ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ ΕΦΑΡΜΟΓΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΤΕΧΝΙΚΕΣ ΔΙΑΧΕΙΡΙΣΗΣ ΚΛΕΙΔΙΩΝ ΣΕ ΠΕΡΙΒΑΛΛΟΝ ΥΠΟΛΟΓΙΣΤΙΚΗΣ ΝΕΦΟΥΣ

Του φοιτητή
ΑΝΔΡΕΑ ΠΑΠΑΔΟΠΟΥΛΟΥ
Αρ. Μητρώου: 06/3143

Επιβλέπων καθηγητής
κ. ΗΛΙΟΥΔΗΣ ΧΡΗΣΤΟΣ

Θεσσαλονίκη 2013

ΠΕΡΙΛΗΨΗ

Η εργασία έχει θέμα τη διανομή κλειδιού σε συστήματα νέφους, και ο σκοπός της είναι να διερευνήσει εάν μπορεί να υλοποιηθεί κάποιο καινούριο μοντέλο διανομής κλειδιού που να παρέχει ένα αποδεκτό επίπεδο ασφάλειας και βελτίωση στη κατανάλωση χρόνου σε σχέση με υπάρχουσες τεχνικές που χρησιμοποιούνται στο κλάδο των επιστημονικών εφαρμογών. Συγκεκριμένα διερευνάται η τροπολογία της υλοποίησης κάποιων αλγορίθμων γύρω από την κρυπτογραφία και τη διανομή κλειδιού, και κάποιων σεναρίων επιθέσεων προς αυτές.

Στη παρούσα εργασία περιγράφεται η έννοια cloud, τι είναι, τι αλλαγές θα επιφέρει και με πιο κόστος. Στη συνέχεια περιγράφεται η έννοια της κρυπτογραφίας(συμμετρική, ασύμμετρη, δημοσίου κλειδιού) και αναφέρονται οι επικρατέστερες τεχνικές και αλγόριθμοι που χρησιμοποιούνται. Επίσης αναλύθηκαν κάποια μοντέλα διανομής κλειδιού που έχουν υλοποιηθεί στα υπάρχουσα δίκτυα, και ένα καινοφανή σχήμα διανομής κλειδιού για επιστημονικές εφαρμογές. Στο τέλος περιγράφεται ένα πείραμα από το οποίο προβήκανε κάποια συμπεράσματα για το αν μπορεί να μειωθεί η κατανάλωση χρόνου για συγκεκριμένες περιπτώσεις.

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ : cloud computing, cryptography, key distribution, νέφος, κρυπτογραφία, διανομή κλειδιού, IKE,CCBKE.

ΠΕΡΙΕΧΟΜΕΝΑ

Ευρετήριο εικόνων

ΕΙΣΑΓΩΓΗ

Η τεχνολογία του cloud computing είναι ένα καινούριο κεφάλαιο στο κόσμο του διαδικτύου που φαίνεται ότι σύντομα θα κυριαρχήσει στο χώρο. Με την υιοθέτηση της τεχνολογίας αυτής, η κοινωνία του διαδικτύου έχει να αποκομίσει πολλά πλεονεκτήματα που καλύπτουν τις ανάγκες του χώρου σήμερα. Παρόλα αυτά δημιουργούνται κάποια κενά κατά την υλοποίησή του, τα οποία έρχονται αντιμέτωπα με διάφορες επιστημονικές ομάδες ανά το παγκόσμιο για την εύρεση λειτουργικών λύσεων. Ένα σημαντικό πρόβλημα που έρχεται αντιμέτωπη η επιστήμη, είναι η ασφάλεια των δεδομένων και η διαφύλαξη της ασφαλούς επικοινωνίας μέσα στη καινούρια τεχνολογία. Επίσης, η φιλοξενία επιστημονικών εφαρμογών σε υβριδικά περιβάλλοντα, όπως είναι το νέφος, δείχνει να συναντά κάποια σημαντικά προβλήματα κατά την υλοποίηση της.

