συμβατές με το SSL 3.0 για να λειτουργεί με παλαιότερα συστήματα. Ακόμα κι αν ο πελάτης και ο διακομιστής υποστηρίζουν μια έκδοση του TLS, το SSL/TLS πρωτόκολλο επιτρέπει την διαπραγμάτευση του ποια έκδοση του πρωτοκόλλου να χρησιμοποιηθεί (αναφέρεται ως "χορός υποβάθμισης-downgrade dance" σε άλλες εκθέσεις). Η επίθεση POODLE αξιοποιεί το γεγονός ότι, όταν μια ασφαλής προσπάθεια σύνδεσης αποτύχει, οι servers θα μεταπέσουν σε παλαιότερα πρωτόκολλα όπως το SSL 3.0. Ένας εισβολέας που μπορεί να προκαλέσει μια αποτυχία σύνδεσης μπορεί στη συνέχεια να αναγκάσει τη χρήση του SSL 3.0 και θα επιχειρήσει τη νέα επίθεση.

Δύο άλλες προϋποθέσεις πρέπει να πληρούνται για να εκτελεστεί με επιτυχία η επίθεση POODLE:

1ον) ο εισβολέας θα πρέπει να είναι σε θέση να ελέγχει τμήματα στην πλευρά του client της σύνδεσης SSL (μεταβάλλοντας το μήκος της εισόδου) και

2ον) ο εισβολέας πρέπει να έχει πρόσβαση στο κρυπτογραφημένο κείμενο που δημιουργείται. Ο πιο συνηθισμένος τρόπος για να επιτευχθούν αυτοί οι όροι θα ήταν να ενεργεί ως MiTM, κάτι που απαιτεί μια ξεχωριστή μορφή επίθεσης για να δημιουργήσει αυτό το επίπεδο πρόσβασης. Η ανακάλυψη της επίθεσης έγινε από τους Bodo Möller, Thai Duong και Krzysztof Kotowicz από την ομάδα ασφάλειας της Google στις 14 Οκτωβρίου 2014 και στις 8 Δεκεμβρίου 2014, ανακοινώθηκε παραλλαγή της ευπάθειας POODLE.

- **Κλοπή ιδιωτικών κλειδιών της RSA**

  Η χρήση TLS χωρίς την παράλληλη χρήση Diffie-Hellman ανταλλαγής κλειδιών κρυπτογράφησης είναι αρκετή για να ληφθεί το ιδιωτικό κλειδί του διακομιστή για την αποκρυπτογράφηση κάθε συνεδρίας (τόσο παρελθοντικής όσο και μελλοντικής) που ξεκίνησαν με αυτόν το server. Η τεχνική αυτή χρησιμοποιείται, για παράδειγμα, από το Wireshark για να επιθεωρήσει την προστασία TLS συνδέσεων.

Είναι γνωστό ότι έχουν κλαπεί (ή άλλο οικειοποιηθεί) ιδιωτικά κλειδιά και έχουν χρησιμοποιηθεί ως μέρος παρακολούθησης –σε μεγάλη κλίμακα- [RFC7258] ορισμένων servers.

Τέτοιες επιθέσεις μπορούν να μετριαστούν με την καλύτερη προστασία του ιδιωτικού κλειδιού, π.χ., χρησιμοποιώντας προστασίες OS ή αποκλειστικό (dedicated) hardware. Ακόμα πιο αποτελεσματική είναι η χρήση κρυπτογράφησης που προσφέρει "προς τα εμπρός απόρρητο", ώστε το κλειδί που αποκαλύπτει ένα μυστικό, δεν εκθέτει παρελθοντικές ή μελλοντικές συνεδρίες σε ένα παθητικό εισβολέα.

- **Παράμετροι Diffie-Hellman**

  Το TLS επιτρέπει τον ορισμό εφήμερων παραμέτρων Diffie-Hellman (DH) και Ελλειπτικών καμπυλών Diffie-Hellman στην ανταλλαγή κλειδιών. Αυτό οδηγεί σε μια επίθεση CROSS PROTOCOL (Mavrogiannopoulos 2010:5) [41]. Η χρήση προκαθορισμένων ομάδων DH, όπως προτείνεται στην [FFDHE-TLS], θα μετριάσει αυτήν την επίθεση.

  Επιπλέον, τα προγράμματα που δεν ελέγχουν σωστά τις λαμβανόμενες παράμετρούς εκτίθενται σε MiTM επιθέσεις. Δυστυχώς, το πρωτόκολλο TLS δεν περιλαμβάνει εντολή τέτοιας επαλήθευσης (Βλέπε [RFC6989] για ανάλογες πληροφορίες για IPsec).

- **Triple Handshake**

  Η επίθεση τριπλής χειραψία επιτρέπει στον εισβολέα να δημιουργήσει δύο συνδέσεις TLS για να μοιραστεί το υλικό που πληκτρολογείται κάτι που οδηγεί σε πληθώρα επιθέσεων, όπως MiTM, σπάζοντας την ασφαλή επαναδιαπραγμάτευση, και σπάσιμο του εμπιστευτικού καναλιού TLS που δημιουργείται [RFC5705] ή "μοναδικού TLS " [RFC5929].

- **Virtual Host Confusion**

  Πρόσφατο άρθρο [Delignat14] περιγράφει ένα ζήτημα ασφαλείας όπου το SSLv3 fallback και η ακατάλληλη χρήση session caches στην πλευρά του server μπορεί να επιτρέψει σε έναν εισβολέα να δημιουργήσει κακόβουλη σύνδεση σε ένα

εικονικό host διαφορετικό από εκείνο που είχε αρχικά προβλεφθεί και εγκριθεί από το διακομιστή. Η επίθεση αυτή είναι ιδιαίτερα σοβαρή στην περιβάλλοντα όπου η απόδοση είναι κρίσιμη και η ανταλλαγή session caches SSLv3 είναι πολύ κοινή.

- **Denial of Service**

  Η επεξεργαστική ισχύς της CPU έχει αυξηθεί με τα χρόνια, ώστε το TLS μπορεί να είναι ενεργοποιημένο από προεπιλογή. Ωστόσο, ο κίνδυνος των κακόβουλων πελατών και συντονισμένων ομάδων ("botnets") που στοχεύουν σε denial-of-service επιθέσεις εξακολουθεί να είναι πραγματική. Το TLS προσθέτει άλλο ένα "πάτημα" για computational επιθέσεις δεδομένου ότι ένας πελάτης μπορεί εύκολα (με λίγη υπολογιστική προσπάθεια) να αναγκάσει το διακομιστή να δαπανήσει σχετικά μεγάλη επεξεργαστική ισχύ. Είναι γνωστό ότι έχουν γίνει τέτοιες επιθέσεις.

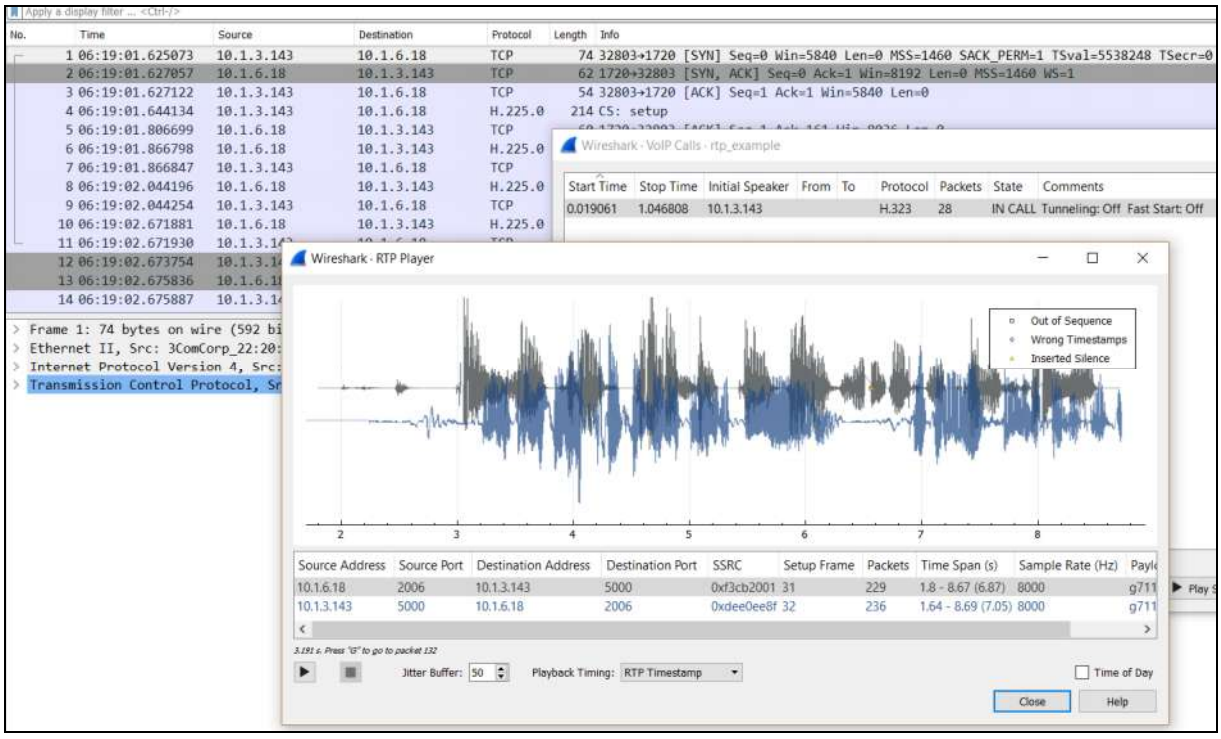
- **Cain & Abel** (oxid.it 2012:1)[45]

  Το πρόγραμμα χρησιμοποιείται για την ανάλυση και την αντιμετώπιση προβλημάτων WLAN και είναι σε θέση να συλλέξει passwords. Έχει πλούσια υποστήριξη για διάφορα πρωτόκολλα WLAN, αλλά και πρωτόκολλα LAN και πλέον υποστηρίζει VoIP επικοινωνίες. Μπορεί να κάνει επίθεση MiTM στο SIPS κάνοντας sniffing την πόρτα 5061, ανακατευθύνοντας την διεύθυνση DNS ενός SIP server, στον εαυτό του και κατόπιν στέλνει τα δεδομένα στο σωστό server. Έτσι μπορεί να συγκεντρώσει τους κωδικούς που περνούνε διαμέσου του. Για να αντιμετωπιστεί πρέπει να γίνεται πάντα χρήση κατάλληλων υπογεγραμμένων πιστοποιητικών στους servers και ο πελάτης VoIP (UA) δεν πρέπει να δέχεται με ευκολία συνδέσεις από SIP TLS servers όταν δεν μπορεί να επαληθευτεί το πιστοποιητικό.

Κεφάλαιο 5
Ασφάλεια στο LinPhone

5.1 "Σύλληψη" και αναπαραγωγή

Το Wireshark είναι σε θέση να κάνει "σύλληψη" (capturing) αλλά και αναπαραγωγή ήχου πλέον μόνο του χωρίς τη χρήση εξωτερικών προγραμμάτων.



Εικόνα 4. Σύλληψη & αναπαραγωγή ήχου με το wireshark

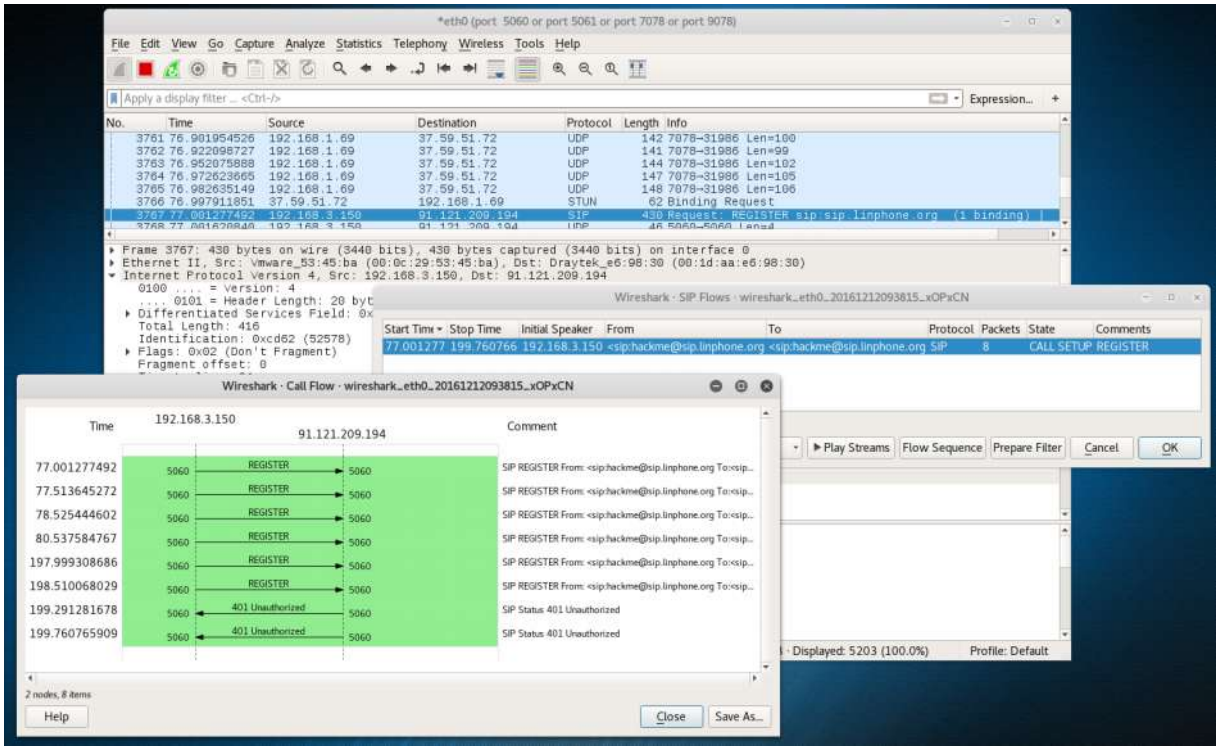
Το capturing και το playback μπορεί να γίνει βέβαια σε απλή μετάδοση SIP, RTP χωρίς κρυπτογράφηση. Επίσης δεν υποστηρίζονται όλοι οι codecs ήχου. Είναι λοιπόν προφανές ότι μία απλή VoIP συνομιλία μπορεί να υποκλαπεί πάρα πολύ εύκολα.

Στο διαδίκτυο κυκλοφορεί πληθώρα εργαλείων που είναι σε θέση να εξάγουν ήχο από μία ροή δεδομένων όπως το παράδειγμα στη σελίδα https://github.com/wbwarnerb/ExercisesFor Python3 book όπου ο δημιουργός του, Brian Warner εκμεταλλεύεται το

PyShark το οποίο είναι ένα python wrapper του tshark δηλαδή της έκδοση γραμμής εντολών του wireshark, για να εξάγει ήχο από το captured αρχείο μίας συνδιάλεξης.

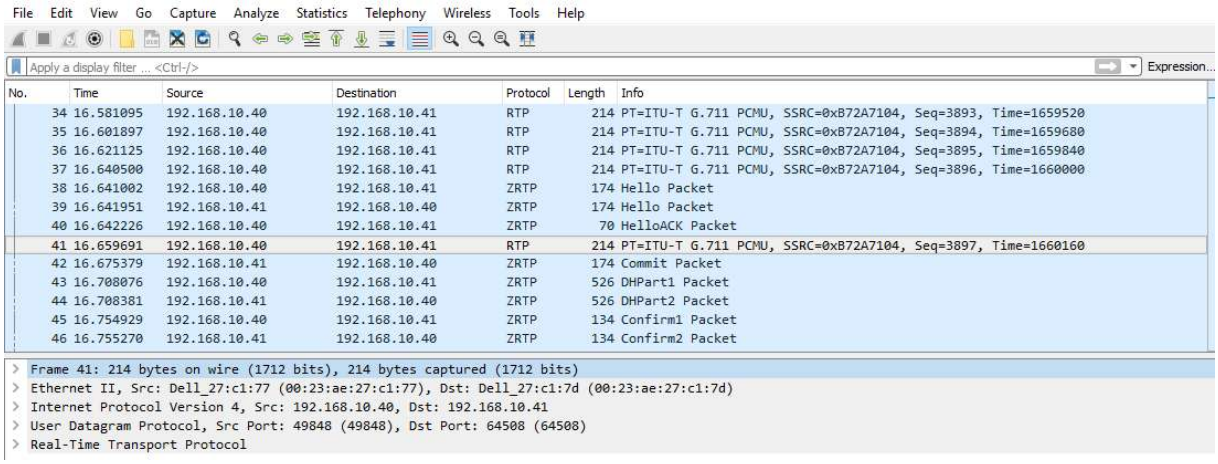
5.2 "Σύλληψη" & ανίχνευση κίνησης

Αν θέλουμε να χρησιμοποιήσουμε το Wireshark για να συλλέξουμε την VoIP κίνηση η οποία κρυπτογραφείται με ZRTP, δεν θα μπορούμε να το κάνουμε σε μηχάνημα που τρέχει το LinPhone αν έχει λειτουργικό Windows. Αυτό συμβαίνει διότι στα Windows ο driver του ZRTP είναι πιο κοντά στο υλικό από ότι ο driver του Winpcap τον οποίο χρησιμοποιεί το Wireshark και άλλα παρόμοια προγράμματα και έτσι μπορούν να συλλέξουν μόνο μη κρυπτογραφημένη IP κίνηση. Το LinPhone έχει αποκρυπτογραφήσει τα εισερχόμενα πακέτα πριν μπορέσει το Wireshark να τα δει, και κρυπτογραφεί τα εξερχόμενα πακέτα αφού έχουν περάσει από το Wireshark. Έτσι θα πρέπει η έρευνα μας να γίνεται σε εικονική μηχανή που τρέχει Linux: το Kali Linux προσφέρεται, λόγω των εργαλείων που διαθέτει.



Εικόνα 5. Capturing VoIP με TLS και ZRTP.

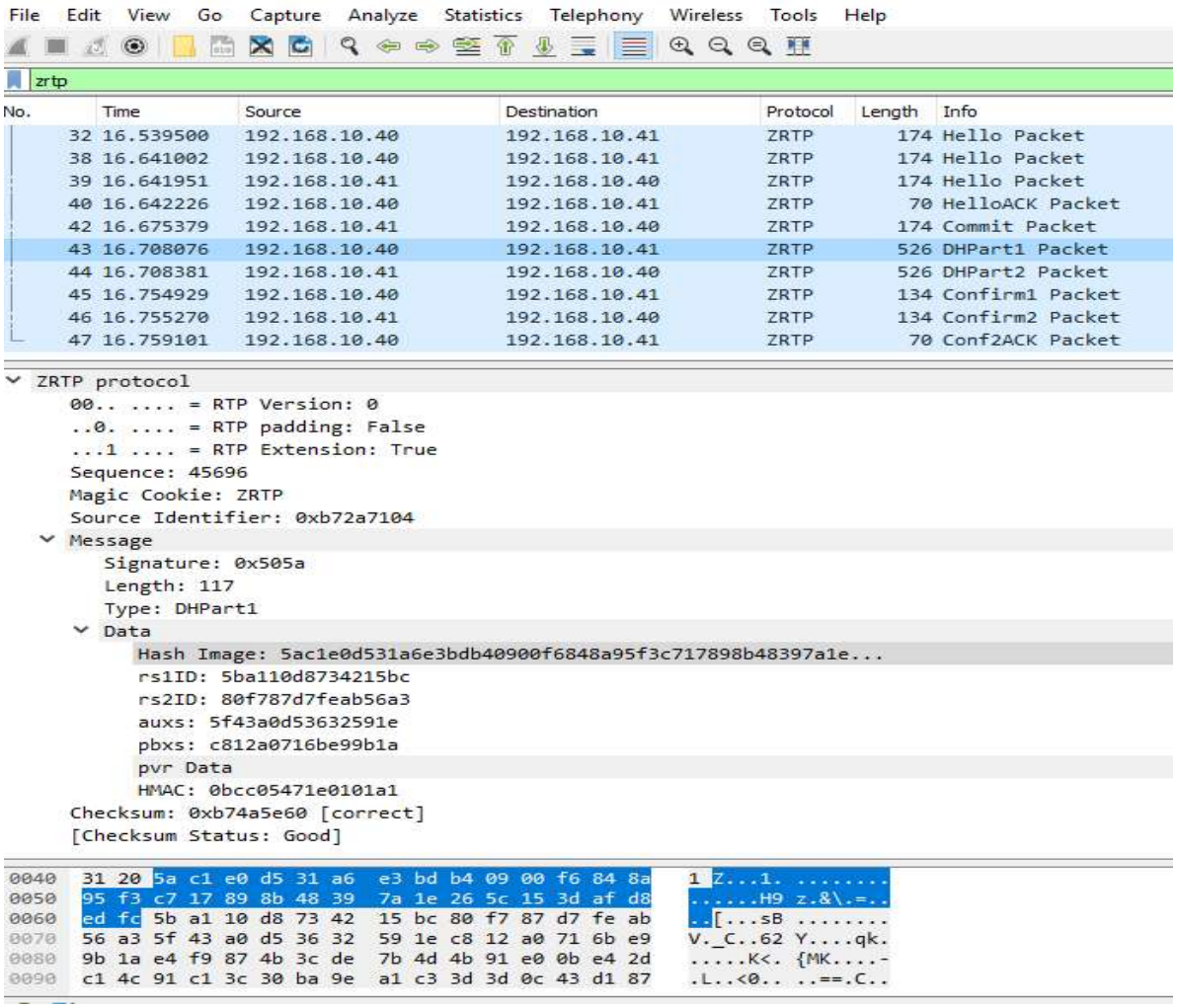
Στην εικόνα 5 φαίνεται ότι με ενεργοποιημένες όλες τις ασφάλειες του Linphone, τα πακέτα που έχουν συλλεχθεί είναι μη αξιοποιήσιμα. Χαρακτηριστικό είναι ότι δεν φαίνονται ούτε τα πακέτα ήχου RTP.



Εικόνα 6. Κλήση VoIP με ZRTP στο Kali Linux

Χωρίς την παρουσία του TLS μπορούμε να "ανιχνεύσουμε" την ZRTP επικοινωνία όπως φαίνεται στην εικόνα 6.

Χρησιμοποιώντας φίλτρα όπως zrtp, zrtp.cc, zrtp.cipher κ.λπ., μπορούμε να απομονώσουμε την αναφορά στη zrtp συνδιαλλαγή.



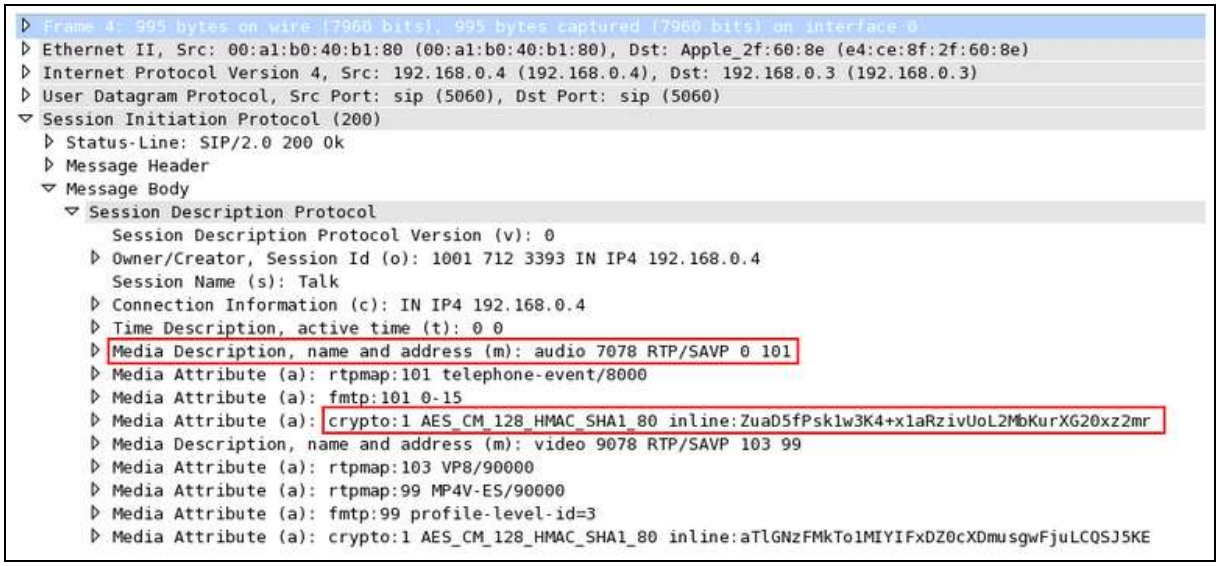
Εικόνα 7. Η ZRTP συνδιαλλαγή

Βέβαια το αποτέλεσμα είναι ίδιο διότι δεν μπορούμε να κάνουμε και πολλά πράγματα με τις πληροφορίες που συλλέξαμε.

Συμπερασματικά βλέπουμε ότι οι σηματοδοσίες μέσω UDP ή TCP χωρίς TLS και η μετάδοση των media streams χωρίς ZRTP, είναι πολύ εύκολο να υποκλαπούν και να αναπαραχθούν. Ακόμη και στην SRTP μετάδοση είναι πολύ εύκολο να υποκλαπούν τα στοιχεία όπως φαίνεται ξεκάθαρα στην ιστοσελίδα acrittelli.com[47], όπου ο συγγραφέας δείχνει σε απλά βήματα πως μπορεί:

- Να "συλλάβει" μία ολόκληρη συνομιλία,
- Να πάρει το κλειδί και να φιλτράρει μία SRTP ροή δεδομένων.
- Να χρησιμοποιήσει ένα software, το "srtp-decrypt" ώστε να αποκρυπτογραφήσει το SRTP και τέλος
- Να αναπαράγει όλη τη ροή ήχου από το αποκρυπτογραφημένο SRTP

Αναφέρει μάλιστα ότι θα χρησιμοποιήσει το LinPhone διότι η εφαρμογή αυτή έχει το κακό συνήθειο να αποστέλλει τα κλειδιά κρυπτογράφησης ως απλό κείμενο.



Εικόνα 8. Τα κλειδιά κρυπτογράφησης SRTP απεσταλμένα από το LinPhone

Ως προς την κρυπτογράφηση της επικοινωνίας στο πλαίσιο του SRTP πρωτοκόλλου, όπως φαίνεται και στην Εικόνα 8, χρησιμοποιείται ο αλγόριθμος AES.

5.3 Lan κλήσεις

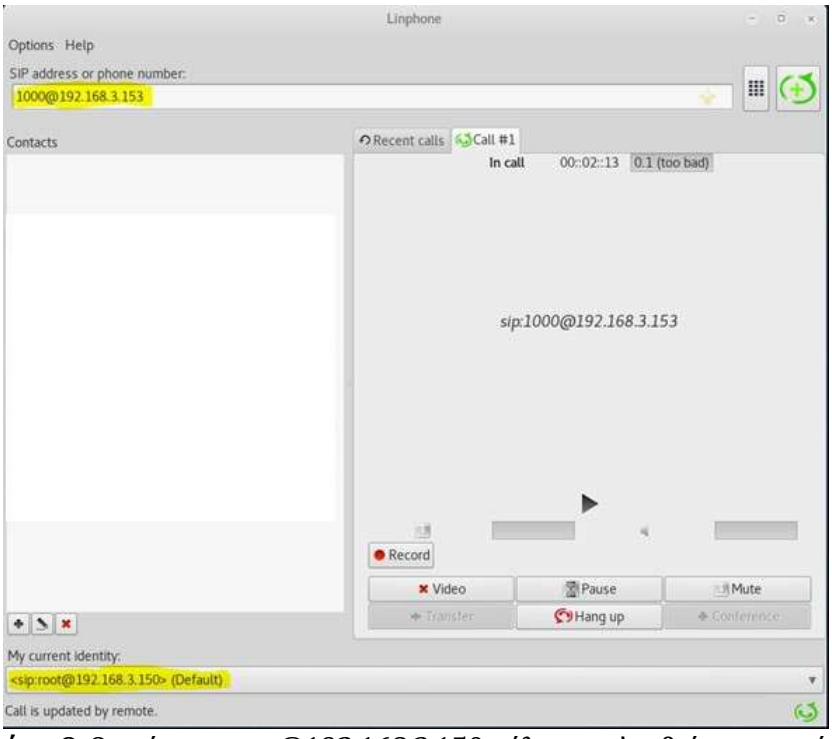
Για τις ανάγκες της διατριβής, τον Νοέμβριο του 2016 και για χρονικό διάστημα δύο εβδομάδων, ομάδα 25 χρηστών εγκατέστησαν και λειτουργούσαν συνεχώς σε εργαστήριο δικτύων την εφαρμογή Linphone - κάποιοι δε εξ' αυτών στα κινητά τους - και ανέφεραν τις παρατηρήσεις τους ώστε να εξαχθούν συμπεράσματα.

Ο εξοπλισμός περιλάμβανε:

- δεκαπέντε υπολογιστές με Pentium 4 και windows XP,
- πέντε με dual core και windows 7 home edition,
- έναν με dual core και OpenSuse Leap και
- τέσσερα κατά μέγιστο smartphones

Η σύνδεση στο Διαδίκτυο γινόταν χρησιμοποιώντας γραμμή με 4 Mbps μέγιστη ταχύτητα download. Επιλέχθηκαν υπολογιστές παλαιότερης τεχνολογίας για να ελέγξουμε την απόδοση και την λειτουργία της εφαρμογής σε περιβάλλον χαμηλών προδιαγραφών.

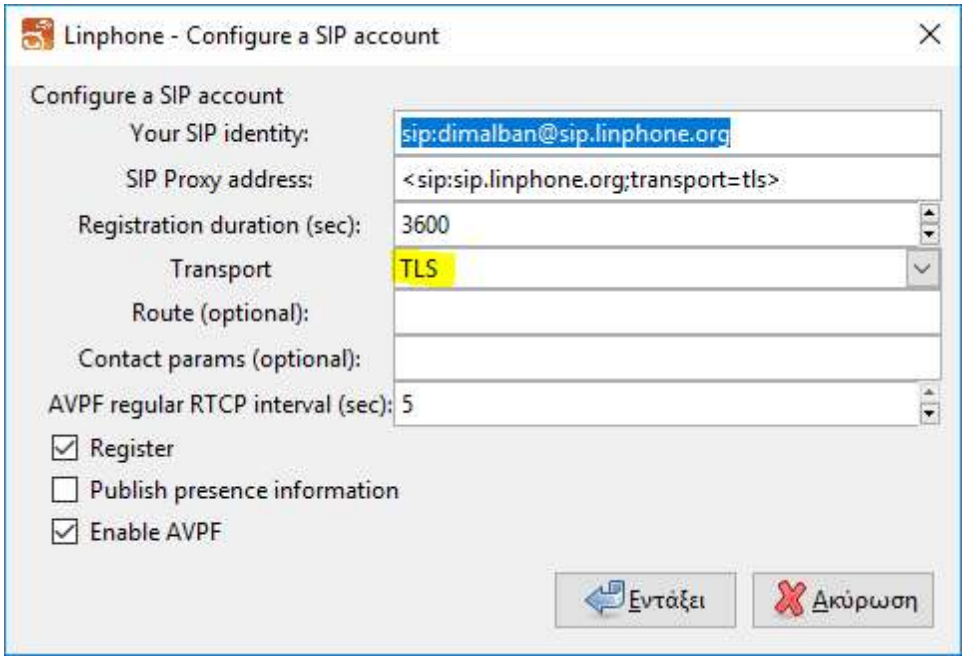
Το LinPhone όπως και όλα τα VoIP προγράμματα που βασίζονται στην τεχνολογία SIP, δεν χρειάζονται proxy servers για να συνδεθούν, δηλαδή μπορούν να πραγματοποιήσουν κλήση αν καλέσουμε απευθείας την IP διεύθυνση του άλλου άκρου.



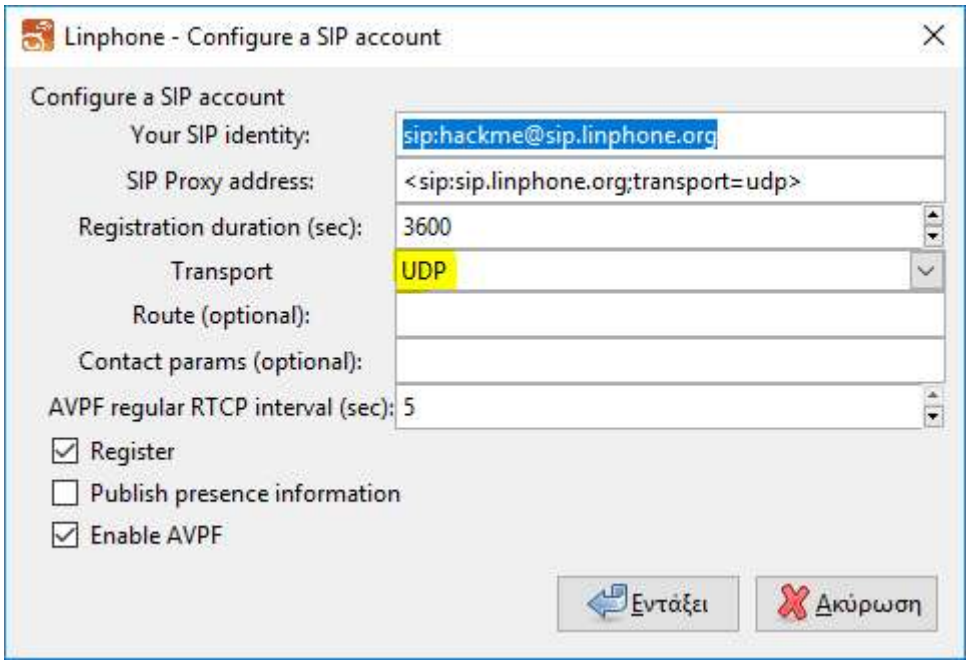
Εικόνα 9. Ο χρήστης root@192.168.3.150 κάλεσε απ' ευθείας τον χρήστη 1000@192.168.3.153 χωρίς τη χρήση proxy.

Η δυνατότητα αυτή υπάρχει για να μπορούμε να καλούμε κάποιον μέσα στο ίδιο δίκτυο χωρίς να διέλθει η κλήση μας από κάποιον proxy server.

Επίσης διαπιστώθηκε ότι αν ο καλών έχει ενεργοποιημένο TLS και ο καλούμενος έχει TCP ή UDP στην αντίστοιχη ρύθμιση, τότε η κλήση δεν επιτυγχάνεται παρόλο που δίνεται η αίσθηση ότι πραγματοποιείται, δηλαδή ο καλών βλέπει και ακούει την κλήση να πραγματοποιείται ενώ στο άλλο άκρο δεν συμβαίνει τίποτα.



Εικόνα 10. Ο καλών έχει ενεργοποιημένο το TLS

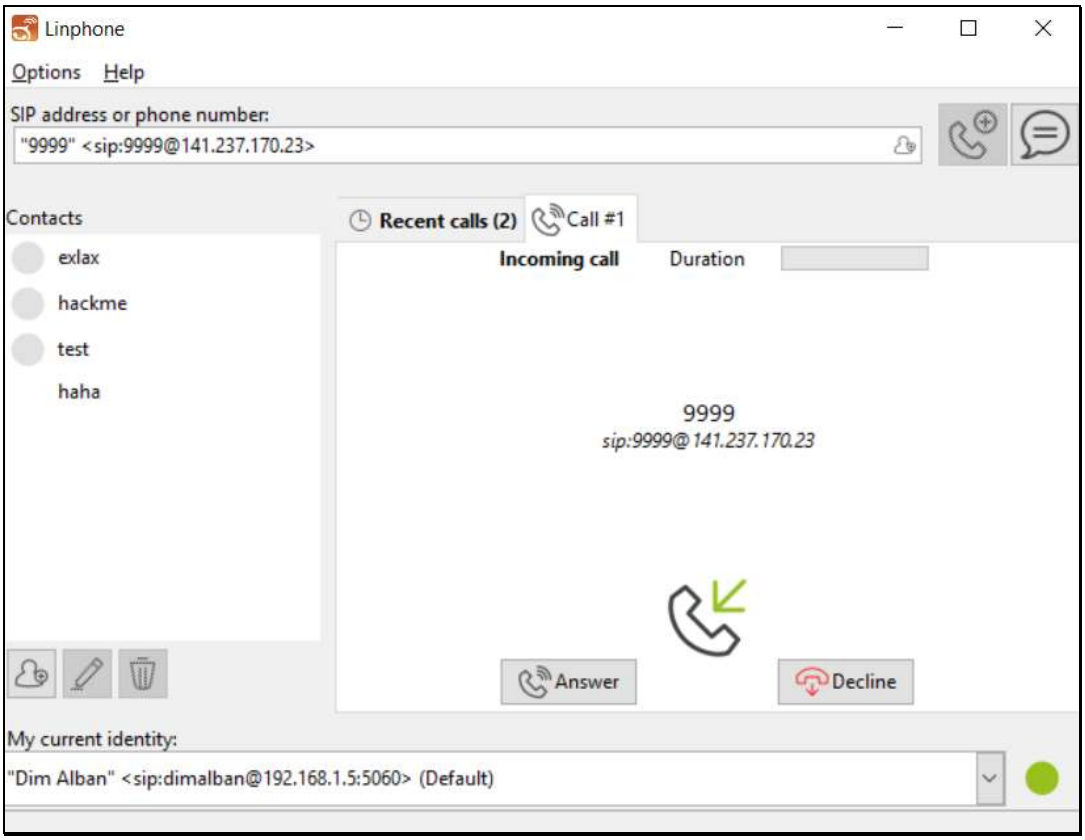


Εικόνα 11. Ο καλούμενος έχει ενεργοποιημένο UDP

Το ενδιαφέρον όμως είναι ότι, όπως διαπιστώθηκε, το αντίστροφο λειτουργεί, δηλαδή η UDP/TCP πλευρά μπορεί να καλέσει την TLS πλευρά. Από την επισκόπηση του κώδικα της εφαρμογής αλλά και από τα αποτελέσματα του wireshark φαίνεται ότι η κλήση γίνεται με ιεραρχία, δηλαδή η UDP κλήση "κατεβάζει" την άλλη πλευρά στο επίπεδό της ακόμη και αν εκείνη είναι ρυθμισμένη σε υψηλότερο επίπεδο ασφαλείας (TLS). Αυτό αποτελεί αναμφίβολα ένα σημαντικό ζήτημα ασφάλειας, αφού είθισται οι χρήστες που έχουν προ-ρυθμισμένο το TLS να θεωρούν ότι το πρωτόκολλο πάντα θα είναι σε λειτουργία για τις επικοινωνίες τους.

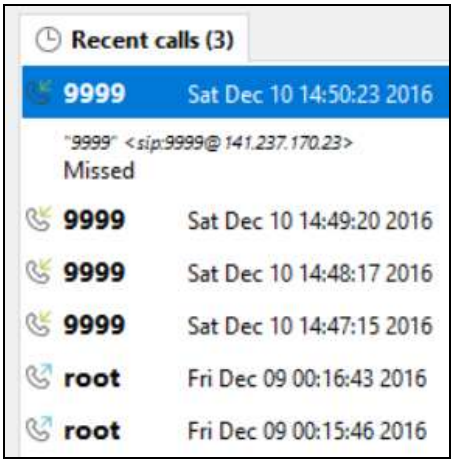
5.4 Απρόσκλητος επισκέπτης

Κατά τη διαδικασία ελέγχου και ανάλυσης της ροής δεδομένων αυτών των κλήσεων με το wireshark (κατά κύριο λόγο), η εφαρμογή δέχτηκε μία παράξενη κλήση από το τηλέφωνο 9999@141.237.170.23



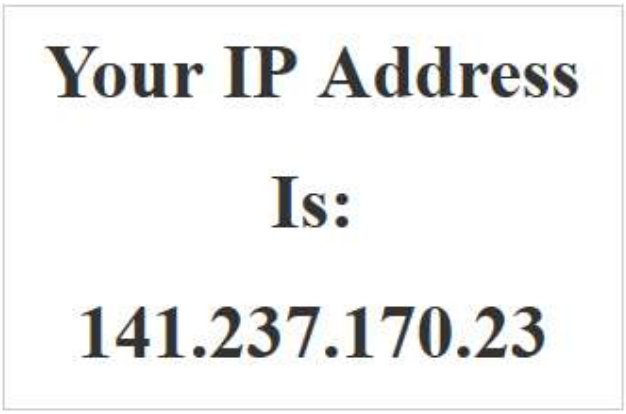
Εικόνα 12. Μία παράξενη εισερχόμενη κλήση.

Η κλήση αυτή δεν ήταν μοναδική αλλά άρχισε να επαναλαμβάνεται κάθε ένα λεπτό περίπου.

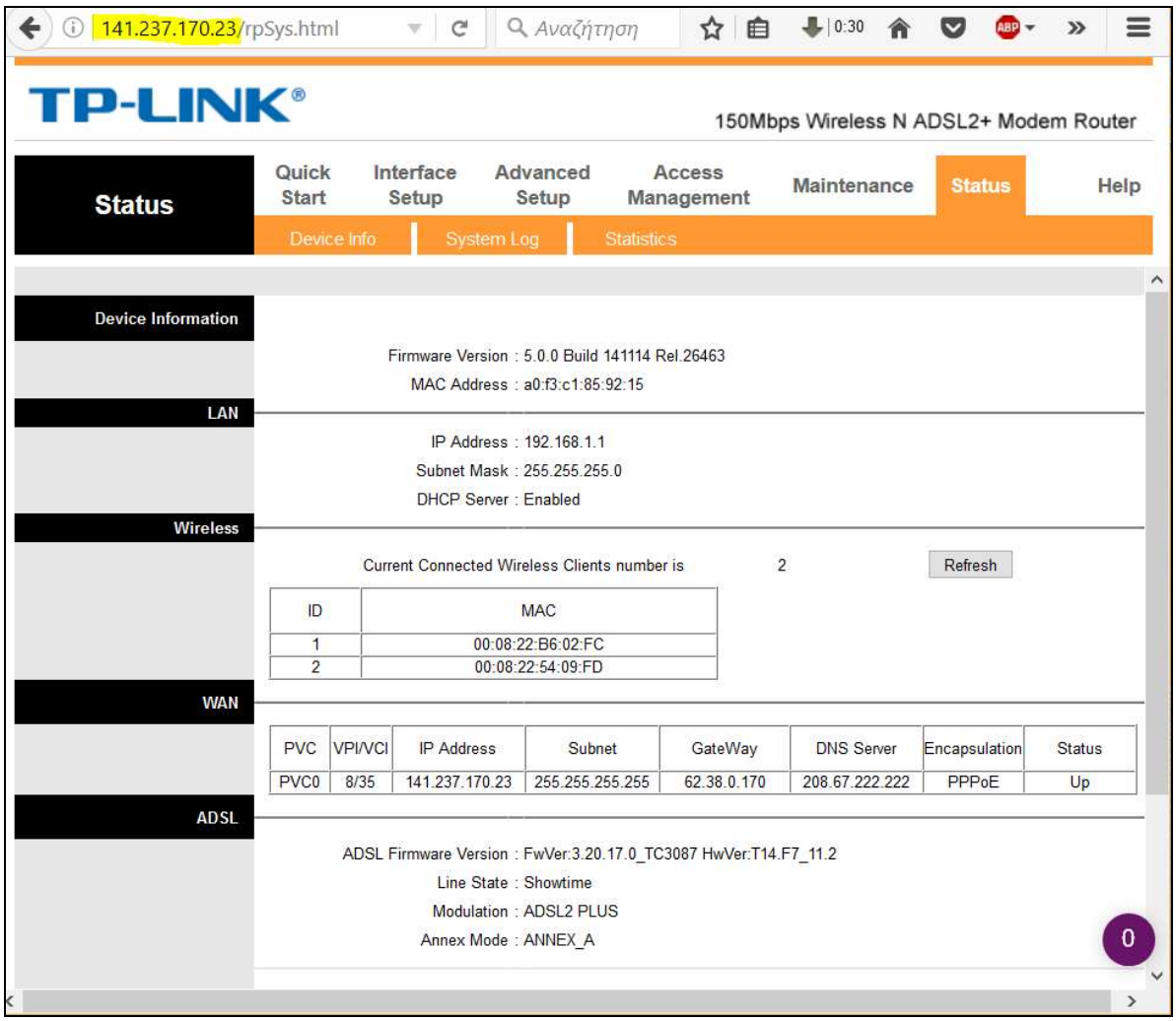


Εικόνα 13. Η ανά ένα λεπτό επανάληψη της "παράξενης" κλήσης.

Η IP διεύθυνση 141.237.170.23 η οποία φαινόταν στο όνομα του καλούντος, φάνηκε αρκετά γνωστή και ένα άνοιγμα της σελίδας WhatIsMyIP.com, απέδειξε ότι δεν θα μπορούσε να είναι περισσότερο γνωστή καθώς ήταν η εξωτερική IP διεύθυνση του δικτύου μας (δηλαδή η IP διεύθυνση του δρομολογητή μας).

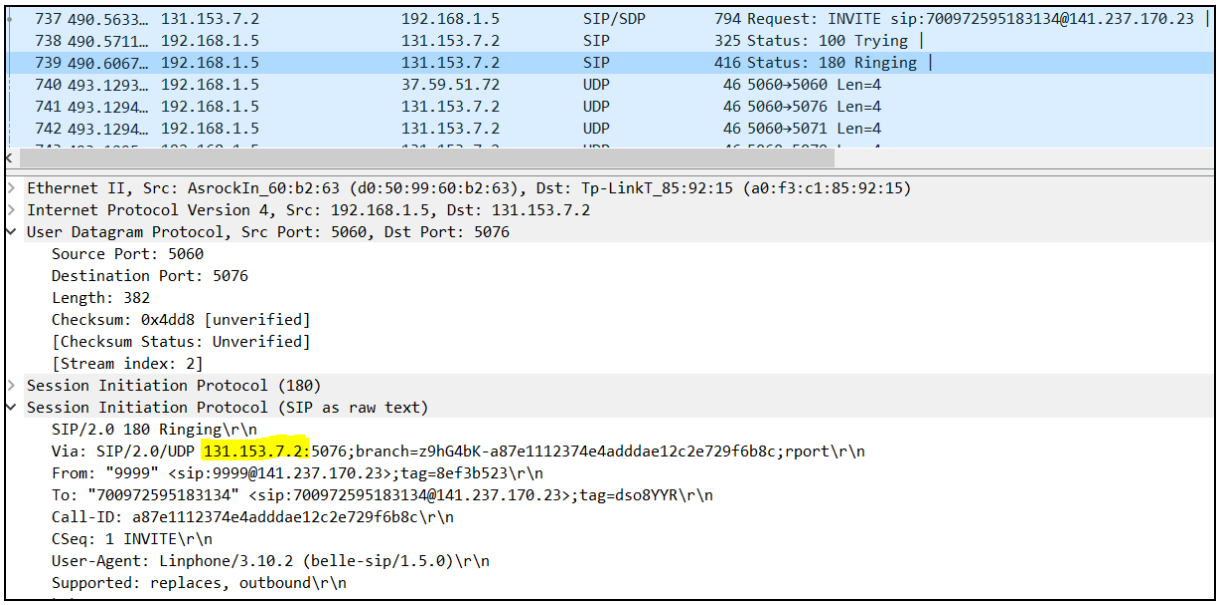


Εικόνα 14. Το WhatIsMyIP.com δείχνει ότι καλούμε τον εαυτό μας επανειλημμένα.!!!



Εικόνα 15. Ο Firefox επαληθεύει ότι εμείς (ή μάλλον ο router μας) καλούμε τον εαυτό μας

Επόμενο βήμα είναι να βρούμε ποιος είναι ο επιτιθέμενος. Με το wireshark κάνοντας capture την πόρτα 5060 “συλλάβαμε” τον αδιάκριτο.