Οι επιστημονικές εφαρμογές έχουν να αντιμετωπίσουν υψηλού ρίσκου και μεγάλου όγκου δεδομένα. Η μεταφορά, αποθήκευση, και επεξεργασία των δεδομένων, είναι χρονοβόρα, και ο χρόνος στις επιστημονικές ανακαλύψεις είναι ένας ζωτικός παράγοντας. Για αυτό τον λόγω η επιστήμη πρέπει να ερευνά και να ανακαλύπτει συνεχώς τρόπους να μειώνει την κατανάλωση του χρόνου στο ελάχιστο για να περιορίσει τα περιστατικά αργοπορημένης ή και αποτυχίας μίας επιστημονικής ανακάλυψης.

Αντικείμενο της έρευνας αποτελούν τα μοντέλα διαχείρισης διανομής κλειδιού σε συστήματα νέφους. Ο σκοπός της ερευνάς αυτής είναι η περιγραφή του νέφους σαν έννοια, η περιγραφική αναφορά των πιο διαδεδομένων τεχνικών κρυπτογράφησης και η ανάλυση των σημαντικότερων μοντέλων διανομής κλειδιών σε υβριδικά περιβάλλοντα όπως είναι το νέφος, και ειδικότερα για τη περίπτωση των επιστημονικών εφαρμογών. Η εργασία αυτή αποσκοπεί στην διερεύνηση των τεχνικών κρυπτογραφίας και διανομής κλειδιού, οι οποίες είναι κατάλληλες για χρήση στο cloud computing και στις επιστημονικές εφαρμογές.

Οι κυριότερες πηγές από τις οποίες πάρθηκαν οι περισσότερες πληροφορίες για την υλοποίηση της εργασίας αυτής είναι το Wikipedia, επίσημα έγγραφα από τα αρχεία του NIST(Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας της Αμερικής), και από σχετικές έρευνες πανεπιστημίων Υ.Π.Α και Αυστραλίας.

Η παρούσα εργασία χωρίζεται σε τέσσερα μέρη. Στο πρώτο μέρος, το οποίο αποτελεί το γενικό μέρος της πτυχιακής, αναφέρεται στη έννοια του cloud computing και γενικότερα στο τι είναι νέφος. Επίσης αναφέρονται τα πλεονεκτήματα που θα κληρονομηθούν καθώς και τα μειονεκτήματα, αφού

φαίνεται να δημιουργούνται κάποια θέματα, κυρίως ασφαλείας, κατά την υιοθέτηση του cloud computing.

Στο δεύτερο μέρος περιγράφεται η έννοια της κρυπτογραφίας η οποία έρχεται να δώσει λύσεις σε διαφόρου τύπου επιθέσεων που έρχονται αντιμέτωπα τα συστήματα δικτύωσης. Περιγράφεται αναλυτικά ο τρόπος λειτουργίας των πιο διαδεδομένων αλγορίθμων κρυπτογραφίας, στη συμμετρική-ασύμμετρη κρυπτογραφία, στη κρυπτογραφία δημοσίου κλειδιού και στις συναρτήσεις κατακερματισμού. Επίσης αναλύονται οι αρχιτεκτονικές και ο τρόπος λειτουργίας των πιο διαδεδομένων πρωτοκόλλων ασφαλείας (SSL, SSH). Στο τέλος του δεύτερου μέρους συγκροτείται ένας πίνακας με τα υπέρ και τα κατά των αλγορίθμων και των τεχνικών κρυπτογραφίας, και εξετάζεται η καταλληλότητα τους για χρήση στο cloud.