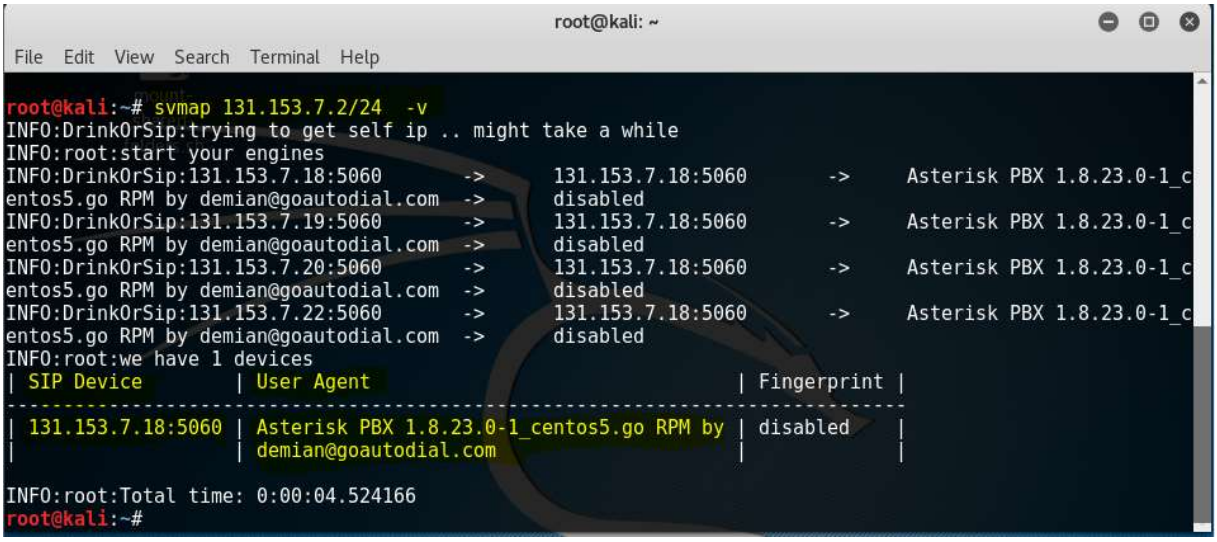


Εικόνα 16. Caprturing της πόρτας 5060 με το WireShark

Το καλών alter ego μας λοιπόν ξεκινάει από την 131.153.7.2 διεύθυνση. Με τη βοήθεια του SipVicious (Gauci 2016:1)[43] το οποίο εγκαταστήσαμε στο Kali Linux 2 στο πλαίσιο της έρευνάς μας με αποκλειστικό σκοπό να μελετήσουμε τη συγκεκριμένη επίθεση και εκτελώντας το svmap το οποίο μπορεί να εμφανίσει όλες τις SIP συσκευές μέσα σε ένα δοθέν IP range

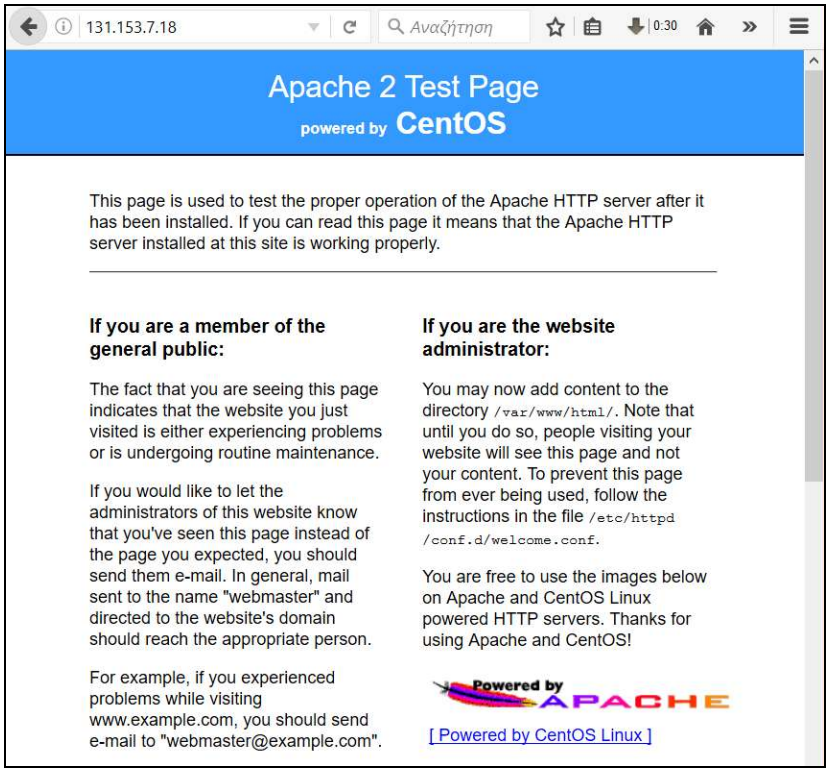
svmap 131.153.7.2/24 –v

βρήκαμε ότι ο καλών είναι ένα Asterisk PBX με λειτουργικό centos5 και IP διεύθυνση 131.153.7.18



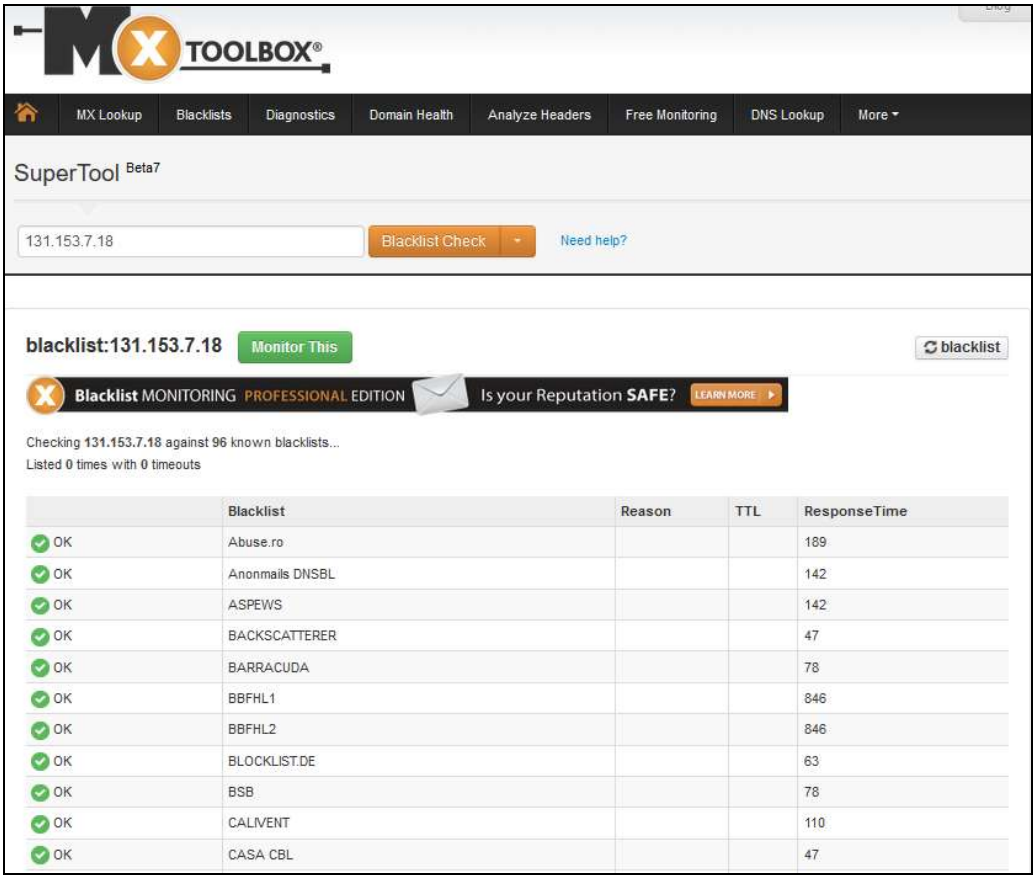
Εικόνα 17. Ανακάλυψη του alter ego μας

Και από τον Firefox είδαμε ότι είναι μία σελίδα ελέγχου Apache2



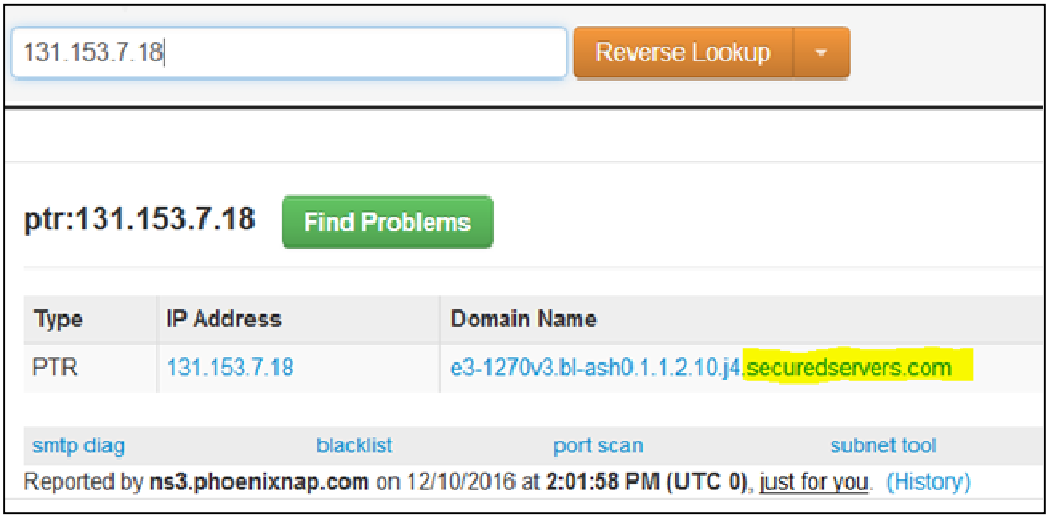
Εικόνα 18. Η σελίδα του εισβολέα

Το mxtoolsbox.com δεν δίνει αρνητικές πληροφορίες για τη σελίδα

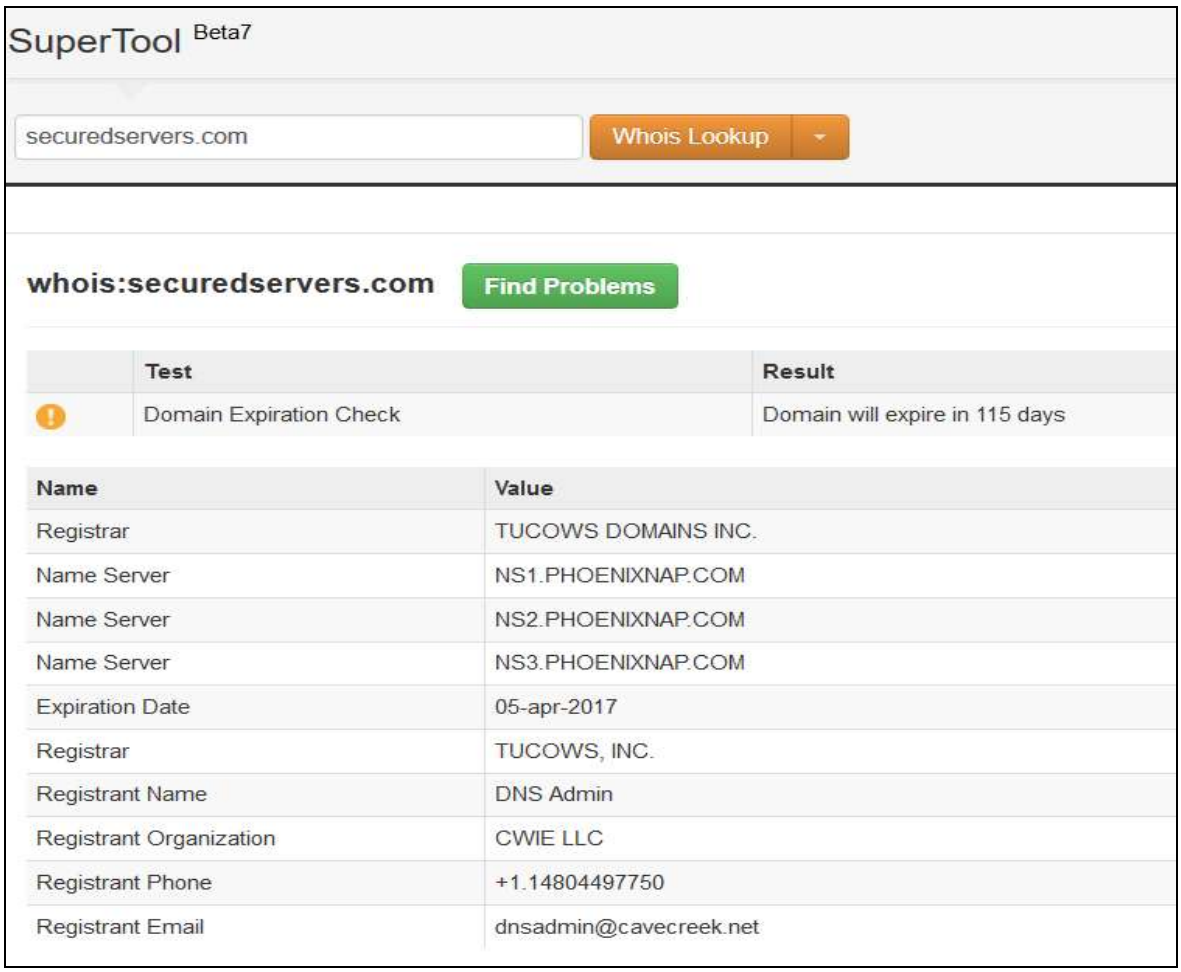


Εικόνα 19. Έλεγχος του εισβολέα στο mxtoolbox.com

Και με reverse lookup βρίσκουμε ότι αντιστοιχεί στη σελίδα securedservers.com η οποία είναι εταιρεία παροχής υπηρεσιών virtual server.



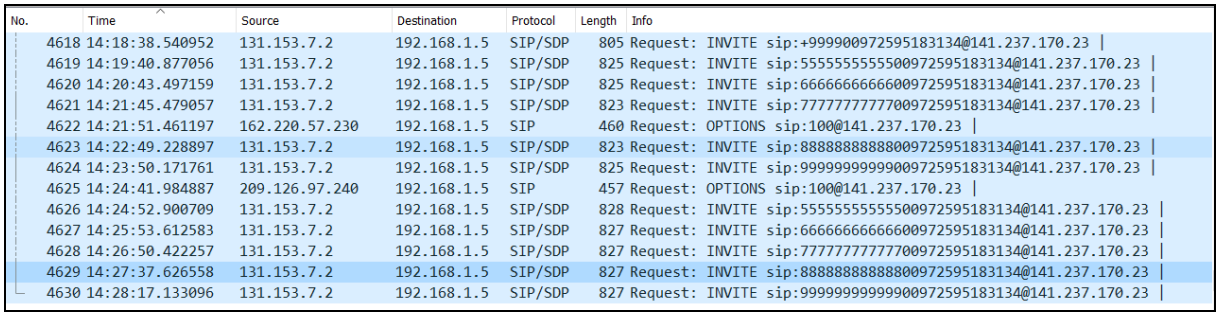
Εικόνα 20. Reverse Lookup του εισβολέα στο mxtoolbox.com



Εικόνα 21. Whois αναζήτηση της σελίδας

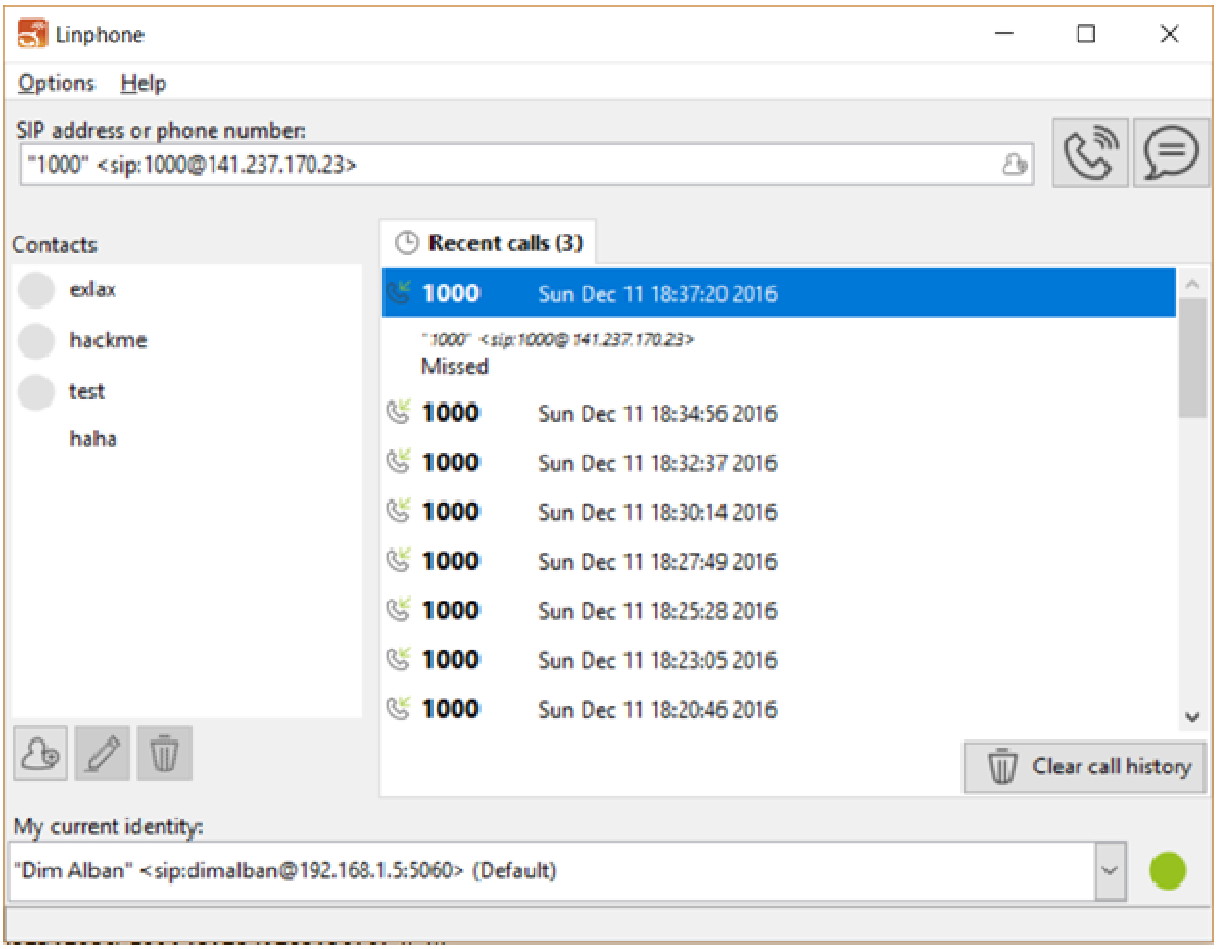
Επομένως η επίθεση που λαμβάνει χώρα, προέρχεται από κάποιον virtual server που έχει υπενοικιάσει η εταιρεία σε άλλους – δεν πραγματοποιείται δηλαδή από την ίδια την εταιρεία. Σε κάθε περίπτωση στείλαμε ενημερωτικό email τόσο στο registrant mail όσο και στο mail επικοινωνίας που αναφέρει η σελίδα.

Μόλις ενεργοποιήσαμε στο LinPhone το TLS, σταμάτησε να αποκρίνεται σε αυτές τις κλήσεις αλλά το Wireshark δείχνει ότι οι προσπάθειες συνεχίζονται ανά ένα λεπτό περίπου.



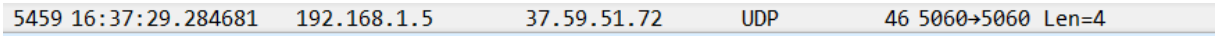
Εικόνα 22. Με ενεργοποιημένο το TLS, σταμάτησε το LinPhone να αποκρίνεται αλλά το Wireshark δείχνει ότι οι κλήσεις εξακολουθούν να υφίσταντο.

Το απόγευμα της επόμενης μέρας άρχισαν πάλι να εκδηλώνονται παρόμοιες κλήσεις αλλά αυτή τη φορά από τον 1000@141.237.170.23

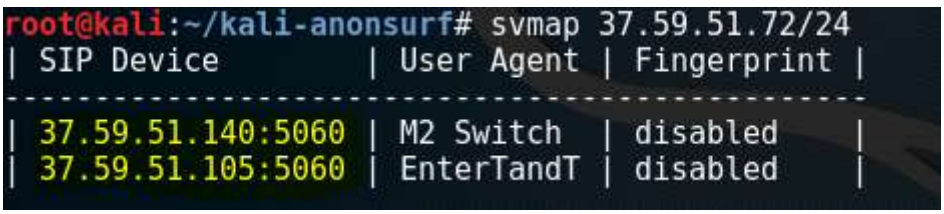


Εικόνα 23. Νέες ενοχλητικές εισερχόμενες κλήσεις την επόμενη ημέρα.

Με το Wireshark



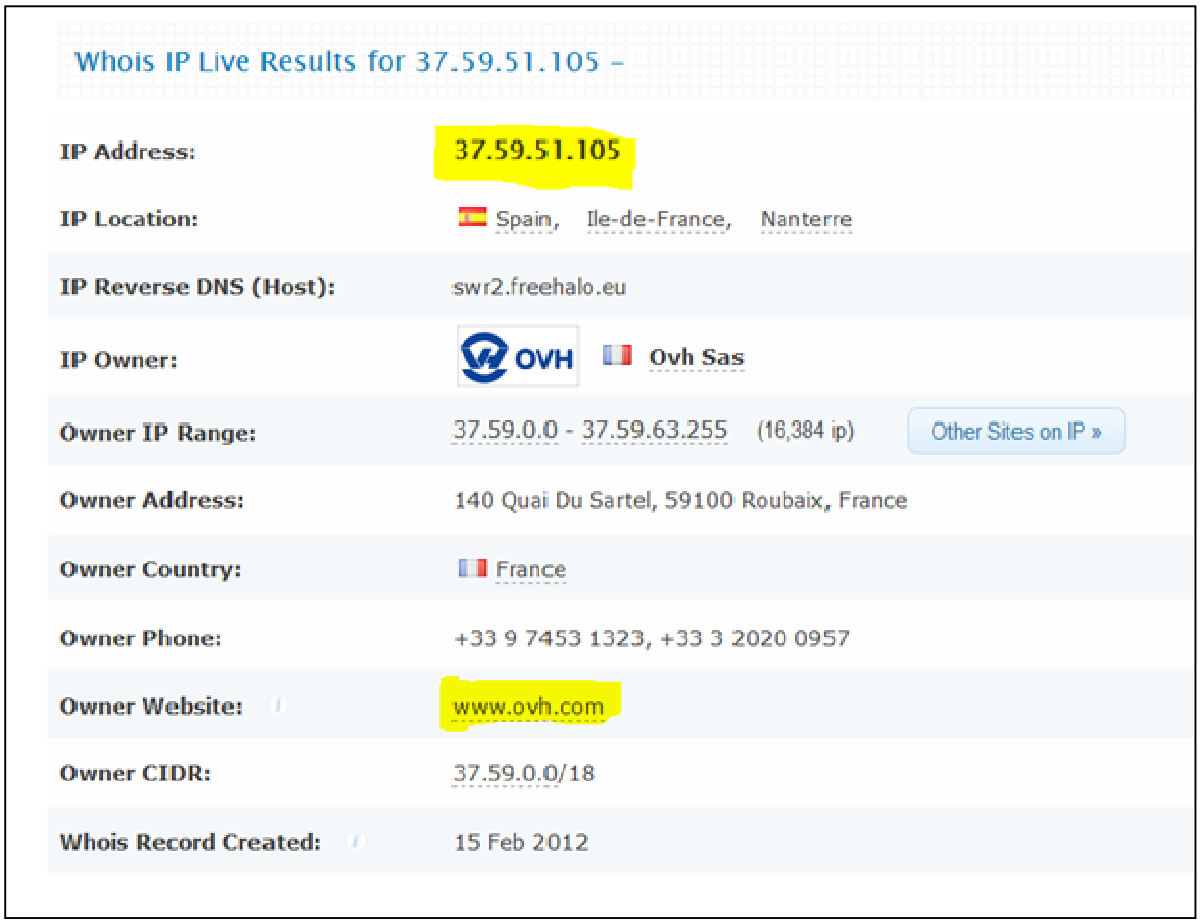
και με την svmap βρήκαμε ότι ο εισβολέας είναι διαφορετικός αυτή τη φορά.