Στο τρίτο μέρος της εργασίας παρουσιάζονται αναλυτικά τα υπάρχουσα μοντέλα διανομής κλειδιών IKE, PKI, και το καινοφανής μοντέλο CCBKE. Περιγράφεται ο τρόπος λειτουργίας των μοντέλων βήμα προς βήμα και γίνεται ανάλυση στο επίπεδο ασφάλειας τους με υποθετικά σενάρια επιθέσεων. Στο τέλος σχηματίζεται ένας πίνακας με τα υπέρ και τα κατά των μοντέλων διανομής κλειδιού, και εξετάζεται η καταλληλότητα τους για χρήση στο cloud.

Στο τέταρτο μέρος της εργασίας, περιγράφεται ένα πείραμα το οποίο υλοποιήθηκε, και παρουσιάζει τη σύγκριση δύο τεχνικών διανομής κλειδιού σε συστήματα cloud network για επιστημονικές εφαρμογές με στόχο την επαλήθευση κάποιων υπολογισμών για τη μείωση στη κατανάλωση χρόνου. Επίσης, στο τέλος της εργασίας, περιλαμβάνονται κάποια συμπεράσματα και προτάσεις για μελλοντικές επεκτάσεις περί του θέματος των επιστημονικών εφαρμογών.

ΚΕΦΑΛΑΙΟ 1. ΥΠΟΛΟΓΙΣΤΙΚΗ ΝΕΦΟΥΣ ΚΑΙ ΑΣΦΑΛΕΙΑ

ΕΙΣΑΓΩΓΗ

Στο πρώτο κεφάλαιο της εργασίας θα περιγράψουμε την έννοια της υπολογιστικής νέφους καθώς και την έννοια τις ασφάλειας στα νέφη. Θα αναφερθούμε στις υπηρεσίες που παρέχει το cloud network και σε κάποια θέματα ασφάλειας που ταλαιπωρούν και προβληματίζουν τον κύκλο της υπολογιστικής νέφους.

1.1 Βασικά χαρακτηριστικά και οφέλη της υπολογιστικής νέφους

Το cloud computing, «Υπολογιστική νέφους», είναι η παροχή υπολογιστικών πόρων μέσω ενός δικτύου υπολογιστών. Καλύτερα θα μπορούσαμε να πούμε ότι η έννοια «νέφος» αναφέρεται στη χρήση υπολογιστικής ισχύος που χωροταξικά βρίσκετε σε ένα «σύννεφο» απόμακρων δικτύων [1] . Δηλαδή ένα μεγάλο κέντρο δεδομένων που προσφέρει υπηρεσίες και πόρους του τύπου, βάσεων δεδομένων, διαφόρων εφαρμογών, υπηρεσίες αρχείων, ηλεκτρονικού ταχυδρομείου και άλλα.

Η νοοτροπία που οδήγησε στη τεχνολογία αυτή είναι, ότι με τα πιο λίγα, μπορείς να κάνεις πιο πολλά. Αυτό σημαίνει, πληρώνει κανείς μόνο για ότι χρειάζεται. Όπως και με το ρεύμα, δεν χρειάζεται να σου ανήκει η γεννήτρια για να έχεις ρεύμα. Απλά χρησιμοποιείς την πρίζα και πληρώνεις για όσο ρεύμα χρειαστείς. Το υπολογιστικό νέφος μετατρέπει την τεχνολογία της πληροφορικής σε ένα είδος κοινή ωφέλειας που καταναλώνεται.