Εικόνα 24. Νέα εισβολή

Εκτελούμε λοιπόν ένα νέο Whois, βρίσκουμε ότι η επίθεση προέρχεται και αυτή τη φορά από virtual server καθότι η εταιρεία που παραπέμπει το Whois η ovh.com είναι

και αυτή εταιρεία παροχής υπηρεσιών cloud hosting – δηλαδή, όπως και πριν, είναι σαφές ότι δεν πραγματοποιεί η συγκεκριμένη εταιρεία κάποια επίθεση..



Εικόνα 25. Web Whois του νέου εισβολέα

Εγκαθιστώντας τη βιβλιοθήκη VipRoy (Ozavci 2016:1) [42], πάντα στο πλαίσιο της έρευνάς μας με αποκλειστικό σκοπό την ανάλυση της συγκεκριμένης επίθεσης ασφαλείας, και βάζοντας τις κατάλληλες ρυθμίσεις στο auxiliary viproy_sip_enumerate, ερευνήσαμε να δούμε αν υπάρχει χρήστης από το 100 – 250 δηλ. (100@37.59.51.105) που να απαντάει στις κλήσεις.

```
msf auxiliary(viproy_sip_enumerate) > show options

Module options (auxiliary/voip/viproy_sip_enumerate):

   Name             Current Setting  Required  Description
   ----             ---------------  --------  -----------
   BRUTEFORCE_SPEED 5                yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS     false            no        Try each user/password couple stored in the current database
   DB_ALL_PASS      false            no        Add all passwords in the current database to the list
   DB_ALL_USERS     false            no        Add all users in the current database to the list
   METHOD           SUBSCRIBE        yes       Method for Enumeration (SUBSCRIBE,REGISTER,INVITE,OPTIONS)
   NUMERIC_MAX      250              yes       Ending extension
   NUMERIC_MIN      100              yes       Starting extension
   NUMERIC_USERS    true             yes       Numeric Username Bruteforcing
   PROTO            UDP              yes       Protocol for SIP service (UDP|TCP|TLS)
   RESPONSEREGEX                     no        Regular expression for responses e.g. ^40[0-3]|^40[5-9]
   RHOSTS           37.59.51.105     yes       The target address range or CIDR identifier
   RPORT            5060             yes       The target port
   STOP_ON_SUCCESS  false            yes       Stop guessing when a credential works for a host
   THREADS          24               yes       The number of concurrent threads
   USER_FILE                         no        File containing usernames, one per line
```

Εικόνα 26. Παραμετροποίηση του viproy_sip_enumerate auxiliary

Ψάχνοντας αν υπάρχει VoIP extension από 100 έως και 250 βρήκαμε ότι όλα είναι υπαρκτά

```
msf auxiliary(viproy_sip_enumerate) > run -j
[*] Auxiliary module running as background job
[+] User 100 is Valid (Server Response: 403 Forbidden (policy))
[+] User 101 is Valid (Server Response: 403 Forbidden (policy))
[+] User 103 is Valid (Server Response: 403 Forbidden (policy))
[+] User 104 is Valid (Server Response: 403 Forbidden (policy))
[+] User 105 is Valid (Server Response: 403 Forbidden (policy))
[+] User 106 is Valid (Server Response: 403 Forbidden (policy))
[+] User 109 is Valid (Server Response: 403 Forbidden (policy))
[+] User 110 is Valid (Server Response: 403 Forbidden (policy))
[+] User 111 is Valid (Server Response: 403 Forbidden (policy))
[+] User 112 is Valid (Server Response: 403 Forbidden (policy))
[+] User 114 is Valid (Server Response: 403 Forbidden (policy))
[+] User 115 is Valid (Server Response: 403 Forbidden (policy))
[+] User 116 is Valid (Server Response: 403 Forbidden (policy))
[+] User 117 is Valid (Server Response: 403 Forbidden (policy))
[+] User 120 is Valid (Server Response: 403 Forbidden (policy))
[+] User 122 is Valid (Server Response: 403 Forbidden (policy))
[+] User 123 is Valid (Server Response: 403 Forbidden (policy))
[+] User 124 is Valid (Server Response: 403 Forbidden (policy))
[+] User 126 is Valid (Server Response: 403 Forbidden (policy))
[+] User 127 is Valid (Server Response: 403 Forbidden (policy))
[+] User 128 is Valid (Server Response: 403 Forbidden (policy))
```

Εικόνα 27. Έλεγχος για υπαρκτά VoIP extensions

Κατόπιν με το auxiliary viproy_sip_register ελέγξαμε ποιοι hosts απαντούν σε όλο το subnet 255.255.255.0

```
msf auxiliary(viproy_sip_register) > show options

Module options (auxiliary/voip/viproy_sip_register):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   FROM      1000             no        The source username to probe at each host
   LOGIN     false            no        Login Using Credentials
   PASSWORD                   no        The login password to probe at each host
   PROTO     UDP              yes       Protocol for SIP service (UDP|TCP|TLS)
   RHOSTS    37.59.51.1-255   yes       The target address range or CIDR identifier
   RPORTS    5060             yes       Port Range (5060-5065)
   THREADS   1                yes       The number of concurrent threads
   TO        1000             no        The destination username to probe at each host
   USERNAME  NOUSER           no        The login username to probe at each host
```

Εικόνα 28. Οι ρυθμίσεις του viproy_sip_register auxiliary στο msf

```
[*] Scanned  26 of 255 hosts (10% complete)
[*] Scanned  51 of 255 hosts (20% complete)
[*] Scanned  77 of 255 hosts (30% complete)
[*] Scanned 102 of 255 hosts (40% complete)
[*] 37.59.51.105:5060
        Response : 401 Unauthorized
        Server           : EnterTandT
        Realm            : sip-x.yourcallback.pl

[*] Scanned 128 of 255 hosts (50% complete)
[*] 37.59.51.140:5060
        Response : 401 Unauthorized
        Server           : M2 Switch
        Realm            : asterisk

[*] Scanned 153 of 255 hosts (60% complete)
```

Εικόνα 29. Εύρεση ενεργών hosts

Μετά από αναζήτηση στο Διαδίκτυο διαπιστώσαμε ότι ο επιτιθέμενος κατά πάσα πιθανότητα χρησιμοποεί την svcrack από το πακέτο SipVicious για να οργανώσει την επίθεσή του (ουσιαστικά, με το εργαλείο αυτό «σκανάρει» το σύστημά σας για τυχόν ευπάθειες): μάλιστα, σε μία συνέντευξή του (Gauci 2012:1)[44] ο Sandro Gauci, δημιουργός του SipVicious, αναφέρει ότι:

"*Το SIPVicious δημιουργήθηκε για να ελέγχει τα PBX συστήματα για πιθανές αδυναμίες. Δεν κάνει καμία κλήση αλλά εκτελεί όλες τις δοκιμές ασφάλειας σιωπηλά. Δεδομένου ότι τα εργαλεία χρησιμοποιούνται από κυβερνο-εγκληματίες, οι προμηθευτές λογισμικού PBX πρέπει να ενημερώσουν το λογισμικό τους ώστε να μην επιτρέπει αυτές τις σαρώσεις.*

Ο λόγος για τον οποίο στοχεύουν τα συστήματα PBX είναι για να εντοπίσουνε σε αυτά, τηλεφωνικές γραμμές με αδύναμο κωδικό πρόσβασης. Στη συνέχεια θα μπορούν να

πραγματοποιήσουν δωρεάν τηλεφωνήματα, μέσα από αυτό, σε διεθνείς αριθμούς ή αριθμούς premium εις βάρος του θύματος. Έτσι θα εκμεταλλευθούν για ιδία χρήση τα PBX ή θα πουλήσουν παροχές διεθνούς τηλεφωνίας σε τρίτους, χωρίς να πληρώσουν οι ίδιοι κανένα αντίτιμο.

Η αποστολή μίας σάρωσης τύπου INVITE σε ένα ευάλωτο σύστημα PBX μπορεί να είναι χρήσιμη για τον χάκερ, όταν όμως αποστέλλεται σε ένα τηλέφωνο IP ή τηλέφωνο VoIP, το κάνει να χτυπάει, να καλεί. Κάποια τηλέφωνα θα χτυπήσουν μόνο όταν τα καλεί εγκεκριμένος αριθμός, όμως κάποια θα χτυπήσουν σε οποιαδήποτε κλήση ή μάλλον από οποιαδήποτε διεύθυνση SIP. Έτσι, οι επιτιθέμενοι / hackers / κυβερνο-εγκληματίες καταλήγουν να κάνουν τα τηλέφωνα να χτυπούν. Νομίζω ότι αυτό είναι λάθος τους, ίσως επειδή δεν διαχωρίζουν ένα τηλέφωνο από ένα σύστημα PBX.

Ο φορέας παροχής υπηρεσιών (δηλαδή ο ISP ή VoIP εταιρεία) θα μπορούσε πιθανώς να αποτρέψει τους εισβολείς από το να φτάνουν στα τηλέφωνα των μελών τους ή θα μπορούσαν να επιτρέπουν κλήσεις μόνο όταν καλείται κατοχυρωμένος αριθμός."

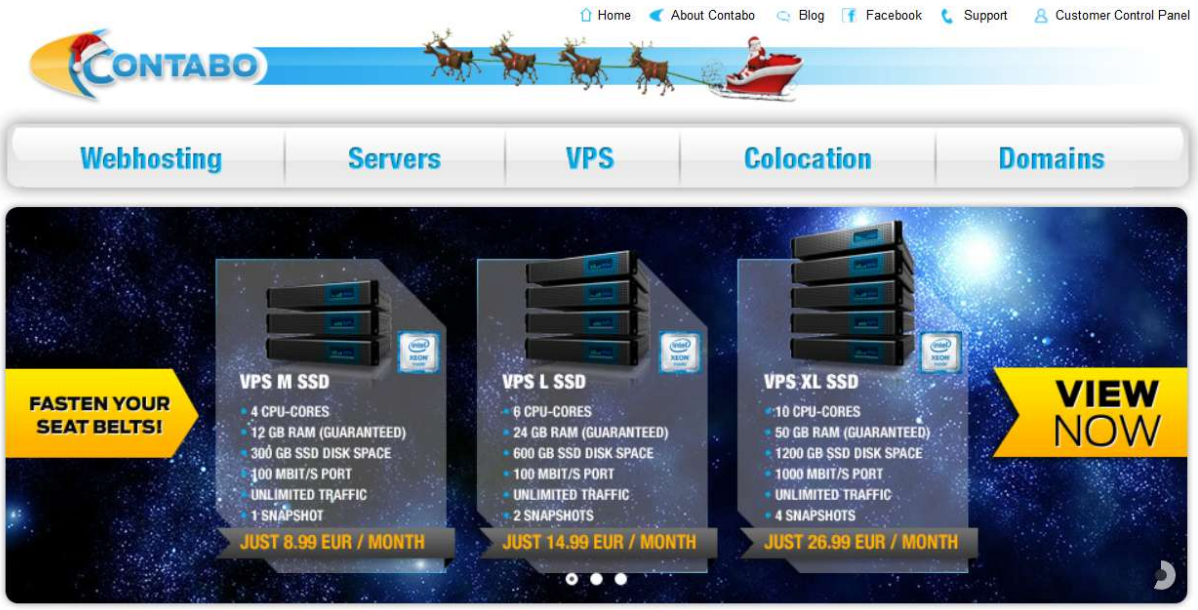
Όπως διαπιστώσαμε, η χρήση TLS στο Linphone σταματάει τις συγκεκριμένες κλήσεις που συναντώνται στη βιβλιογραφία με τον όρο "κλήσεις-φαντάσματα" (ghost calls). Ωστόσο, πέρα από αυτήν την παρατήρησή μας, στο Διαδίκτυο προτείνονται και διάφορες λύσεις στο εν λόγω πρόβλημα όπως η χρήση firewalls με σκοπό τον αποκλεισμό - φραγή των ανεπιθύμητων επισκεπτών που δεν προέρχονται από συγκεκριμένο SIP proxy ή ζητώντας τη βοήθεια του παρόχου αλλά και ο Sandro Gauci στην ιστοσελίδα του (Gauci 2012)[54] δίνει μία λύση για την αντιμετώπιση των "ghost calls" με τη χρήση του SNORT (γνωστού εργαλείου ανίχνευσης εισβολών) λέγοντας:
"Η προστασία του δικτύου από απειλές VoIP είναι μόνο η μισή ιστορία. Το υπόλοιπο μισό αποτελεί η ανίχνευση της επίθεση στο σύστημά σας. Ενα Intrusion Detection System, όπως το Snort μπορεί να ρυθμιστεί για να βοηθήσει σε αυτό. Παραθέτω κάποιους κανόνες (rules) για το Snort με τους οποίους είναι σε θέση να ανιχνεύσει επιθέσεις που δημιουργούνται με εργαλεία όπως το svwar και το svcrack και δημιουργούν ένα μεγάλο αριθμό από αιτήματα INVITE ή REGISTER SIP καθώς και αποκρίσεις SIP "401 Unathorized".

Κεφάλαιο 6
VPN συνδέσεις

Ένας πολύ καλός τρόπος επίτευξης επικοινωνίας VoIP είναι μέσω καναλιών VPN. Κάτι τέτοιο βέβαια δεν μπορεί να γίνει μέσα από τον κώδικα της εφαρμογής καθώς προϋποθέτει την ύπαρξη VPN server σε οποιαδήποτε λειτουργικό σύστημα ή/και μορφή π.χ. OpenVPN σε Ubuntu server ή σε Pfsense ή RAS σε Windows Server κ.λπ.

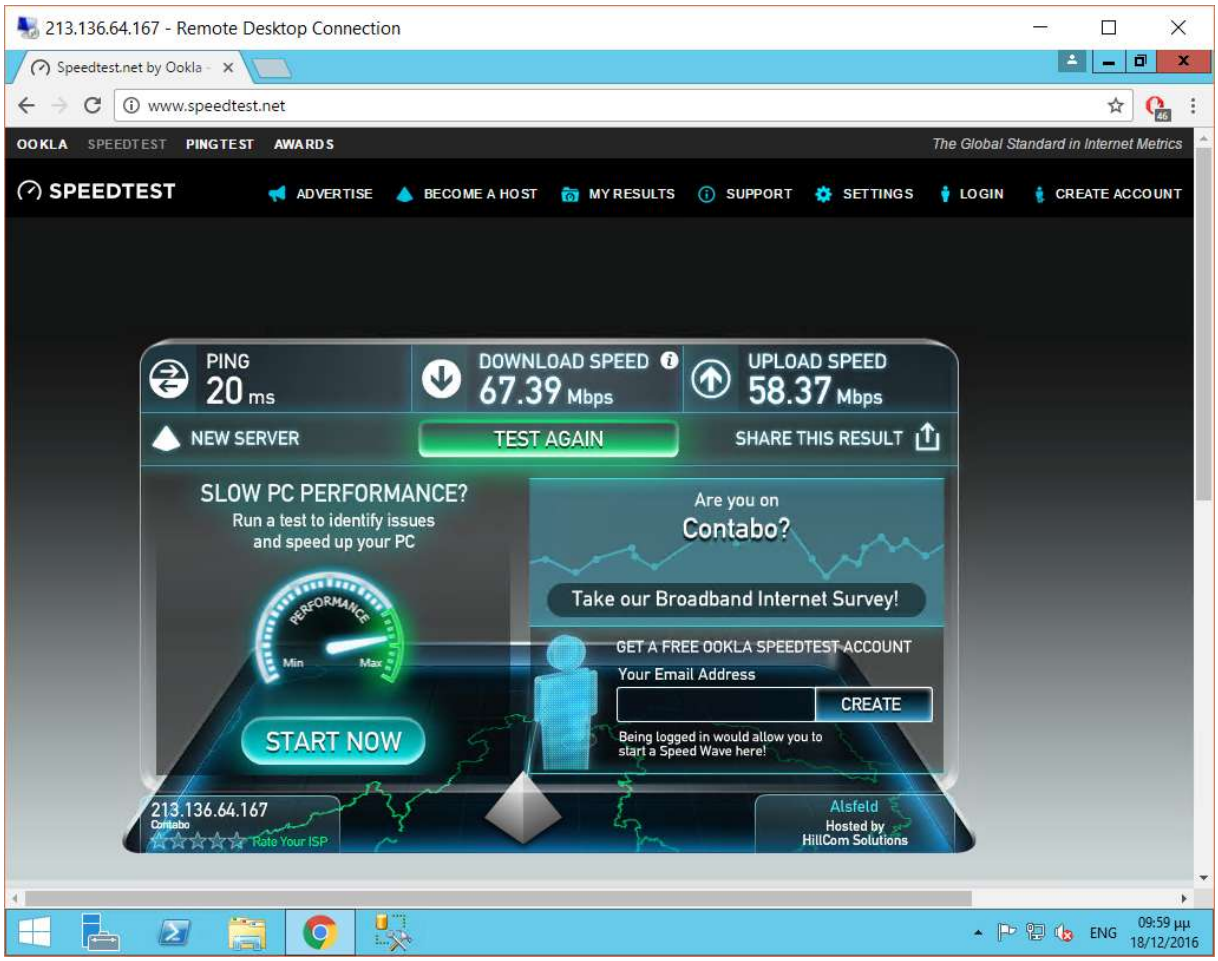
6.1 Εγκατάσταση VPN server και δημιουργία χρηστών σε αυτόν

Για τις ανάγκες της εργασίας, προβήκαμε στην ενοικίαση εικονικής μηχανής στην Contabo.com, μία εταιρεία ενοικίασης Virtual server, χαμηλού κόστους, καλής αξιοπιστίας και έδρα στη Γερμανία.



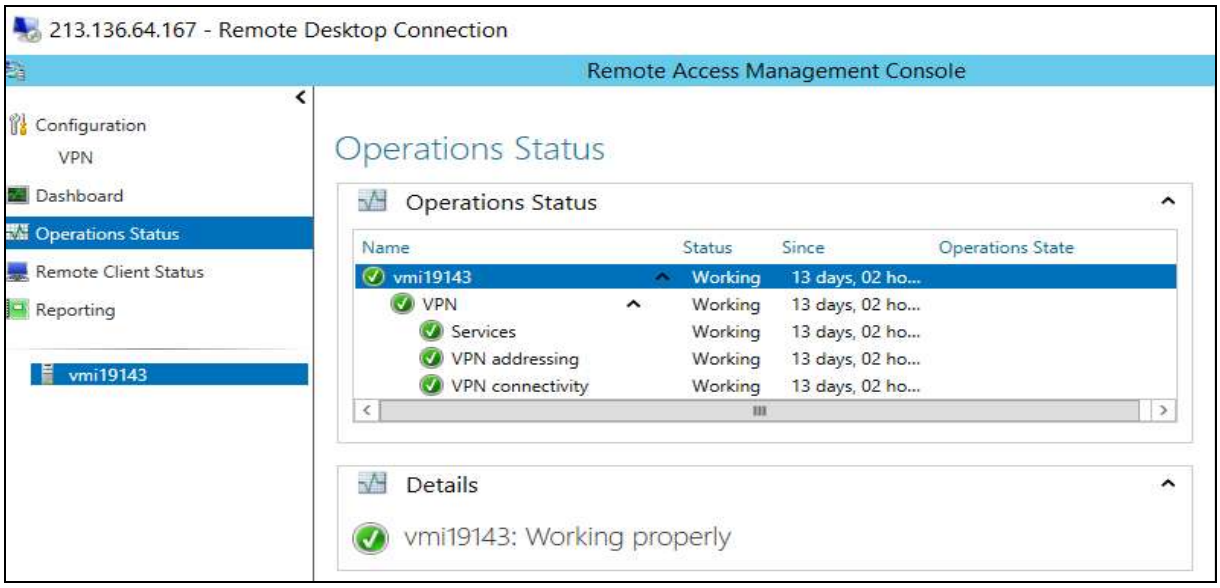
Εικόνα 30. Ενοικίαση cloud server στην contabo.com

Η εταιρεία παρέχει καλύτερες ταχύτητες σύνδεσης σε σχέση με την Ελλάδα, κάτι που φαίνεται στην εικόνα 31.

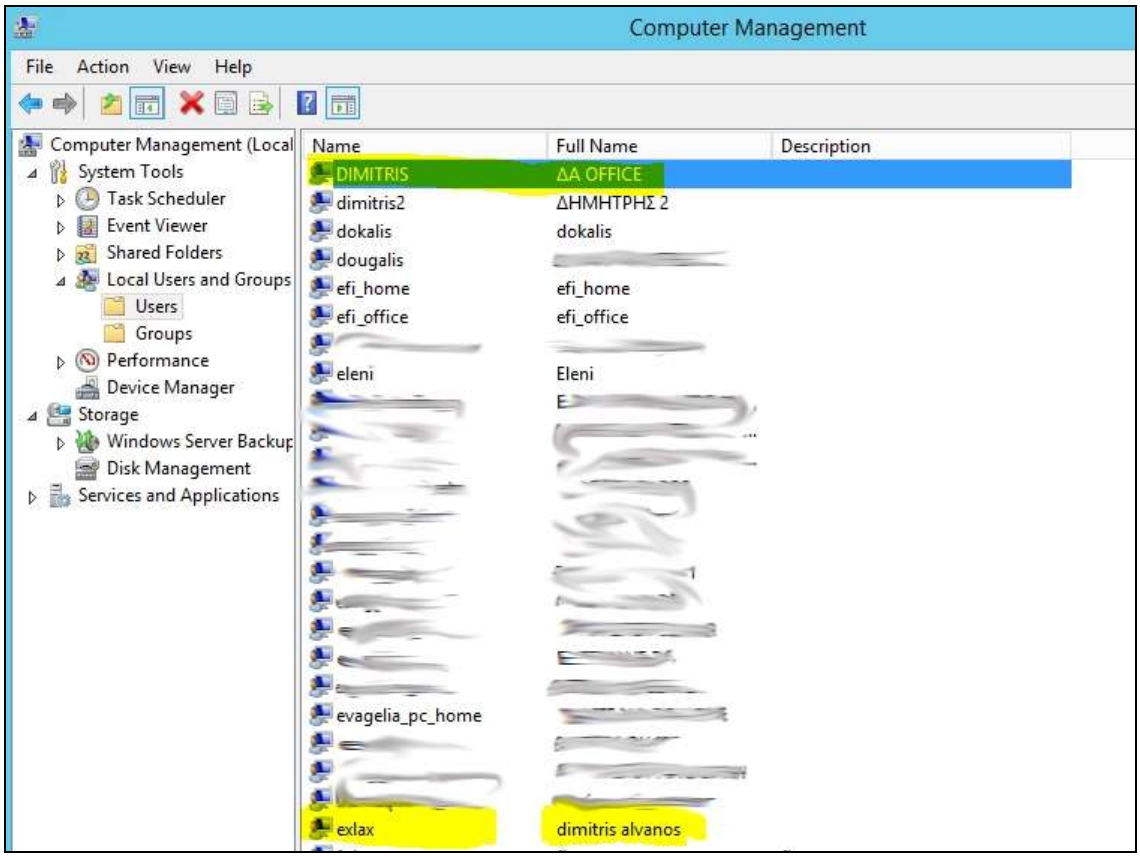


Εικόνα 31. Ταχύτητες σύνδεσης του virtual server μας.

Εκεί εγκαταστήσαμε ένα Windows server 2012 και μέσα από το Remote access management console, εγκαταστήσαμε τον VPN server και κατόπιν δημιουργήσαμε τους VPN users.