Τα οφέλη [2] που μπορεί να αποκομίσει ένας οργανισμός-εταιρία από την υιοθέτηση ενός μοντέλου Iaas(Infrastructure As A Service) , (σύμφωνα με τους Khajeh-Hosseini οι οποίοι διερεύνησαν το οικονομικό όφελος της χρήσης των νεφών στις επιχειρήσεις) είναι ότι παρουσιάζονται πολλές ευκαιρίες να βελτιωθεί η διαχείριση των εσόδων και των εξόδων για τα εσωτερικά οικονομικά της επιχείρησης, αλλά και για τις συνδιαλλαγές της με τους πελάτες. Επίσης ευνοεί την διαχείριση της ροής του κεφαλαίου στα εσωτερικά οικονομικά της ζητήματα, καθώς το μοντέλο τιμολόγησης των υπηρεσιών νέφους έχει ελάχιστο αρχικό κόστος και μηνιαία τιμολόγηση, ενώ παράλληλα μειώνει τη μεταβλητότητα στις ανάγκες ηλεκτρισμού. Αυτά είναι τα οφέλη, συγκρινόμενα με ένα κέντρο επεξεργασίας δεδομένων εντός της επιχείρησης, καθώς μπορεί να είναι πολυέξοδη η αναβάθμιση των μηχανημάτων και του λογισμικού του. Το θέμα του κόστους της ενέργειας που κάνει χρήση, δεν αποτελεί πλέον ανησυχία, διότι περιέχεται πλέων στις ευθύνες τρίτων και δεν επιβαρύνει την επιχείρηση πια. Η δομή του νέφους

μειώνει το διοικητικό φόρτο εργασίας και για αυτό είναι πολύ βοηθητική στο οικονομικό τμήμα μιας επιχείρησης. Η μεταβίβαση σε τρίτους όλων αυτών των διαδικασιών παρέχει λύσεις οι οποίες βοηθούν στη διαχείριση των εσόδων από του πελάτες, τις πωλήσεις και το μάρκετινγκ. Οι Khajeh-Hosseini κατέληξαν ότι η υπολογιστική νέφους είναι μια αποδιοργανωτική διαδικασία, της οποίας σκοπός είναι να αλλάξει τον τρόπο που λειτουργούν τα πληροφοριακά συστήματα σε μία επιχείρηση και πως αυτά αναπτύσσονται εντός μιας επιχείρησης επειδή είναι φτηνά, απλά στη χρήση, και χαρακτηρίζονται από δυνατότητες επεκτασιμότητας. Η υπολογιστική νέφους μπορεί να είναι σημαντικά φθηνότερη συγκριτικά με την αγορά και συντήρηση ενός εσωτερικού κέντρου επεξεργασίας πληροφοριών, καθώς εξαλείφει πλήρως την ανάγκη υποστήριξής του αφού δεν υπάρχει κάποια φυσικής μορφής δομή για να συντηρηθεί.

1.1.1 Υπηρεσίες cloud

Γενικά, η «υπολογιστική νέφους» είναι ένας όρος που ασχολείται με την παροχή υπηρεσιών μέσω του διαδικτύου. Αυτές οι υπηρεσίες χωρίζονται σε 3 διακριτές κατηγορίες: [3]

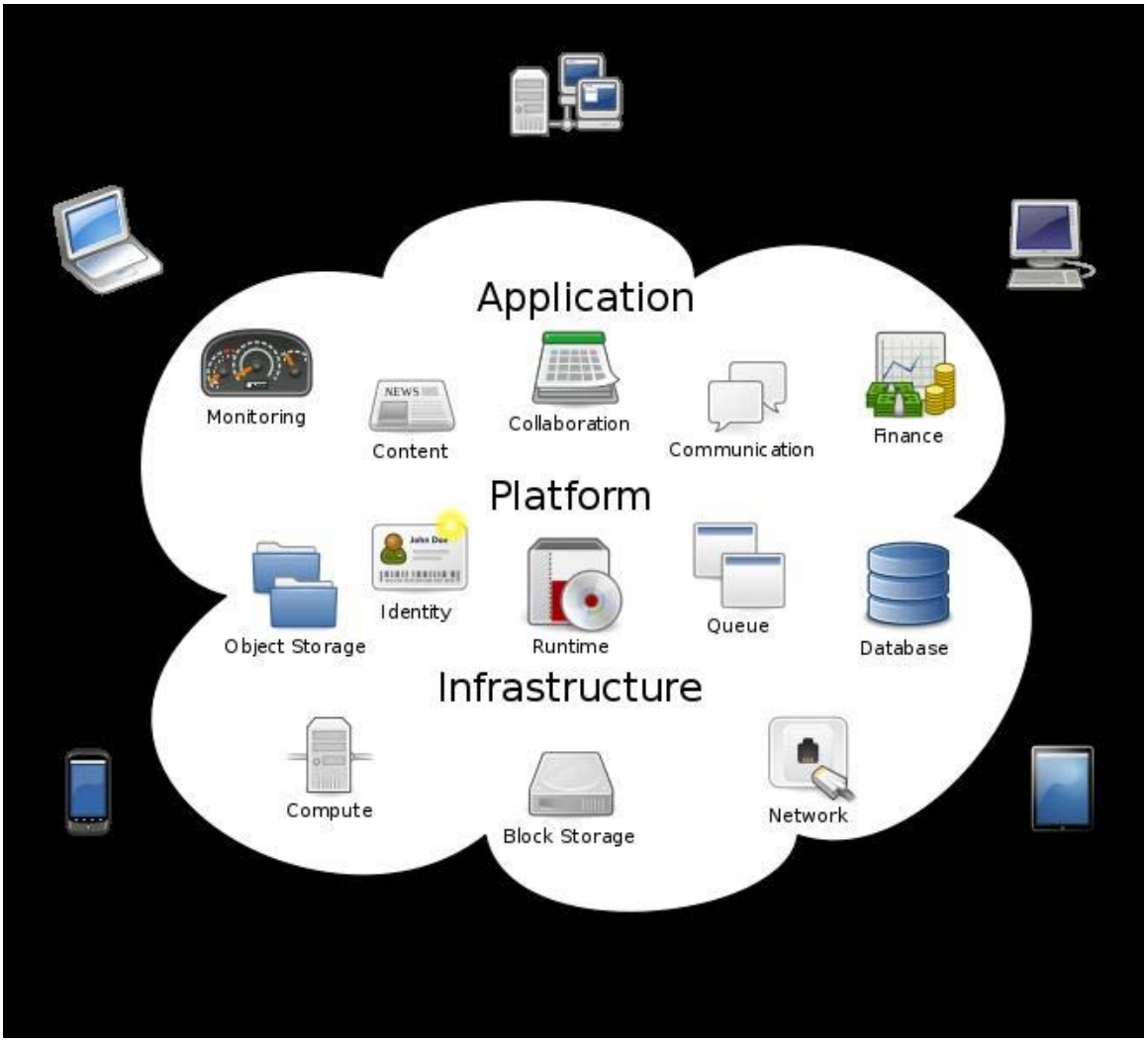
-Οι Υπηρεσίες Υποδομής (IaaS), είναι το πιο βασικό μοντέλο υπηρεσίας νέφους. Ο πάροχος προσφέρει υπολογιστές σαν φυσικές, ή πιο συχνά, σαν εικονικές μηχανές για αποθηκευτικούς χώρους, τοίχοι προστασίας, ισορροπιστές φόρτου και δικτύου. Ο πάροχος προσφέρει αυτούς τους πόρους κατά απαίτηση λόγω του μεγάλου όγκου δεδομένων που διαθέτουν τα κέντρα πληροφοριών. Τοπικά δίκτυα συμπεριλαμβανομένου και IP διευθύνσεων είναι μέρος τις υπηρεσίας αυτής. Για την ευρύτερη συνδεσιμότητα, από το διαδίκτυο, μπορεί να γίνει χρήση του VPN. Για να αναπτύξουν οι χρήστες νέφους τις εφαρμογές τους μπορούν να εγκαταστήσουν στις μηχανές αυτές λειτουργικά συστήματα επίσης και πιο εξειδικευμένα λογισμικά που τυχών να τους εξυπηρετούν. Σε αυτή την υπηρεσία, ο χρήστης είναι υπεύθυνος για την παρακολούθηση, διαχείριση και ενημέρωση των εφαρμογών. Τυπικά ο πάροχος κοστολογεί τον πελάτη βάση των πόρων που δέσμευσε και κατανάλωσε ο πελάτης.