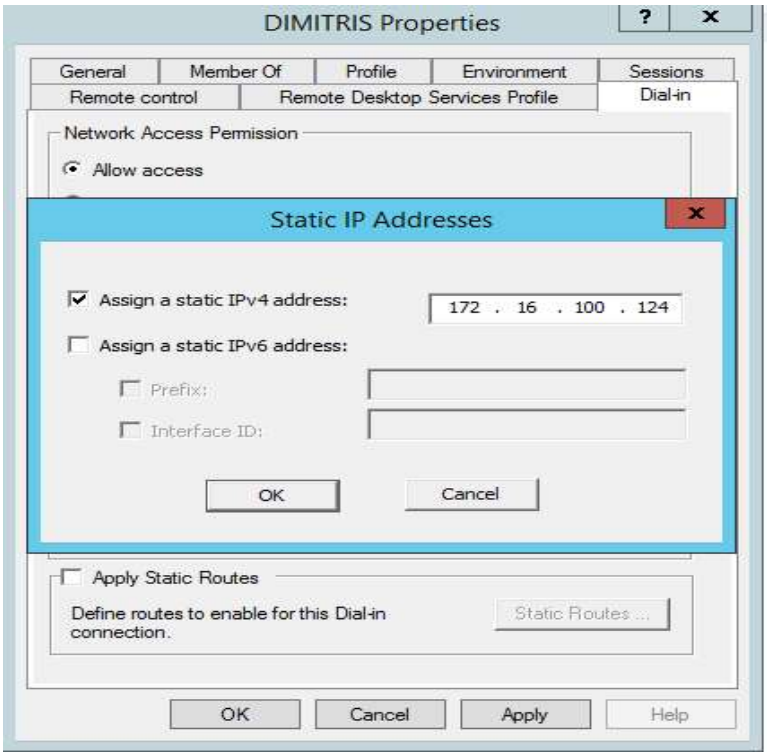


Εικόνα 32. Ο απομακρυσμένος VPN server μας.

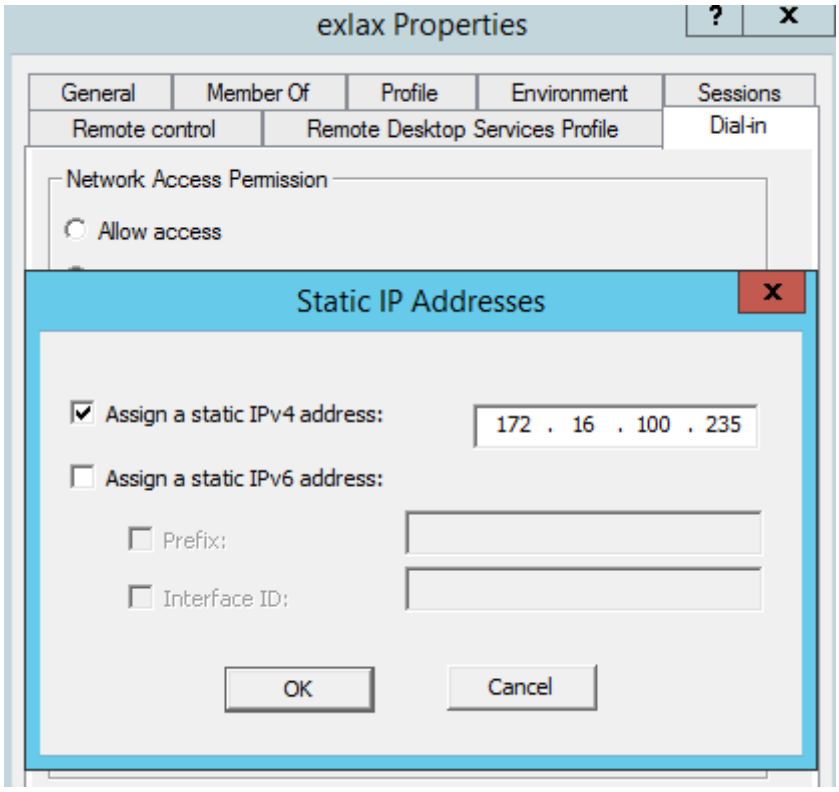


Εικόνα 33. Μερικοί από τους χρήστες που δημιουργήσαμε και οι οποίοι έχουν πρόσβαση στον VPN server μας.

Ο κάθε χρήστης έχει τώρα την δική του IP στον VPN server.



Εικόνα 34. Η IP του χρήστη Dimitri στον VPN server

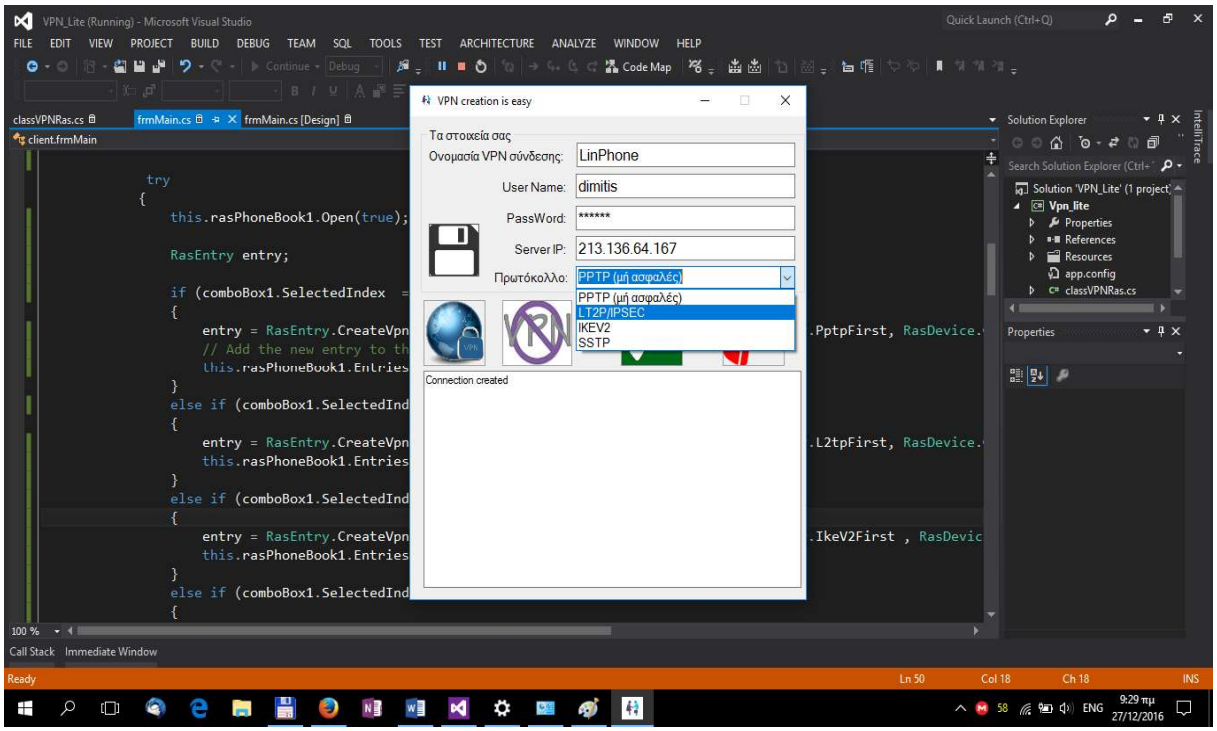


Εικόνα 35. Η IP του χρήστη exlax στον VPN server

Η δημιουργία του VPN server και η κατοχύρωση χρηστών σε αυτόν είναι δουλειά που πρέπει να γίνεται από τον SIP provider. Από εκεί και πέρα ο κάθε χρήστης είναι υπεύθυνος για τη δημιουργία, τη κλήση και τον τερματισμό της VPN σύνδεσης του. Αυτή η διαδικασία όμως προϋποθέτει γνώσεις και δεξιότητες που ξεπερνούν τα όρια ενός απλού χρήστη και έτσι δημιουργήσαμε μία εφαρμογή η οποία είναι σε θέση να εγκαθιστά την VPN σύνδεση στον υπολογιστή, να κάνει κλήση και να την τερματίζει.

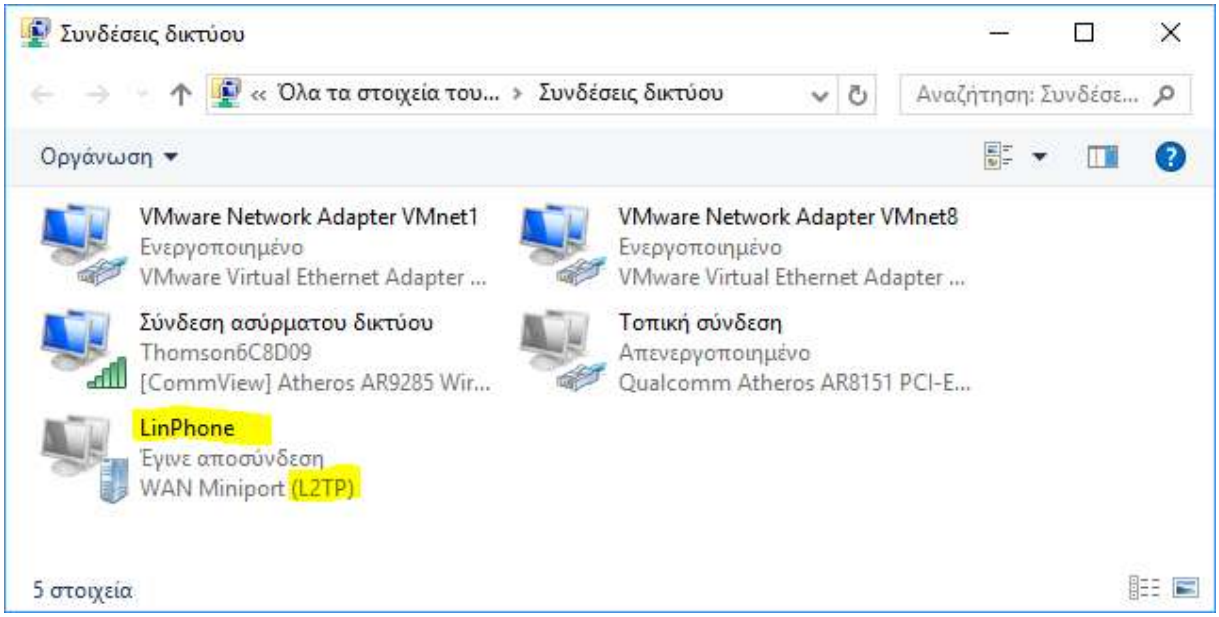
6.2 Software διαχείρισης VPN σύνδεσης στον χρήστη

Με αυτά κατά νου, δημιουργήσαμε μία εφαρμογή η οποία είναι σε θέση να διαχειριστεί VPN συνδέσεις σε περιβάλλον Windows. Η εφαρμογή μπορεί να δημιουργήσει συνδέσεις VPN με ενεργοποιημένο το PPTP, το LT2P, το IKEV2 ή το SSTP.

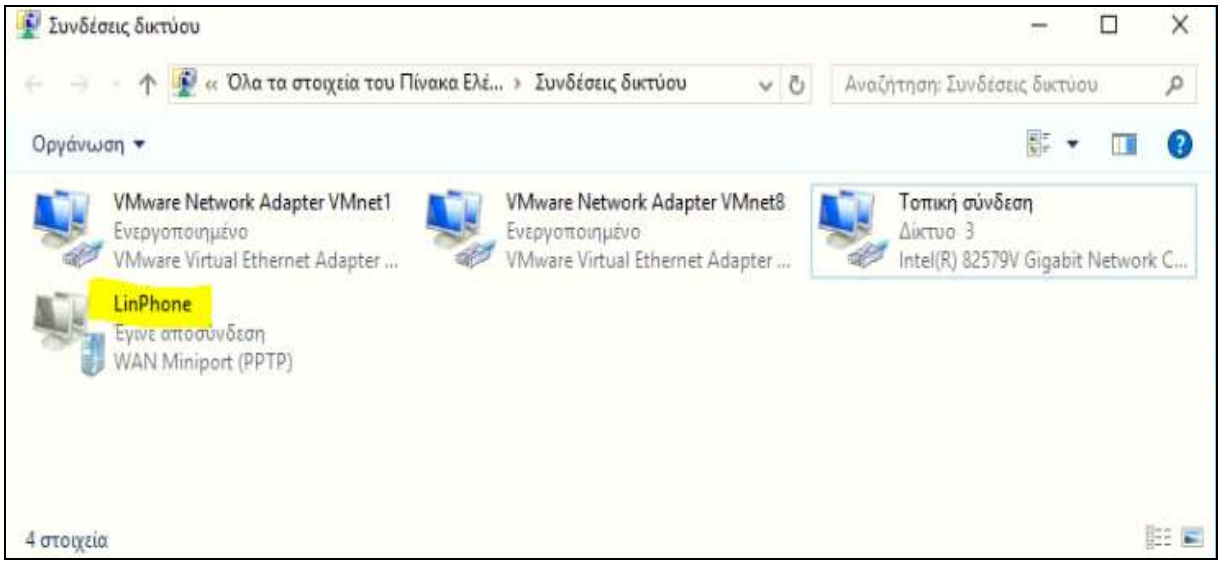


Εικόνα 36. Η εφαρμογή μας σε C#

Η εφαρμογή είναι σε θέση να κάνει όλη τη δουλειά για τον τελικό χρήστη, βάζοντας απλώς τα στοιχεία της VPN σύνδεσής του. Στην εικόνα 41 φαίνεται ότι μόλις δημιούργησε τη σύνδεση, όπως φαίνεται στην εικόνα 42, με τα credentials του χρήστη “dimitris”



Εικόνα 37. Η δημιουργημένη VPN σύνδεση με πρωτόκολλο L2TP

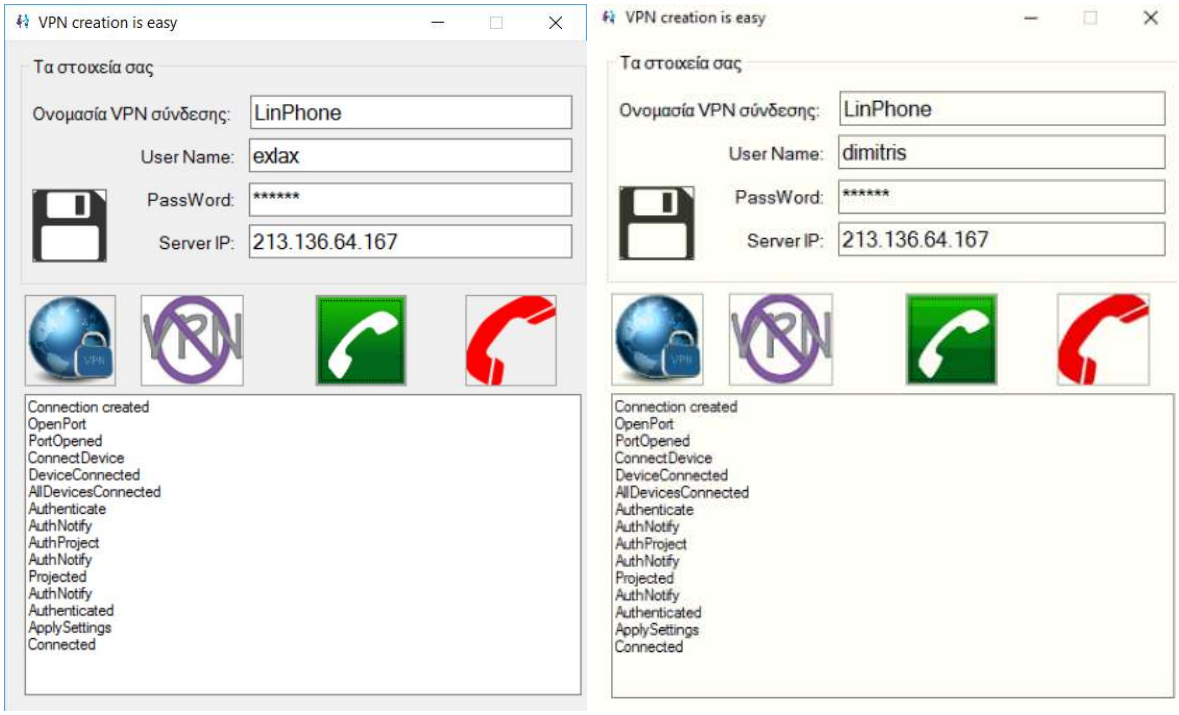


Εικόνα 38. Η VPN σύνδεση με πρωτόκολλο PPTP

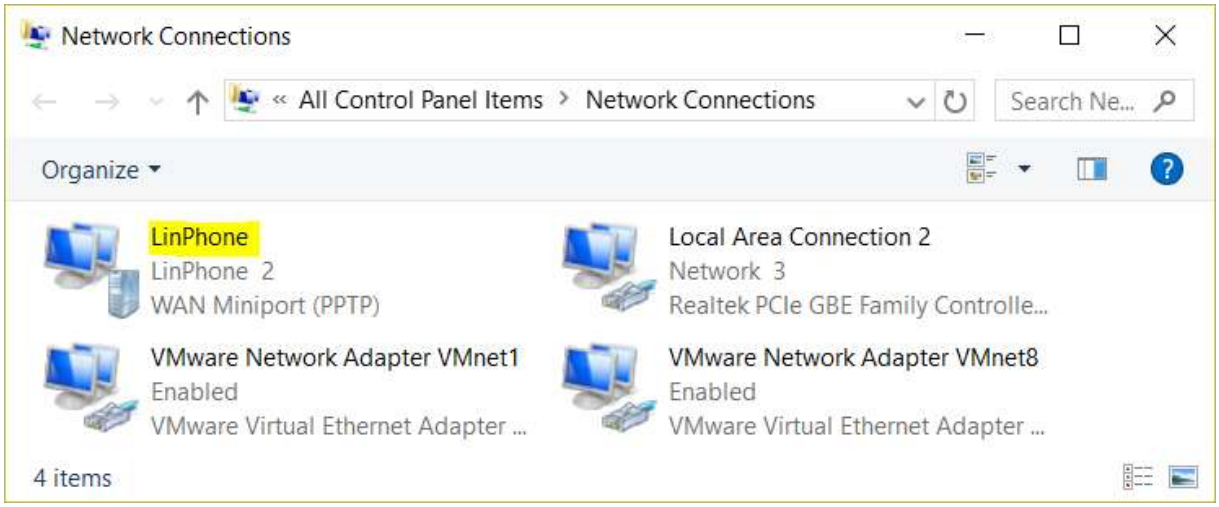
Σε άλλον υπολογιστή (εκτός του LAN του προηγούμενου), δημιουργήσαμε νέα VPN σύνδεση για τον χρήστη “exlax”

Εικόνα 39. Η 2η VPN σύνδεση

Τώρα αν ενεργοποιήσουμε τις δύο συνδέσεις, θα είμαστε σε θέση να επικοινωνήσουμε μέσω VPN σύνδεσης σαν να είμασταν σε LAN.

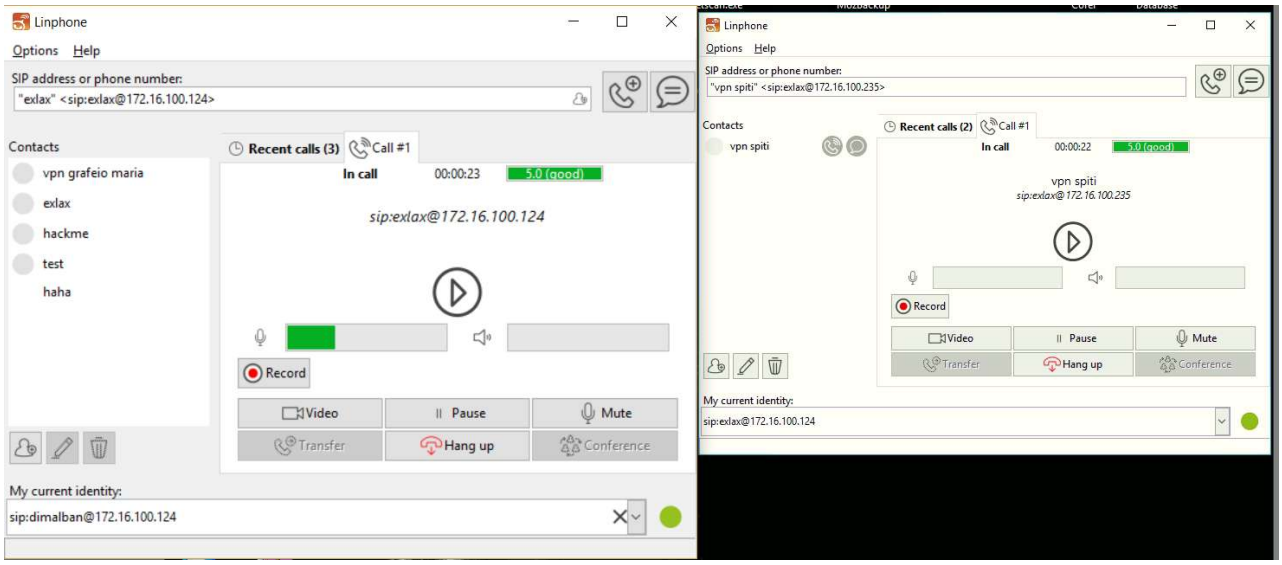


Εικόνα 40. Οι δύο συνδέσεις σε λειτουργία



Εικόνα 41. Ενεργή VPN σύνδεση

Τώρα πλέον είμαστε σε θέση να χρησιμοποιήσουμε το LinPhone για να καλέσουμε από το ένα άκρο στο άλλο μέσω της VPN σύνδεσης που δημιουργήσαμε.



Εικόνα 42. Κλήση μέσω VPN

Ο κώδικας της εφαρμογής παρατίθεται στο παράρτημα Α.

6.3 Απόδοση της VPN σύνδεσης

Στο συγκεκριμένο θέμα, δημιουργήσαμε VPN συνδέσεις με PPTP πρωτόκολλο που δεν θεωρείται και το ασφαλέστερο. Η ποιότητα της επικοινωνίας ήταν παραπάνω από ικανοποιητική ακόμη και με ενεργοποιημένη την αποστολή video μέσω web cams.

Στο εργαστηριακό περιβάλλον, όπως προαναφέρθηκε στο κεφ. 5.3, πειραματιστήκαμε και με συνδέσεις L2TP. Κατά τη διάρκεια των δοκιμών (που κράτησαν δύο εβδομάδες), κανένας δεν παραπονέθηκε για την ποιότητα της επικοινωνίας σε γενικές γραμμές. Υπήρξαν βέβαια περιπτώσεις (ελάχιστες) κατά τις οποίες ο δείκτης ποιότητας του LinPhone έπεφτε κάτω από το επίπεδο ”Good”.

Ως προς τον έλεγχο της απόδοσης χρησιμοποιήθηκαν γνωστά και κοινά εργαλεία ελέγχου ταχύτητας (download και upload speed testers) και τα οποία έδειξαν ελαφρώς αυξημένη ταχύτητα όταν η VPN σύνδεση ήταν ενεργή αλλά και αυτή η διαφορά εμπίπτει στα όρια του σφάλματος (π.χ. download speed 7,2 Mbps με VPN και 6,72 χωρίς).

Όπως αναφέρει και ο ιστότοπος techrepublic.com (Lowe 2002:1) [53] μπορεί κανείς να εφαρμόσει διάφορες τεχνικές για να αυξηθεί η ταχύτητα στο VPN. Αν χρησιμοποιείται PPTP πρωτόκολλο τότε ναι μεν είναι μειωμένη η ασφάλεια αλλά τα πακέτα δεδομένων υπόκεινται σε μικρότερη λιγότερη επεξεργασία. Το πακέτο PPP (point to point protocol) ενθυλακώνεται σε ένα πακέτο γενικής δρομολόγησης GRE (generic routing encapsulation), το οποίο κατόπιν περικλείεται σε ένα IP πακέτο και αποστέλεται.

Στο L2TP, τα πακέτα υπόκεινται σε τέσσερεις έως έξι ενθυλακώσεις, αναλόγως την IPSec πολιτική που ακολουθεί το VPN. Βέβαια το L2TP παρέχει αυξημένη ασφάλεια καθώς χρησιμοποιεί DES ή 3DES κρυπτογράφηση. Όλα αυτά το καθιστούν ελαφρώς πιο αργό από το PPTP.

Αξίζει να σημειωθεί ότι το PPTP βασίζεται στο TCP πρωτόκολλο ενώ το L2TP χρησιμοποιεί UDP πρωτόκολλο. Αυτό σημαίνει ότι το πρώτο είναι πιο αργό από το δεύτερο λόγω της αυξημένης πολυπλοκότητας του TCP πρωτοκόλλου.

Σε συνδυασμό των προαναφερθέντων βλέπει κανείς ότι από την μία κερδίζει το PPTP και από την άλλη αποκτάει προβάδισμα το L2TP, άρα διαφαίνεται ότιι σε πρακτικό επίπεδο ό,τι πρωτόκολλο και να επιλέξουμε για την VPN σύνδεση, θα πάρουμε σχεδόν παρόμοια απόδοση: αν χρειαστούμε μεγαλύτερη ταχύτητα, τότε καλύτερα να αυξήσουμε το bandwidth των δύο άκρων της σύνδεσης.

Κεφάλαιο 7
Αποτίμηση κινδύνων και βελτιώσεις

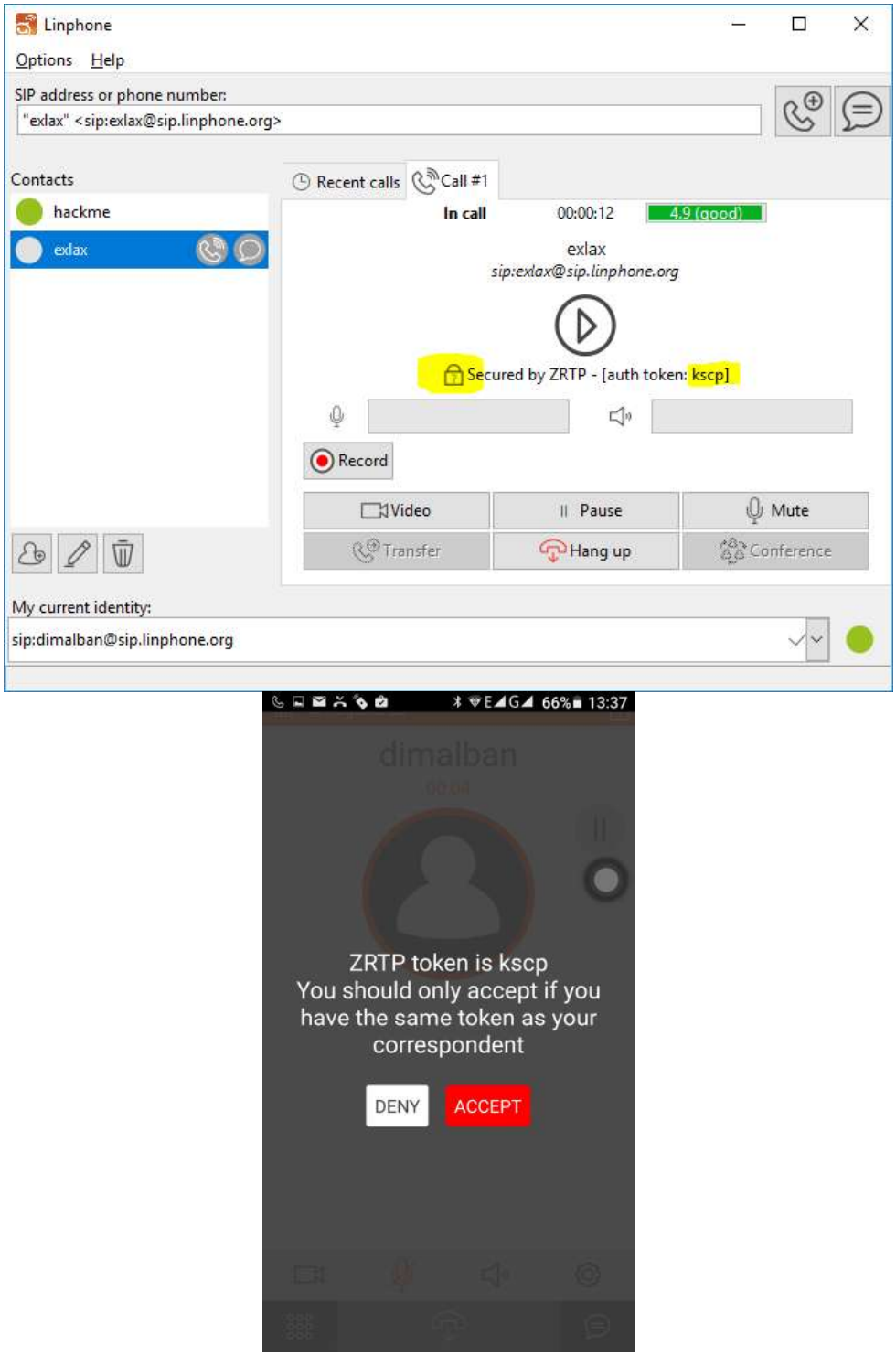
"*Είμαστε πιο ασφαλείς από μια τακτική τηλεφωνική γραμμή*" ισχυρίζεται μία εταιρεία παροχής υπηρεσιών VoIP στη βόρεια Αμερική.

7.1 Το παρόν της VoIP επικοινωνίας και κίνδυνοι που απειλούν την ύπαρξη και την εξέλιξη του VoIP

Είναι γεγονός ότι το VoIP είναι εδώ για να μείνει. Στην πραγματικότητα, πολλοί φορείς κλασικών τηλεπικοινωνιών έχουν αρχίσει να προσφέρουν υπηρεσίες VoIP και άλλοι παρέχουν μόνο τηλεφωνία μέσω VoIP π.χ. Cyta. Εκτός από θέματα όπως η ποιότητα των υπηρεσιών, η ασφάλεια ή η έλλειψη της είναι παρεξηγημένη από μερικούς από τους παρόχους υπηρεσιών VoIP. Στο σχήμα 25 φαίνεται ότι ο παράγοντας κίνδυνος είναι συνάρτηση πολλών παραμέτρων.

Σχήμα 25. Παράγοντες που επηρεάζουν την έννοια του κινδύνου.

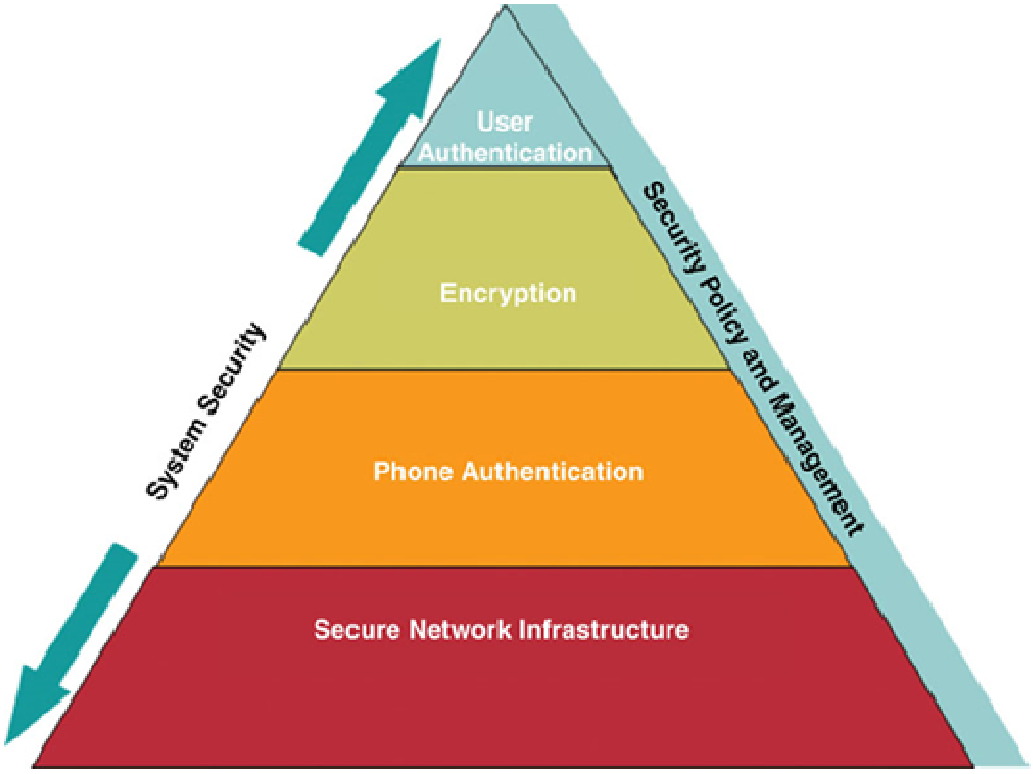
Ο ίδιος ο δημιουργός του ZRTP (όπως και του PGP και του ZFone), Phil Zimmermann [46] λέει ότι "*εγώ έκανα ότι μπορούσα για να κρυπτογραφήσω την συνομιλία (εννοώντας τη δημιουργία του ZRTP). Τώρα είναι στο χέρι των συνομιλούντων να πειθαρχήσουν στο SAS string που τους εμφανίζεται στον UA τους ώστε να διασφαλίσουν την μη ύπαρξη ενδιάμεσου "ωτακουστή"."*



Εικόνα 43. Κλήση από υπολογιστή σε android smartphone.

Στην εικόνα 43 φαίνεται το SAS string που εμφανίζεται σε PC και android συσκευή ώστε να γίνει η επαλήθευση της αυθεντικότητας του συνομιλούντος.

7.2 Ολοκληρωμένη ασφάλεια



Σχήμα 26. Ολοκληρωμένη ασφάλεια

Στο κατώτερο μέρος της πυραμίδας της ασφάλειας βρίσκονται οι υποδομές ασφαλούς δικτύωσης όπου εφαρμόζονται ασφάλειες στο 2ο και στο 3ο επίπεδο. Σε αυτό το τμήμα η αρχιτεκτονική 802.1x ελέγχει την πρόσβαση στην ασφάλεια που βασίζεται στις θύρες (ports) και δεν χορηγείται καμία πρόσβαση, εκτός εάν η συσκευή μπορεί να αυθεντικοποιηθεί. Επιπλέον, σε αυτή την κατώτατη βαθμίδα της πυραμίδας, εφαρμόζεται η ανίχνευση εισβολής (IDS) και η πρόληψη αυτής (IPS). Εδώ υλοποιείται και το τείχος προστασίας φροντίζοντας για την επικοινωνία VoIP και NAT. Η παρακολούθηση όλων των δραστηριοτήτων για οποιαδήποτε μορφή κακόβουλης συμπεριφοράς θα κλείσει τις θύρες που εμφανίζουν κακοδιαχείριση (μιλάμε πάντα για ασφαλή πρόσβαση δικτύου).

Στη 2η κλίμακα της πυραμίδας είναι ο έλεγχος ταυτότητας τηλεφώνου ή αυθεντικοποίησης. Για άλλη μια φορά με ένα αμοιβαίο σύστημα που βασίζεται σε πιστοποιητικά, οι UA's δεν θα εξουσιοδοτούν ούτε θα τους επιτρέπεται η πρόσβαση στο δίκτυο ή στο PBX, εκτός εάν πληρούνται οι προδιαγραφές ασφάλειας. Σε επίπεδο

δικτύου (επίπεδο 2 ή 3), τα τηλέφωνα θα πρέπει να έχουν πρόσβαση στο τοπικό δίκτυο μετά τον επιτυχή έλεγχο ταυτότητας στις συσκευές δικτύου (switches επίπεδου 2 ή 3, που χρησιμοποιούν το πρωτόκολλο 802.1x και ταυτότητα επίπεδου MAC). Στο PBX το τηλέφωνο θα πρέπει, επίσης, να επικυρώνεται στο διακομιστή. Αυτή είναι η δεύτερη φάση του ελέγχου ταυτότητας στο οποίο λαμβάνει χώρα η αμοιβαία ανταλλαγή πιστοποιητικών. Η διαδικασία αυτή χρησιμοποιείται για τις περισσότερες μεθόδους πρόσβασης σε ένα δίκτυο δεδομένων, και θα πρέπει να χρησιμοποιούνται όμοιες σε ένα περιβάλλον VoIP.

Το επόμενο επίπεδο στην πυραμίδα είναι το στάδιο ασφάλειας για την εξασφάλιση των δεδομένων και των καναλιών ελέγχου. Εδώ μπαίνει η κρυπτογράφηση VPN/IPSec. Η κρυπτογράφηση θα πρέπει να είναι μεταξύ των συσκευών σηματοδότησης. Το VPN απ' άκρο σ' άκρο είναι ένα χαρακτηριστικό παράδειγμα του τι πρέπει να χρησιμοποιείται για αυτές τις συσκευές καθώς και μία IPSec σήραγγα. Το επίπεδο αυτό καλείται να χειριστεί αξιόπιστες συνδέσεις των συσκευών, δεν θα ήταν σκόπιμο να δημιουργεί ένα VPN ή IPSec τούνελ με μια άγνωστη συσκευή που δεν είναι υπό τον έλεγχό μας.

Στην κορυφή της πυραμίδας είναι ο χρήστης και ειδικά ο έλεγχος ταυτότητας του χρήστη. Καθώς ο άνθρωπος βρίσκεται στα δύο ακραία σημεία (τηλέφωνο με τηλέφωνο) ο χρήστης θα πρέπει υπόκειται σε έλεγχο ταυτότητας στο διακομιστή χρησιμοποιώντας το όνομα χρήστη και τον κωδικό πρόσβασης, ή μια συσκευή παραγωγής token (όπως αυτές που χορηγούν οι τράπεζες), ή μια αμοιβαία ανταλλαγή πιστοποιητικών. Ανεξαρτήτως της μεθοδολογίας, ο στόχος είναι να επιτραπεί μόνο σε εξουσιοδοτημένους και επικυρωμένους χρήστες να πραγματοποιούν και να δέχονται κλήσεις. Αν αυτό το σημείο παραμεληθεί, τότε η πιθανή χρεωστική απάτη θα μπορούσε να είναι εξαιρετικά υψηλή όπως και ο κίνδυνος sniffing, υποκλοπής κ.λπ. Επιπλέον, εάν η θύρα για ένα τηλέφωνο δεν κλειδωθεί, ένας δυνητικός χάκερ θα μπορούσε να την χρησιμοποιήσει ως μέσο πρόσβασης για να σφετεριστεί τη βασική ασφάλεια του τηλέφωνο εκτελώντας ένα sniffer (Vomit, Wireshark, Ettercap κ.λπ.) και να προσπαθήσει να εισέλθει μεταξύ των φωνής και των δεδομένων VLANs (άλμα του τείχος μεταξύ των VLANs). Στην ουσία, όσο λιγότερη πρόσβαση δίνουμε, τόσο μειώνεται η ποσότητα των αποπειρών cracking και hacking που μπορούν να χρησιμοποιηθούν.

Στις δύο πλευρές της πυραμίδας υπάρχουν δύο ξεχωριστές ετικέτες. Στην αριστερή πλευρά βρίσκεται η ολοκληρωμένη προσέγγιση. Αυτό εννοεί ο όρος ολοκληρωμένη ασφάλεια, όταν λαμβάνεται ως προσέγγιση για ολόκληρο το σύστημα. Οποιοδήποτε από τα κομμάτια της πυραμίδας μπορεί να εφαρμοστεί μεμονωμένα αλλά το σύνολο του συστήματος δεν θα είναι ασφαλές. Στην πραγματικότητα, μια ψεύτικη αίσθηση ασφάλειας θα πλανάται αν εφαρμοστούν τμήματα μόνο ασφαλείας παρά μια ολιστική προσέγγιση. Στη δεξιά πλευρά της πυραμίδος βρίσκονται αυτά που πρέπει να έρθουν σε επαφή. Οι πολιτικές ασφάλειας πρέπει να ενσωματωθούν τόσο με τα δεδομένα όσο και την VoIP πλευρά του δικτύου, καθώς και ένα ενιαίο σύνολο εργαλείων διαχείρισης που να είναι σε θέση να παρακολουθεί το σύνολο του συστήματος. Βεβαίως, μπορεί να υπάρχουν μερικά εργαλεία που αφορούν μόνο στη φωνή (λόγω των media και των καναλιών σηματοδοσίας), αλλά θα εξακολουθούν να ενσωματώνονται σε ένα ομοιογενές σύστημα για να παρακολουθούν, να προλαμβάνουν, και να αντιδρούν κάθε φορά που συμβαίνει κάτι. Οι στόχοι αυτής της ολιστικής προσέγγισης είναι να αυξηθεί η παραγωγικότητα του τελικού χρήστη, έτσι ώστε αν επικυρωθούν μία φορά, να μην χρειάζεται να ανησυχούν για παραβιάσεις ασφαλείας που θα τους επηρεάσει. Ένας άλλος σαφής στόχος είναι η διαχείριση περιουσιακών στοιχείων, η προστασία αυτών. Σε μία εταιρεία, οι διαχειριστές VoIP, πρέπει να είναι ευαισθητοποιημένοι σε θέματα ασφαλείας διότι κάθε παράβαση είναι ένα χτύπημα για την εταιρεία (δείτε τι έγινε με τη Yahoo και την μετοχή της, όταν κλάπηκαν χιλιάδες στοιχείων πελατών της τον Δεκέμβριο του 2016). Ο επανειλημμένες παραβιάσεις θα κοστίσουν σε χρήματα, σε εμπιστοσύνη που επιδεικνύει ο πελάτης και σε θέσεις εργασίας προφανώς. Αν όλα λειτουργούν, το τελικό αποτέλεσμα θα είναι ένα ασφαλές δίκτυο, ασφαλής επικοινωνία και μείωση των συνολικών λειτουργικών εξόδων.

7.3 Προτάσεις βελτίωσης

Στα πλαίσια της διατριβής μελετήθηκε ο κώδικας του LinPhone (σε Windows επίπεδο) και ανακαλύψαμε ότι πέρα από μικροδιορθώσεις και αισθητικές επεμβάσεις (που δικαιολογούν άλλωστε και τις διάφορες εκδόσεις ενός προγράμματος), δεν χρήζει στην παρούσα φάση βελτίωσης πέρα από την δημιουργία μηχανισμού αποφυγής των ανιχνευτικών κλήσεων των οποίων γίναμε μάρτυρες τουλάχιστον δύο φορές. Οι κλήσεις αυτές που ονομάζονται “ghost calls” είναι πολύ εκνευριστικές και αποτρέπουν το χρήστη από το να δημιουργήσει “δεσμό” με την εφαρμογή.

Κάτι παρόμοιο συμβαίνει και με το γνωστό Skype στο οποίο εμφανίζονται διαφημίσεις και πυκνά-συχνά κάποιος/α/ο μας ζητάει άδεια για να προστεθεί στις επαφές μας. Επειδή όμως το Skype και άλλα παρόμοια προγράμματα (Viber κ.λπ.) είναι εμπορικά, δεν μπορούμε να κάνουμε ή να δεχτούμε κλήσεις από κάποιον εκτός του συγκεκριμένου δικτύου. Με UA's τύπου LinPhone μπορούμε να δεχτούμε κλήση από οποιονδήποτε έχει μία SIP ταυτότητα ή βρίσκεται στο τοπικό μας δίκτυο ή μπορεί να μας προσεγγίσει χωρίς τη χρήση proxy. Κάτι τέτοιο όμως μπορεί να οδηγήσει σε παρενέργειες τύπου "ghost calls" που περιγράψαμε το κεφάλαιο 5.

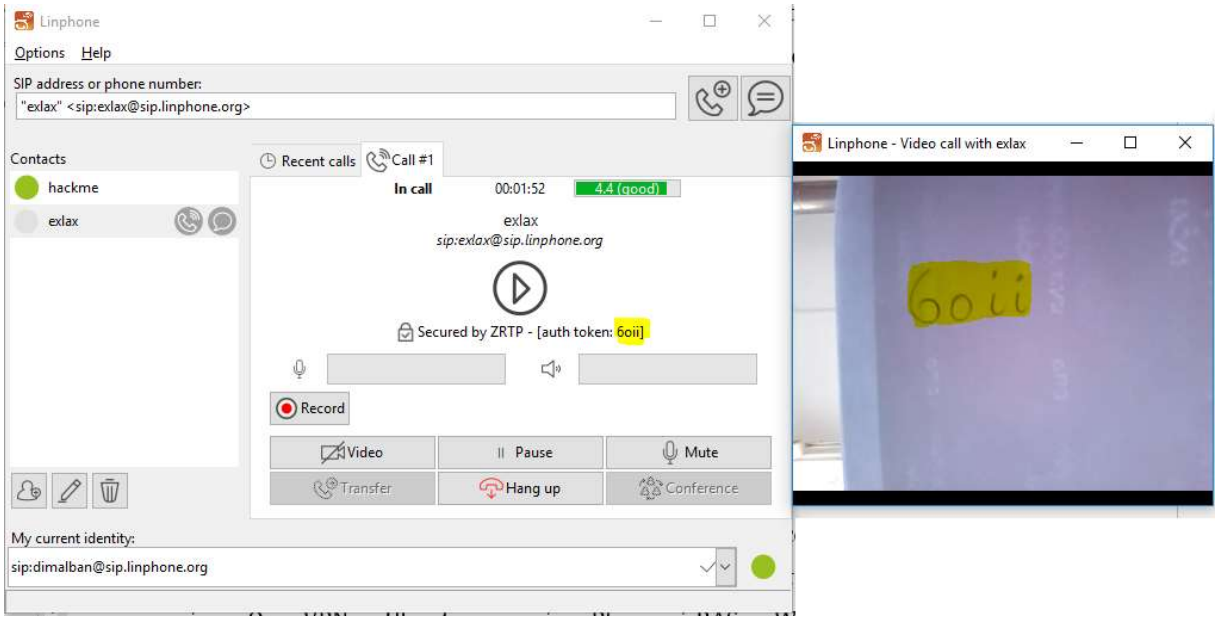
Με βάση αυτό στο μυαλό μας, προτείναμε μία βελτίωση στους δημιουργούς του Linphone: Να δώσουν στο χρήστη το δικαίωμα της επιλογής αν θα δέχονται κλήσεις από μη εξουσιοδοτημένους χρήστες δηλαδή από χρήστες εκτός του xxx@sip.linphone.org. Κάτι τέτοιο βέβαια θα φέρει το LinPhone στην ίδια θέση με το Skype, Viber κ.λπ. και θα δέχονται κλήσεις μόνο από "ενδοεταιρικούς" πελάτες. Φυσικά αυτό δεν θα αποτρέψει και πολύ τους υποψιασμένους χάκερς από το να δημιουργήσουν λογαριασμό στο sip.linphone.org και να επιτίθενται με αυτόν.

Θετικό είναι το γεγονός ότι το LinPhone δεν φαίνεται να είναι ευπαθές σε επιθέσεις τύπου BEAST που αφορούν σε CBC cipher διότι χρησιμοποιεί Counter mode CBC cipher καθώς και σε επιθέσεις που αφορούν RC4 κρυπτογραφικούς αλγορίθμους διότι δεν χρησιμοποιεί ούτε αυτόν.

Επίσης, θα πρέπει να γίνεται οπωσδήποτε χρήση του TLS πρωτοκόλλου σηματοδοσίας και να ειδοποιείται ο χρήστης αν υποβιβαστεί η ασφάλεια σε επίπεδο UDP ή TCP καθώς και του ZRTP πρωτοκόλλου μετάδοσης φωνής εκτός και αν ο χρήστης δεν ενδιαφέρεται καθόλου αν θα υποκλαπούν τα στοιχεία του και η ίδια η συνομιλία. Θα πρέπει όχι μόνο να ενεργοποιείται de facto το ZRTP πρωτόκολλο αλλά και να μην ξεκινάει το κύριο μέρος της συνομιλίας αν δεν γίνει click στο checkbox του SAS string δηλώνοντας έτσι ότι έγινε η επαλήθευση και από τα δύο μέλη.

Μία πρόταση που βελτιώνει σε μέγιστο βαθμό (μέχρι να αποδειχτεί το αντίθετο φυσικά), την αυθεντικοποίηση του συνομιλούντος και αποφεύγει τις επιθέσεις MiTM (όπως περιγράφηκαν στο 4.2.3) είναι η ανάγνωση της SAS τιμής να μη γίνεται από το

τηλέφωνο αλλά να γράφεται σε ένα χαρτί και να εμφανίζεται μέσω της κάμερας στον συνομιλών, με την προϋπόθεση ότι θα εμφανίζονται και τα πρόσωπα των συνομιλούντων στην ίδια ροή βίντεο. Στην εικόνα 44 φαίνεται το SAS string και η απάντηση που δίνει το άλλο μέλος γράφοντας το σε ένα χαρτί και εμφανίζοντάς το στην κάμερα του κινητού του. Έγινε πρόταση γι' αυτό στην Belledonne communications και μας απάντησαν ότι θα περιληφθεί σε μελλοντική έκδοση του εγχειριδίου χρήσης ως προτροπή προς τους χρήστες.



Εικόνα 44. Έλεγχος του SAS string μέσω χαρτιού-κάμερας

Επίλογος

Αποτελέσματα της έρευνας

Στην παρούσα διατριβή μελετήθηκε ενδελεχώς η εφαρμογή Linphone, τόσο ως προς τη θεωρητική ανάλυση της ασφάλειας των υποκείμενων πρωτοκόλλων όσο και ως προς ζητήματα ασφάλειας που άπτονται της υλοποίησής του και τα οποία αναδείχτηκαν σε ρεαλιστικό πειραματικό περιβάλλον. Τα κύρια αποτελέσματα της παρούσας έρευνας συνοψίζονται ως εξής:

1) Η εφαρμογή Linphone επιτρέπει σε δύο χρήστες να συνομιλήσουν χωρίς να απαιτεί υποχρεωτικά τη χρήση της κρυπτογράφησης στα δεδομένα σηματοδοσίας (πρωτόκολλο TLS). Σε αυτήν την περίπτωση, είναι εφικτό να αναγνωριστούν τα τμήματα της επικοινωνίας που αντιστοιχούν στην ZRTP επικοινωνία.

2) Σε συνέχεια του ανωτέρω, ακόμα και αν το ένα μέλος της επικοινωνίας έχει ενεργοποιημένο TLS, μπορεί να πραγματοποιηθεί μη κρυπτογραφημένη επικοινωνία, εφόσον αυτή εκκινηθεί από άλλο μέλος που δεν έχει ενεργοποιημένο το TLS (με άλλα λόγια, η UDP επικοινωνία σηματοδοσίας που εκκινείται υποβαθμίζει το παρεχόμενο επίπεδο ασφάλειας «αναγκάζοντας» την άλλη πλευρά, ακόμα και αν αυτή έχει τη βέλτιστη ρύθμιση ασφάλειας, να προσαρμοστεί στη όχι ασφαλή ρύθμιση).

3) Όταν δεν χρησιμοποιείται το TLS, διαπιστώθηκε στην πράξη ότι το σύστημα μπορεί εύκολα να αποτελέσει στόχο επιτιθέμενου ο οποίος πραγματοποιεί κακόβουλες κλήσεις («ghost calls») που σκοπό φαίνεται ότι έχουν να ελέγξουν συνολικά, εν είδη ανίχνευσης ευπαθειών, το σύστημά μας.

4) Δεδομένου ότι υπάρχουν εξειδικευμένες τεχνικές που μπορούν να παρακάμψουν την ασφάλεια του ZRTP και να επιτύχουν επίθεση MiTM, είτε γιατί οι τελικοί χρήστες δεν προβαίνουν σε σύγκριση του SAS string είτε γιατί η σύγκριση είναι μόνο φωνητική, στην παρούσα διατριβή προτείνεται η υποχρεωτική σύγκριση του SAS string μέσω βίντεο – γεγονός το οποίο θα καταστήσει αμέσως αντιληπτό τον επιτιθέμενο σε μία τέτοιου τύπου επίθεση.

5) Η χρήση του Linphone εντός ενός εγκαθιδρυμένου VPN δεν φαίνεται, κατ' αρχάς, να δημιουργεί πρόβλημα στην ποιότητα της επικοινωνίας (καθυστερήσεις κτλ.), γεγονός το οποίο υποδηλώνει ότι VPNs θα μπορούσαν να αξιοποιηθούν, για παράδειγμα σε περιπτώσεις επικοινωνιών εντός ενός μεγάλου οργανισμού, για παροχή πρόσθετης ασφάλειας.

Συμπεράσματα - Μελλοντική έρευνα

Σε αυτή την ενότητα δίνονται κάποιες επιπλέον σκέψεις που θα πρέπει να εξεταστούν κατά την προστασία και την ενσωμάτωση ενός συστήματος VoIP σε ένα δίκτυο δεδομένων. Είναι εξαιρετικά σημαντικό να χρησιμοποιηθεί μια πολυεπίπεδη προσέγγιση στο δίκτυο VoIP.

- Ο βασικός στόχος είναι να διατηρηθεί η QoS του VoIP, ενώ ταυτόχρονα να παρέχει την απαραίτητη ασφάλεια ελέγχου, σηματοδότησης, και κανάλια μετάδοσης δεδομένων.

- Όταν εξετάζονται θέματα κοινωνικής μηχανικής (social engineer) πρέπει να γίνει αντιληπτό ότι για έναν επιτιθέμενο, τόσο η πλαστογράφηση του ID του καλούντος όσο και η λήψη πληροφοριών για την επικοινωνία VoIP είναι εύκολη. Οι τελικοί χρήστες θα πρέπει να γνωρίζουν τη δυνατότητα αυτή και να εκπαιδεύονται ώστε να μην εμπιστεύονται το Caller ID που εμφανίζεται. Μια τέτοια πλαστή κλήση εμφάνιζε ότι προερχόταν από ένα πολύ υψηλόβαθμο στέλεχος σε μια οργάνωση και το προσωπικό του γραφείου υποστήριξης έδωσε το όνομα χρήστη και τον κωδικό πρόσβασης στον καλούντα, επιτρέποντας την πρόσβαση του στο λογαριασμό του ανώτερου διευθυντικού στελέχους της.

- Έλεγχοι για τα συστήματα VoIP θα πρέπει να γίνονται συχνά για να εξασφαλιστεί ότι κανένας κακόβουλος εισβολέας δεν έχει θέσει σε κίνδυνο το σύστημα. Εάν ένας κακόβουλος χρήστης θέσει σε κίνδυνο το σύστημα, μπορούν να ανακατευθύνει τις εξερχόμενες κλήσεις.

- Το πρωτόκολλο TLS θα πρέπει να αποτελεί μία εξ ορισμού προ-ρυθμισμένη επιλογή για τη σηματοδοσία. Αντίστοιχα, το SRTP για τη μετάδοση φωνής θα

πρέπει να συνοδεύεται με ασφαλές πρωτόκολλο ανταλλαγής κλειδιού: ως προς αυτό, εφόσον χρησιμοποιείται το ZRTP, να απαιτείται υποχρεωτική επιβεβαίωση του SAS string μέσω εικόνας video των δύο μελών.

- Σκόπιμο είναι να δίνεται στους χρήστες η δυνατότητα της επιλογής ως προς το αν θα δέχονται κλήσεις από μη εξουσιοδοτημένους χρήστες.

- Η χρήση VPN για την κυκλοφορία και τον έλεγχο των πληροφοριών πρέπει να είναι εκ των ουκ άνευ σε περιπτώσεις κρίσιμων ως προς τη σημασία τους επικοινωνιών. Σαφέστατα ωστόσο, χρειάζεται περαιτέρω έρευνα για την ποσοτικοποίηση της απόδοσης του Linphone σε διάφορα VPN πρωτόκολλα..

- Να γίνεται χρήση της ZRTP μετάδοσης γνωρίζοντας ότι μπορεί να εκτεθούν οι κεφαλίδες και να τεθεί σε κίνδυνο η επικοινωνία γι' αυτό πρέπει να ελέγχουμε πάντα το SAS string.

Υστερόγραφο

Σε ένα εξαιρετικό άρθρο της η Spiegel [49], γράφει: "*Το Skype collection αποτελεί ένα ακόμη βήμα στην κούρσα μεταξύ των μυστικών υπηρεσιών που επιδιώκουν την άρνηση στους χρήστες της ιδιωτικής τους ζωής και όσων θέλουν να εξασφαλίσουν την προστασία τους. "* και "*για την NSA, οι κρυπτογραφημένες επικοινωνίες θεωρούνται απειλή*."

Ο Ed Snowden, το 2013, πριν διαφύγει στο Χονγκ-Κονγκ, έδωσε στη δημοσιότητα έγγραφο που δείχνει ποια κρυπτογραφικά συστήματα έχει ξεκλειδώσει η υπηρεσία και σε ποια δεν είχε ακόμη επιτυχία.

Επίσης γίνεται αναφορά (Masnick 2014:1) [50] στο γεγονός ότι η NSA προσπαθεί να "σπάσει" τα VPN δίκτυα και το έχει καταφέρει σε πολλά από αυτά, ιδιαίτερα σε όσα βασίζονται στο PPTP καθώς και σε μερικά που βασίζονται στο IPSEC.

Η NSA από την άλλη έχει μεγάλες δυσκολίες με τα open source λογισμικά διότι δεν μπορεί να εγκαταστήσει backdoors ή οτιδήποτε άλλο χωρίς να γίνουν αντιληπτά, όπως πιθανώς να κάνει σε εμπορικά λογισμικά.

Η NSA ταξινομεί τη δυσκολία αποκρυπτογράφησης που αντιμετωπίζει σε πέντε επίπεδα "trivial, minor, moderate, major και catastrophic". Η σημασία των λέξεων έχει την αντίστροφη έννοια από αυτή έχει στον καθένα από εμάς. Έτσι, με **trivial** χαρακτηρίζουν την πιο εύκολη υφαρπαγή δεδομένων όπως μίας απλής ιστοσελίδας στο διαδίκτυο. Οι συνομιλίες στο Facebook χαρακτηρίζονται ως **minor** δυσκολίας. Η αποκρυπτογράφηση των emails που αποστέλλονται μέσα από κρυπτογραφημένους servers όπως ο mail.ru θεωρείται **moderate**. **Major** χαρακτηρίζονται τα προβλήματα όπως η μεταφορά emails μέσα από βαριά κρυπτογραφημένους mail servers όπως ο Zoho ή η παρακολούθηση των χρηστών μέσα στο Tor δίκτυο όπου χρησιμοποιώντας εθελοντές υπολογιστές (πάνω από 6000) ως πύλες, τα δεδομένα κρυπτογραφούνται και δεν φεύγουν όλα μαζί από έναν εθελοντή αλλά διασπώνται. **Major** πρόβλημα θεωρείται και το πρόγραμμα κρυπτογράφησης Truecrypt το οποίο κρυπτογραφεί αρχεία στον υπολογιστή. Οι κατασκευαστές του μάλιστα σταμάτησαν την ανάπτυξή του, από τον Μάιο του 2013, αφήνοντας αιχμές για κυβερνητικές παρεμβάσεις. Τέλος στο επίπεδο πέντε - "**catastrophic**" υπάρχει ένας συνδυασμός από Tor δίκτυο, επιπλέον υπηρεσία ανωνυμοποίησης, το σύστημα μηνυμάτων CSpace και το ZRTP όπου η εθνική υπηρεσία πληροφοριών της Αμερικής δεν έχει σχεδόν καθόλου πρόσβαση στα δεδομένα.

Τον Ιούλιο του 2016 εμφανίστηκε ένα σχέδιο (Internet draft) (Wikipedia 2016) [51] το οποίο αναφέρει προδιαγραφές για το TLS v1.3 χωρίς να είναι οριστικές και χωρίς να γνωρίζουμε σε τι ακριβώς θα διαφέρει από την v1.2. Ωστόσο, γνωρίζουμε μερικά πράγματα που πρόκειται να παρέχει η νέα έκδοση v1.3 όπως, την πλήρη αφαίρεση των κρυπτογραφικά ασθενών στοιχείων σαν το MD5, το RC4, τις αδύναμες ελλειπτικές καμπύλες και όσων στοιχείων χρησιμοποιούνται σπανίως όπως, η συμπίεση και ο αλγόριθμος "αλλαγής κρυπτογράφησης" και από την άλλη θα προστεθούν νέοι αλγόριθμοι ελλειπτικών καμπυλών.

Όταν η έκδοση οριστικοποιηθεί και σταθεροποιηθεί στις τυποποιημένες TLS βιβλιοθήκες (π.χ. OpenSSL), θα υπάρξει μια ισχυρή ώθηση απομάκρυνσης από τις TLS v1.0 και v1.1 και αποκλειστική υιοθέτηση των εκδόσεων v1.2 και v1.3 και ίσως στο

μέλλον, η έκδοση v1.4 να αρχίσει να περιλαμβάνει στοιχεία από τους αλγορίθμους New Hope post-Quantum της Google (Braithwaite 2016:1) [52].

Παραδοσιακά, ο μέσος χρήστης διατηρεί ένα επίπεδο εμπιστοσύνης για το υπάρχων PSTN (Δημόσιο τηλεφωνικό δίκτυο) υποθέτοντας ότι είναι ασφαλής από υποκλοπές ενώ γνωρίζουμε ότι το PSTN δεν παρέχει καμία κρυπτογράφηση για την προστασία του απορρήτου. Σε σύντομο χρονικό διάστημα όμως, η VoIP τεχνολογία θα αντικαταστήσει την PSTN (πρόβλεψη που υποστηρίζεται από τις δαπάνες των εταιρειών σε υλικό και υπηρεσίες) και αυτό θα δώσει τη δυνατότητα σε κάθε επίδοξο κακόβουλο επιτιθέμενο, να προσπαθεί να "διαρρήξει" το σύστημα από την άνεση του σπιτιού του. Οι εταιρείες λοιπόν καθώς και οι οργανισμοί επενδύουν και ερευνούν όλο και πιο έντονα το θέμα της ασφάλειας διότι οι χρήστες VoIP αυξάνονται και μαζί τους αυξάνεται και η απαίτηση για ασφαλείς επικοινωνίες. Η IETF έχει κάνει αρκετές βελτιώσεις στην προστασία σηματοδότησης και ροής δεδομένων στο VoIP. Οι πιο σημαντικές συστάσεις της είναι η χρήση του TLS για την προστασία σηματοδότησης SIP και το ZRTP για την προστασία της ροής ήχου και εικόνας. Η υιοθέτηση αυτών των πρακτικών από τις εταιρείες όμως γίνεται με πιο αργό ρυθμό από ότι θα ήθελαν με εξαίρεση προγράμματα όπως το LinPhone. Επιπλέον, ορισμένοι πάροχοι υπηρεσιών VoIP έχουν συγκεχυμένη ιδέα περί ασφάλειας. Ένα παράδειγμα αυτού, που αναφέρθηκε στην αρχή του κεφ. 6, βρίσκουμε σε διαφημιστικό περίοπτου φορέα παροχής υπηρεσιών VoIP στη Βόρεια Αμερική που υποστηρίζει ότι, "*Είμαστε πιο ασφαλείς από μια κοινή τηλεφωνική γραμμή.*" ☺

Παράρτημα Α
Ακρωνύμια, Επεξηγήσεις όρων

ARP – Address Resolution Protocol (πρωτόκολλο επίλυσης διευθύνσεων) - χρησιμοποιείται για να βρεθεί μια διεύθυνση του επιπέδου συνδέσμου (link layer) ή διεύθυνση υλικού (hardware address) ενός ξένου υπολογιστή με βάση μια διεύθυνση του επιπέδου επικοινωνίας (network layer). Συναντάτε κυρίως στα πρωτόκολλα IPv4 και Ethernet.

ARP Spoofing ή ARP poisoning – Τεχνική επίθεσης MiTM όπου πλαστογραφημένα (spoofed) πακέτα ARP αποστέλλονται στα θύματα προσποιούμενα μία νέα συσχέτιση IP/MAC. Ως αποτέλεσμα, όλα τα πακέτα IP ανακατευθύνονται στην MAC-address του επιτιθέμενου.

BNF - Στη θεωρητική πληροφορική, η BNF (Backus Normal Form ή Backus–Naur Form) είναι μια τεχνική συμβολισμού (μετασύνταξη) για γραμματικές χωρίς συμφραζόμενα (context-free grammars), που συχνά χρησιμοποιείται για να περιγράψει τη σύνταξη μιας γλώσσας όπως οι γλώσσες προγραμματισμού, ή τύπους εγγράφων (document formats), ή σύνολα εντολών (instruction sets) και πρωτόκολλα επικοινωνιών. Εφαρμόζεται όπου χρειάζονται ακριβείς περιγραφές γλωσσών, για παράδειγμα σε επίσημους ορισμούς γλωσσών, σε εγχειρίδια, ή σε βιβλία για θεωρία γλωσσών προγραμματισμού. Υπάρχουν πολλές επεκτάσεις και παραλλαγές της αρχικής BNF, κάποιες από αυτές είναι αυστηρά ορισμένες, όπως η Εκτεταμένη Μορφή Μπάκους-Νάουρ (Extended Backus–Naur Form, EBNF) και η Επαυξημένη Μορφή Μπάκους-Νάουρ (Augmented Backus–Naur Form, ABNF).

Fuzz testing ή **fuzzing** - Τεχνική δοκιμής λογισμικού, συχνά αυτοματοποιημένη ή ημι-αυτοματοποιημένη, η οποία περιλαμβάνει την παροχή άκυρων, απρόσμενων, ή

τυχαίων δεδομένα ως είσοδο σε μία εφαρμογή με σκοπό να ελεγχθεί η ευστάθειά της.

Heuristic methods - Ευρετικές μέθοδοι: Ευρετική ονομάζεται κάθε μη αλγοριθμική μέθοδος επίλυσης προβλημάτων, στην οποία η πορεία προς ένα τελικό αποδεκτό αποτέλεσμα στηρίζεται σε μια σειρά προσεγγιστικών αποτελεσμάτων. Αν και οι ευρετικές μέθοδοι δίνουν απλές και ικανοποιητικές λύσεις σε μερικά προβλήματα, τίποτα δεν εγγυάται ότι αυτές οι λύσεις είναι οι καλύτερες δυνατές. Συνήθως δίνουν προσεγγίσεις των βέλτιστων λύσεων και κάποιες φορές προτιμώνται επειδή δίνουν αποδεκτές απαντήσεις σε μικρό χρόνο. Συνεπώς δεν μπορούν να αποτελέσουν κύριο εργαλείο βελτιστοποίησης.

Jitter Buffer – Χώρος προσωρινής αποθήκευσης πακέτων δεδομένων που καταφθάνουν σε μία συσκευή, προκειμένου να ελαχιστοποιηθούν διακυμάνσεις στη καθυστέρηση αφίξεως. Αν τα πακέτα φθάσουν πολύ αργά, τότε απορρίπτονται. Θέλει πολύ προσοχή στη ρύθμισή του ώστε να μην είναι ούτε πολύ μεγάλος ούτε πολύ μικρός.

MAC (Message Authentication Codes) – Κώδικας αυθεντικοποίησης μηνύματος. Πρόκειται για συναρτήσεις κατακερματισμού (hash) που χρησιμοποιούνται για να ελέγξουν την αυθεντικότητα ενός μηνύματος.

MIME - Το Multipurpose Internet Mail Extensions (MIME) είναι ένα πρότυπο δικτύου για την ηλεκτρονική αλληλογραφία. Σχεδόν όλο το ηλεκτρονικό ταχυδρομείο του διαδικτύου διαβιβάζεται μέσω SMTP σε μορφή (format) MIME. Το ηλεκτρονικό ταχυδρομείο διαδικτύου συνδέεται τόσο πολύ με τα πρότυπα SMTP και MIME ώστε μερικές φορές καλείται SMTP/MIME e-mail.

QoS (Quality of Service) – Ποιότητα της παρεχόμενης υπηρεσίας.

RTP – Real-time Transport Protocol. Παρέχει λειτουργίες μεταφοράς δικτύου end-to-end κατάλληλο για εφαρμογές μετάδοσης δεδομένων σε πραγματικό χρόνο, όπως δεδομένα ήχου, βίντεο ή προσομοίωση, πάνω από multicast ή unicast υπηρεσίες δικτύου.

SCTP – Stream Control Transmission Protocol (Πρωτόκολλο ελέγχου μετάδοσης ροών δεδομένων). Θεωρείται ως ο διάδοχος του TCP και είναι σε θέση να μεταφέρει ταυτόχρονα πολλές ροές (streams) δεδομένων μεταξύ δύο σημείων.

SDP – Session Description Protocol (Πρωτόκολλο περιγραφής συνόδου). Format για να περιγράψουμε τις παραμέτρους εκκίνησης των μέσων ροής (media streaming). Δημοσιεύτηκε από την IETF ως RFC 4566.

SIP - Πρωτόκολλο για κλήσεις Voice over IP. Είναι ανοικτής αρχιτεκτονικής που δημοσιεύθηκε από τον οργανισμό IETF (Internet Engineering Task Force), ο οποίος δημιούργησε τη πλειονότητα των πρωτοκόλλων διαδικτύου. Είναι εμπνευσμένο από το e-mail και το http.

SKINNY: Client Control Protocol. Ιδιόκτητο πρωτόκολλο VoIP της Cisco.

TFTP – Trivial File Transfer Protocol, μία απλούστερη μορφή του πρωτοκόλλου FTP το οποίο χρησιμοποιεί UDP επικοινωνία χωρίς ασφάλεια.

UA – User Agent, οποιαδήποτε συσκευή ή εφαρμογή η οποία μετατρέπει τις πληροφορίες από ένα δίκτυο σε μορφή κατάλληλη για τον χρήστη π.χ. IP τηλέφωνα, εφαρμογές που εκτελούνται σε υπολογιστή με δυνατότητες Multimedia ή αναλογικοί μετατροπείς (**ATA's** - Analog Telephone Adapters) που συνδέουν παραδοσιακές αναλογικές συσκευές (τηλέφωνα, φαξ), σε συστήματα VoIP.

URI - Κάθε πόρος ενός SIP δικτύου, όπως για παράδειγμα ένας user agent ή ένα voice mailbox, ταυτοποιείται από ένα Uniform resource identifier (URI), που βασίζεται σε ένα γενικό συντακτικό το οποίο χρησιμοποιείται ευρέως και σε web services και σε email. Ένα SIP URI έχει την εξής μορφή sip:username:password@host:port

War Dialler – Εφαρμογή που χρησιμοποιείται για τον εντοπισμό αριθμών τηλεφώνου οι οποίοι μπορούν με επιτυχία να συνδεθούν με μόντεμ.

X509 - πιστοποιητικά server. Στην κρυπτογραφία, το X.509 είναι ένα σημαντικό πρότυπο για μια υποδομή δημόσιου κλειδιού (public key infrastructure - PKI) για τη διαχείριση των ψηφιακών πιστοποιητικών και δημόσιου κλειδιού κρυπτογράφησης και ένα βασικό μέρος του πρωτοκόλλου Transport Layer Security (TLS) που χρησιμοποιείται για να ασφαλίσει τις επικοινωνίες στο web και στα e-mail. Το X.509 είναι ένα πρότυπο ITU-T και καθορίζει το format για τα πιστοποιητικά δημόσιου κλειδιού, τις λίστες ανάκλησης πιστοποιητικών, και έναν αλγόριθμο επαλήθευσης της διαδρομής πιστοποίησης.

Παράρτημα Β
Λογισμικό

Παρατίθεται ο κώδικας που δημιουργήθηκε για την δημιουργία VPN συνδέσεων, την κλήση και την αποσύνδεση από αυτήν. Ο κώδικας δημιουργήθηκε σε C# με το Visual Studio 2015, χρησιμοποιώντας τη βιβλιοθήκη RASDial.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using System.Configuration;
using System.Reflection;

using DotRas;

namespace client
{
 public partial class frmMain : Form
 {
 public frmMain()
 {
 InitializeComponent();
 ReadFromConfig();
 }

 private RasHandle handle = null;

 private delegate void CrossThreading(object obj, StateChangedEventArgs sc_event);

 private void cmdCreateVPN_Click(object sender, EventArgs e)
 {

 try
 {
 this.rasPhoneBook1.Open(true);

 RasEntry entry;

 if (comboBox1.SelectedIndex == 0)
 {
 entry = RasEntry.CreateVpnEntry(textBox1.Text, textBox4.Text,
RasVpnStrategy.PptpFirst, RasDevice.GetDeviceByName("(PPTP)", RasDeviceType.Vpn),
false);
```

```
 // Add the new entry to the phone book.
 this.rasPhoneBook1.Entries.Add(entry);
 }
 else if (comboBox1.SelectedIndex == 1)
 {
 entry = RasEntry.CreateVpnEntry(textBox1.Text, textBox4.Text,
RasVpnStrategy.L2tpFirst, RasDevice.GetDeviceByName("(L2TP)", RasDeviceType.Vpn),
false);
 this.rasPhoneBook1.Entries.Add(entry);
 }
 else if (comboBox1.SelectedIndex == 2)
 {
 entry = RasEntry.CreateVpnEntry(textBox1.Text, textBox4.Text,
RasVpnStrategy.IkeV2First, RasDevice.GetDeviceByName("(IKEV2)", RasDeviceType.Vpn),
false);
 this.rasPhoneBook1.Entries.Add(entry);
 }
 else if (comboBox1.SelectedIndex == 3)
 {
 entry = RasEntry.CreateVpnEntry(textBox1.Text, textBox4.Text,
RasVpnStrategy.SstpFirst, RasDevice.GetDeviceByName("(SSTP)", RasDeviceType.Vpn),
false);
 this.rasPhoneBook1.Entries.Add(entry);
 }


 this.txtStatus.AppendText("Connection created" + "\r\n");
 }

 catch (Exception ex)
 {
 this.txtStatus.AppendText(ex.ToString() + "\r\n");
 }
 }

 private void cmdDialVPN_Click(object sender, EventArgs e)
 {
 this.rasDialer1.EntryName = textBox1.Text ;
 this.rasDialer1.PhoneBookPath = RasPhoneBook.GetPhoneBookPath(RasPhoneBookType.User);

 try
 {
 // Set the credentials the dialer should use.
 this.rasDialer1.Credentials = new System.Net.NetworkCredential(textBox2.Text,
textBox3.Text );

 // NOTE: The entry MUST be in the phone book before the connection can be dialed.
 // Begin dialing the connection; this will raise events from the dialer instance.
 this.handle = this.rasDialer1.DialAsync();
 }
 catch (Exception ex)
 {
 this.txtStatus.AppendText(ex.ToString() + "\r\n");
 }
 }

 private void rasDialer1_StateChanged(object sender, StateChangedEventArgs e)
 {
 if (this.InvokeRequired)
 {
 this.Invoke(new CrossThreading(rasDialer1_StateChanged), new object[] { sender, e });
 }
 else
 {
 this.txtStatus.AppendText(string.Format("{0}\r\n", e.State.ToString()));
```

```
}
}

private void rasDialer1_DialCompleted(object sender, DialCompletedEventArgs e)
{
if (e.Cancelled)
{
this.txtStatus.AppendText("Canceled");
}
else if (e.TimedOut)
{
this.txtStatus.AppendText("Connection attempt timed out");
}
else if (e.Error!=null)
{
this.txtStatus.AppendText(e.Error.ToString());
}
}

private void cmdHangUp_Click(object sender, EventArgs e)
{
if (this.rasDialer1.IsBusy)
{
// The connection attempt has not been completed, cancel the attempt.
this.rasDialer1.DialAsyncCancel();
}
else
{
if (this.handle != null)
{
// The connection attempt has completed, attempt to find the connection in the active
connections.
RasConnection connection = RasConnection.GetActiveConnectionByHandle(this.handle);
if (connection != null)
{
// The connection has been found, disconnect it.
connection.HangUp();
this.txtStatus.AppendText("Connection Disconected" + "\r\n");
}
}
else
{ this.txtStatus.AppendText("There is NO active connection" + "\r\n"); }
}
}

private void button1_Click(object sender, EventArgs e)
{

Properties.Settings.Default.setVPNname = textBox1.Text;
Properties.Settings.Default.setUserName = textBox2.Text;
Properties.Settings.Default.setPassword = textBox3.Text;
Properties.Settings.Default.setServerIP = textBox4.Text;
Properties.Settings.Default.Save();
}

private void ReadFromConfig()
{
textBox1.Text = Properties.Settings.Default.setVPNname;
textBox2.Text = Properties.Settings.Default.setUserName;
textBox3.Text = Properties.Settings.Default.setPassword;
textBox4.Text = Properties.Settings.Default.setServerIP;
}

private void cmdDeleteVPN_Click(object sender, EventArgs e)
{
```

```
RasDialer dialer = new RasDialer();
RasPhoneBook allUsersPhoneBook = new RasPhoneBook();
try
{
allUsersPhoneBook.Open();
if (allUsersPhoneBook.Entries.Contains(textBox1.Text))
{
allUsersPhoneBook.Entries.Remove(textBox1.Text);
}
}
catch (Exception ex)
{
this.txtStatus.AppendText(ex.ToString() + "\r\n");
}

}

}
}
```

Και η κλάση classVPNRas που κάνει όλη τη δουλειά

```
using System;
using System.Collections.Generic;
using System.Text;
using System.Diagnostics;
using DotRas;
using System.Text.RegularExpressions;
namespace XXX
{
 public class classVPNRas
 {
 // path C:\windows\system32\
 private static string WinDir =
Environment.GetFolderPath(Environment.SpecialFolder.System) + @"\";
 // rasdial.exe
 private static string RasDialFileName = "rasdial.exe";
 // VPN path C:\windows\system32\rasdial.exe
 private static string VPNPROCESS = WinDir + RasDialFileName;
 // VPN address
 public string IPToPing { get; set; }
 // VPN
 public string VPNName { get; set; }
 // VPN User name
 public string UserName { get; set; }
 // VPN password
 public string PassWord { get; set; }

 public classVPNRas()
 {
 }
 /// <summary>
 /// with parameter constructor
 /// </summary>
 /// <param name="_vpnIP"></param>
 /// <param name="_vpnName"></param>
 /// <param name="_userName"></param>
 /// <param name="_passWord"></param>
 public classVPNRas(string _vpnIP, string _vpnName, string _userName, string
_passWord)
 {
 this.IPToPing = _vpnIP;
 this.VPNName = _vpnName;
 this.UserName = _userName;
 this.PassWord = _passWord;
 }
 /// <summary>
 /// attempts to connect VPN( default VPN)
 /// </summary>
 /// <returns></returns>
 public void TryConnectVPN()
 {
 this.TryConnectVPN(this.VPNName, this.UserName, this.PassWord);
 }
 /// <summary>
 /// tries to disconnect the default create or update a default ( connection VPN)
 /// </summary>
 /// <returns></returns>
 public void TryDisConnectVPN()
 {
 this.TryDisConnectVPN(this.VPNName);
```

```
}
/// <summary>
/// tried to delete the default VPN attempts to connect
/// </summary>
public void CreateOrUpdateVPN()
{
this.CreateOrUpdateVPN(this.VPNName, this.IPToPing);
}
/// <summary>
/// specified ( name VPN)
/// </summary>
/// <returns></returns>
public void TryDeleteVPN()
{
this.TryDeleteVPN(this.VPNName);
}
/// <summary>
/// username VPN( password VPN tries to disconnect ,  specifies the name of the ,  )
/// </summary>
/// <returns></returns>
public void TryConnectVPN(string connVpnName, string connUserName, string
connPassWord)
{
try
{
string args = string.Format("{0} {1} {2}", connVpnName, connUserName, connPassWord);
ProcessStartInfo myProcess = new ProcessStartInfo(VPNPROCESS, args);
myProcess.CreateNoWindow = true;
myProcess.UseShellExecute = false;
Process.Start(myProcess);
}
catch (Exception Ex)
{
Debug.Assert(false, Ex.ToString());
}
}
/// <summary>
/// create or update a VPN( specify VPN connection name )
/// </summary>
/// <returns></returns>
public void TryDisConnectVPN(string disConnVpnName)
{
try
{
string args = string.Format(@"{0} /d", disConnVpnName);
ProcessStartInfo myProcess = new ProcessStartInfo(VPNPROCESS, args);
myProcess.CreateNoWindow = true;
myProcess.UseShellExecute = false;
Process.Start(myProcess);

}
catch (Exception Ex)
{
Debug.Assert(false, Ex.ToString());
}
}
/// <summary>
/// connect VPN ( connect VPN ,  And IP)
/// </summary>
public void CreateOrUpdateVPN(string updateVPNname, string updateVPNip)
{
RasDialer dialer = new RasDialer();
RasPhoneBook allUsersPhoneBook = new RasPhoneBook();
allUsersPhoneBook.Open(true);
// if has the name of the VPN existing ,  update the VPN server address
```

```
 if (allUsersPhoneBook.Entries.Contains(updateVPNname))
 {
 allUsersPhoneBook.Entries[updateVPNname].PhoneNumber = updateVPNip;
 // regardless of the current VPN is connected ,  the address of the server update
always successful ,  if connecting to ,  need VPN after the restart to
 allUsersPhoneBook.Entries[updateVPNname].Update();
 }
 // created a new VPN
 else
 {
 RasEntry entry = RasEntry.CreateVpnEntry(updateVPNname, updateVPNip,
RasVpnStrategy.PptpFirst, RasDevice.GetDeviceByName("(PPTP)", RasDeviceType.Vpn));
 allUsersPhoneBook.Entries.Add(entry);
 dialer.EntryName = updateVPNname;
 dialer.PhoneBookPath = RasPhoneBook.GetPhoneBookPath(RasPhoneBookType.AllUsers);
 }
 }
 /// <summary>
 /// delete the specified name VPN
 /// if VPN running ,  will delete in the phone book ,  But don't break open connection
,  so ,  is best to disconnect ,  and delete operation
 /// </summary>
 /// <param name="delVpnName"></param>
 public void TryDeleteVPN(string delVpnName)
 {
 RasDialer dialer = new RasDialer();
 RasPhoneBook allUsersPhoneBook = new RasPhoneBook();
 allUsersPhoneBook.Open();
 if (allUsersPhoneBook.Entries.Contains(delVpnName))
 {
 allUsersPhoneBook.Entries.Remove(delVpnName);
 }
 }
 /// <summary>
 /// gets the current connection is in the VPN name
 /// </summary>
 public List<string> GetCurrentConnectingVPNNames()
 {
 List<string> ConnectingVPNList = new List<string>();
 Process proIP = new Process();
 proIP.StartInfo.FileName = "cmd.exe ";
 proIP.StartInfo.UseShellExecute = false;
 proIP.StartInfo.RedirectStandardInput = true;
 proIP.StartInfo.RedirectStandardOutput = true;
 proIP.StartInfo.RedirectStandardError = true;
 proIP.StartInfo.CreateNoWindow = true;// does not display window cmd
 proIP.Start();
 proIP.StandardInput.WriteLine(RasDialFileName);
 proIP.StandardInput.WriteLine("exit");
 // command - line running results
 string strResult = proIP.StandardOutput.ReadToEnd();
 proIP.Close();
 // using regular expressions matching command line results ,  only limited to
simplified Chinese system Oh ^_^
 Regex regger = new Regex("( <= is connected \r\n)(.*\n)*( = command has completed )",
RegexOptions.Multiline);
 // if matching ,  said that is trying to connect to the VPN
 if (regger.IsMatch(strResult))
 {
 string[] list = regger.Match(strResult).Value.ToString().Split('\n');
 for (int index = 0; index < list.Length; index++)
 {
 if (list[index] != string.Empty)
 ConnectingVPNList.Add(list[index].Replace("\r", ""));
```

```
 }
 }
 // No VPN,  is trying to connect to the returns an empty List<string>
 return ConnectingVPNList;
 }
 }
}
```

Ο κώδικας (full project με βιβλιοθήκες) είναι διαθέσιμος στο:
https://mega.nz/#!ANtFlKAJ!sSoPE1M2jQJJ2vyErZS736X2hFwxuY8LWiUD6cvweHM

Καθώς και στο
http://cardware.gr/downloads/vpn_lite.zip

Βιβλιογραφία - Αναφορές

[1] Jeffrey Albers, Bradley Hahn, Shawn McGann, "An Analysis of Security Threats and Tools in SIP-Based VoIP Systems" Seungwoo Park, Rundong Zhu M.Sc. Capstone Paper, 2005. https://pdfs.semanticscholar.org/dab5/fa646f6b066d7a97bece85a7fd49385a2d98.pdf

[2] J. E. Canavan, "Fundamentals of Network Security", Boston: Artech House, 2001 http://www.eureka.com.ve/libros/Network-Administration/Artech.House.Fundamentals.Of.Network.Security.Feb.2001.ISBN.1580531768.pdf

[3] D.R. Kuhn, T. J. Walsh and S. Fries, "Special Publication 800-58: Security Considerations for Voice Over IP Systems," NIST, Jan. 2005; http://csrc.nist.gov/publications/nistpubs/800-58/SP800-58-final.pdf

[4] N. Dadoun, Security Framework for IP Telephony, tech. report TR-41.4.4, TR-41.4, Polycom, 15 Feb. 2002; http://ftp.tiaonline.org/TR-41/TR-41_InactiveArchive/TR4144inactive/Public/2002-02-Vancouver/TR41.4.4-02-02-008SecurityFrameworknd.pdf

[5] J. Thalhammer, "*Security in VoIP-Telephony Systems",* master's thesis, Graz Univ. of Technology,2002. https://pdfs.semanticscholar.org/d8c3/037a70c3f6bfd4997487bb22534f69128087.pdf

[6] M. Thomas, "SIP Security Requirements", IETF Internet draft, work in progress, Nov. 2001. https://tools.ietf.org/html/draft-thomas-sip-sec-req-00

[7] Yu-Sung Wu, S. Bagchi, S. Garg and N. Singh, "*SCIDIVE:A Stateful and Cross Protocol Intrusion Detection Architecture for Voice-over-IP Environments*", Conf. on Dependable Systems and Networks, Proc. of the 2004 Int'l Conf. on Dependable Systems and Networks (DSN'04), Jun.-July. 2004, pp. 433 - 442. http://ieeexplore.ieee.org/document/1311913/?reload=true

[8] Si DF, Long Q, Han XH and Zou W, "Security mechanisms for SIP-based multimedia communication infrastructure" IEEE Conf. on Comm, Circuits and Systems (ICCCAS), ed. Proc. of 2nd ed., IEEE CS Press, 27-29 June 2004, pp.575-578. http://www.academia.edu/2106556/SIP_Security_Mechanism_Techniques_on_Voice_over_Internet_Protocol_VoIP_System

[9] S. Salsano, L. Veltri and D. Papalilo, "*SIP Security Issues: The SIP Authentication Procedure and its Processing Load*", IEEE Network, vol. 16, no. 6, Nov./Dec. 2002, pp.38- 44. http://ieeexplore.ieee.org/document/1081764/

[10] L. McKeag, "*How to cope with ARP attacks on LANs*", Techworld, 20 July. 2004; http://www.techworld.com/security/features/index.cfm?featureid=727&Page=2&pagePos=326

[11] R. Spangler, "*Packet Sniffing on Layer 2 Switched Local Area Networks*," Packetwatch Research, Dec. 2003, pp.1; http://www.packetwatch.net/documents/papers/layer2sniffing.pdf

[12] C. Kaufman, R. Perlman and B. Sommerfeld, "*DoS Protection for UDP- Based Protocols*", Conf. on Computer and Comm. Security, Proc. of the 10th ACM Conf. on Computer and Comm. security, Washington DC, 2003, pp. 2-7. https://www.eecis.udel.edu/~mills/teaching/eleg867b/dos/p2-kaufman.pdf

[13] Ryan Angelo, " *The Anatomy of a Distributed Denial of Service Attack (DDoS). How Zombies Can Take Down Giants*", 2009, http://docplayer.net/19515454-The-anatomy-of-a-distributed-denial-of-service-attack-ddos-how-zombies-can-take-down-giants.html

[14] "*Intrusion Prevention: The Future of VoIP Security*", Tech whitepaper, Tipping Point, 2004. http://www.preferredtechnology.com/files/VoIPSecurity.pdf

[15] I. Dubrawsky, "*Safe Layer 2 Security In-Depth*," Cisco, 2004, pp.12 http://www.cisco.com/warp/public/cc/so/cuso/epso/sqfr/sfblu_wp.pdf

[16] "*VoIp Security Tools*", http://www.voipsa.org/Resources/tools.php

[17] The Internet Society, 2002, RFC 3261 "*SIP: Session Initiation Protocol*" http://www.cs.columbia.edu/sip/drafts/rfc3261.pdf

[18] "*Understanding SIP. Today's Hottest Communications Protocol Comes of* Age", White Paper, Ubiquity Software Corporation, 2003. http://www.sipforum.org/component/option,com_docman/task,doc_view/gid,16/Itemid,75/

[19] "*Session Initiation Protocol*", Telecom Space, http://www.telecomspace.com/voip-sip.html

[20] "*Session Initiation Protocol Tutorial. SIP - Network Elements*", https://www.tutorialspoint.com/session_initiation_protocol/session_initiation_protocol_network_elements.htm

[21] "*Secured Communications using Linphone & Flexisip. Solution description"*, Belledonne communications. http://www.belledonne-communications.com/uploads/images/Solutions-SecuredCommunications.pdf

[22] T. Dierks; E. Rescorla (August 2008). "*The Transport Layer Security (TLS) Protocol, Version 1.2*". https://tools.ietf.org/html/rfc5246

[23] John Wagnon "*SSL Profiles. Part 8: Client Authentication.*", 2013 https://devcentral.f5.com/articles/ssl-profiles-part-8-client-authentication

[24] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, "*RTP: A Transport Protocol for Real-Time Applications*", Network Working Group. 2003. https://www.ietf.org/rfc/rfc3550.txt

[25] RSA Labs. “Public-Key Cryptography Standards (PKCS)”. Chapter 3.6.1 “What is Diffie-Hellman?”, http://www.emc.com/emc-plus/rsa-labs/standards-initiatives/what-is-diffie-hellman.htm

[26] Baugher, M., McGrew, D., et al.: RFC3711, “*The Secure Realtime Transport Protocol (SRTP)*”, IETF, March 2004, https://tools.ietf.org/html/rfc3711

[27] Diffie, W., Hellman, M. E.: “*New directions in cryptography, IEEE Transactions on Information Theory*”, IT-22(6):644-654, November 1976 http://www.scirp.org/(S(lz5mqp453edsnp55rrgjct55))/reference/ReferencesPapers.aspx?ReferenceID=702910

[28] Rosenberg, J., Schulzrinne, H., et al.: RFC3261, “SIP: Session Initiation Protocol”, IETF, June 2002, https://www.ietf.org/rfc/rfc3261.txt

[29] Handley, M., Jacobson, V., Perkins, C.: RFC 4566, “SDP: Session Description Protocol”, IETF, July 2006, https://tools.ietf.org/html/rfc4566

[30] O’ Whielacronx, Z.: “Human-oriented base-32 encoding”, 2002,2007,2009 https://philzimmermann.com/docs/human-oriented-base-32-encoding.txt

[31] Juola, P., Zimmermann, P.: “*Whole-Word Phonetic Distances and the PGPfone Alphabet, Proceedings of the International, Conference of Spoken Language Processing*”, 1996 https://www.researchgate.net/publication/3703307_Whole-word_phonetic_distances_and_the_PGPfone_alphabet

[32] PGP Word List, https://en.wikipedia.org/wiki/PGP_word_list

[33] NATO–φωνητικό αλφάβητο, https://en.wikipedia.org/wiki/NATO_phonetic_alphabet

[34] Whalen, S., et al.: “*An Introduction to ARP Spoofing*”, rootsecure.net, April 2001 http://machacking.net/kb/files/arpspoof.pdf

[35] Rescorla, E., McGrew, D., Fischl, J., Tschofenig, H.: "*DTLSSRTP, Proceedings of the 68th Internet Engineering Task Force",* Prague, Czech Republic, 2007 https://www.rfc-editor.org/pdfrfc/rfc7879.txt.pdf

[36] Regis J. Bates, "*Securing Voip*", SyngRess, ch.8 Approaches to voip security. 2014 https://www.safaribooksonline.com/library/view/securing-voip/9780124170391/B9780124170391000085/B9780124170391000085.xhtml

[37] Y. Sheffer, R. Holz : RFC7457: *"Summarizing Known Attacks on Transport Layer Security (TLS) and Datagram TLS (DTLS)*", February 2015. https://tools.ietf.org/html/rfc7457

[38] Alexander Sotirov, Marc Stevens, Jacob Appelbaum, Arjen Lenstra, David Molnar, Dag Arne Osvik, Benne de Weger, "*Creating a rogue CA certificate*", December 2008 https://www.win.tue.nl/hashclash/rogue-ca/

[39] M. Ray, S. Dispensa, "*Transport Layer Security (TLS) Renegotiation Indication Extension*", RFC 5746, February 2010 https://tools.ietf.org/html/rfc5746

[40] Red Team Pentesting, "*TLS Renegotiation Vulnerability*", 2009 https://www.redteam-pentesting.de/en/publications/tls-renegotiation/-tls-renegotiation-vulnerability-proof-of-concept-code

[41] Nikos Mavrogiannopoulos, Frederik Vercauteren, VesselinVelichkov, Bart Preneel, "*A Cross-Protocol Attack on the TLS protocol*." ESAT/SCD/COSIC-IBBT. http://www.cs.usfca.edu/~ejung/courses/f12683/presentations/Paper2.pdf

[42] Fatih Ozavci, "*Viproy, VoIP Penetration Testing and Exploitation Kit*", http://www.viproy.com/

[43] Sandro Gauci, "*SipVicious, SIP Penetration Testing and Exploitation Kit*", http://blog.sipvicious.org

[44] SipVicious, Συνέντευξη του Sandro Gauci σε κορεάτικο site ειδήσεων τεχνολογίας (ελεύθερη μετάφραση), http://blog.sipvicious.org/2012/12/if-sipvicious-gives-you-ring.html

[45] Cain & Abel, password recovery tools, http://www.oxid.it/projects.html

[46] Phil Zimmermann, "How To Get Phone Companies To Just Say No To Wiretapping", Ομιλία στο DEF CON 22, https://www.youtube.com/watch?v=HuHm1vzzm1g

[47] A. Critelli, "*Hacking VoIP – Decrypting SDES Protected SRTP Phone Calls*", 2014. https://www.acritelli.com/hacking-voip-decrypting-sdes-protected-srtp-phone-calls/

[48] Martin Petraschek, Thomas Hoeher, Oliver Jung, Helmut Hlavacs, Wilfried Gansterer, "*Security and Usability Aspects of Man-in-the-Middle Attacks on ZRTP*", Journal of Universal Computer Science, vol. 14, no. 5, Μάρτιος 2008 http://www.jucs.org/jucs_14_5/security_and_usability_aspects

[49] Spiegel online, "*Inside the NSA's War on Internet Security*", Δεκέμβριος 2014, http://www.spiegel.de/international/germany/inside-the-nsa-s-war-on-internet-security-a-1010361.html

[50] Mike Masnick, "*How The NSA Works Hard To Break Encryption Any Way It Can*", Δεκέμβριος 2014, https://www.techdirt.com/articles/20141229/06331329532/how-nsa-works-hard-to-break-encryption-any-way-it-can.shtml

[51] "*Transport Layer Security*", Wikipedia, 2016, https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_1.3_.28draft.29

[52] Matt Braithwaite, "*Experimenting with Post-Quantum Cryptography*", Software Engineer, Ιούλιος 2016, https://security.googleblog.com/2016/07/experimenting-with-post-quantum.html

[53] Scott Lowe, "Factors that can boost VPN performance", TechRepublic, Δεκέμβριος 2002, http://www.techrepublic.com/article/factors-that-can-boost-vpn-performance/

[54] SipVicious, Detecting SIP attacks with Snort,
http://blog.sipvicious.org/2008/02/detecting-sip-attacks-with-snort.html