-Η Υπηρεσία Πλατφόρμας (SaaS), σε ένα νέφος ορίζεται σαν μια ομάδα εργαλείων λογισμικού και προϊόντων στην υποδομή του παρόχου. Οι προγραμματιστές μπορούν να αναπτύξουν και να τρέξουν τα προγράμματα τους σε μία πλατφόρμα νέφους, χωρίς να έχουν εγκατεστημένα τα απαραίτητα προγράμματα τοπικά. Με αυτό τον τρόπο οι εταιρίες γλιτώνουν από το κόστος και

την πολυπλοκότητα της αγοράς και της διαχείρισης των ειδικών προγραμμάτων λογισμικού και υλικού που θα χρειαζόταν σε διαφορετική περίπτωση. Οι Force.com και GoogleApps είναι παραδείγματα υπηρεσιών πλατφόρμας.

-Στο μοντέλο νέφους Υπηρεσιών Λογισμικού (PaaS), ο πωλητής παρέχει την υποδομή υλικού, το λογισμικό προϊόν και επικοινωνεί με τον πελάτη μέσω μιας διαδικτυακής πύλης (cloud client). Οι υπηρεσίες λογισμικού είναι μια ευρεία αγορά. Οι υπηρεσίες μπορεί να είναι οτιδήποτε, από ηλεκτρονικό ταχυδρομείο μέχρι έλεγχο απογράφης και επεξεργασίας βάσεων δεδομένων. Επειδή ο πάροχος υπηρεσιών «φιλοξενεί» τόσο την εφαρμογή όσο και τα δεδομένα, ο τελικός χρήστης είναι ελεύθερος να χρησιμοποιήσει τις υπηρεσίες από οπουδήποτε.

Στην εικόνα 1 που ακολουθεί προβάλλονται οι παροχές που προσφέρει το νέφος σε κάθε υπηρεσία, και θα μας βοηθήσει να αντιληφθούμε τη σημασία του όρου «νέφος».



Εικόνα 1: υπηρεσίες του νέφους

Οι χρήστες μπορούν να έχουν πρόσβαση στο υπολογιστικό νέφος μέσο συσκευών που είναι συνδεδεμένες σε κάποιο δίκτυο. Συσκευές του τύπου υπολογιστή γραφείου, ταμπλέτας, έξυπνου κινητού τηλεφώνου, φορητού υπολογιστή. Μερικές από αυτές τις συσκευές βασίζονται στη τεχνολογία νέφους κατά την πλειοψηφία των εφαρμογών τους ή ακόμη και εξολοκλήρου σε υλικό και λογισμικό. Κατ' ουσία, αν δεν υπάρχουν εφαρμογές που να υποστηρίζουν την τεχνολογία αυτή σε μία συσκευή, το νέφος είναι παντελώς άχρηστο.

Ένα παράδειγμα ενός νεφελώδους υπολογιστή είναι ο φορητός υπολογιστής τις Google, ο chrome book ο οποίος τρέχει το λειτουργικό Google chrome OS. Πολλές εφαρμογές νέφους δεν χρειάζονται συγκεκριμένα προγράμματα στον πελάτη, αλλά χρησιμοποιούν ένα πρόγραμμα περιήγησης ιστού το οποίο αλληλεπιδρά με την εφαρμογή νέφους.

1.1.2 Τύποι-μοντέλα ανάπτυξης νέφους

Τα νέφη διαχωρίζονται σε κάποια διακριτά είδη, τα οποία καθορίζονται από τον σκοπό και το κοινό που τείνουν να εξυπηρετήσουν. Τα κυριότερα από αυτά αναφέρονται πιο κάτω. [4]

Δημόσιο νέφος (public cloud), εφαρμογές, αποθηκευτικός χώρος και άλλοι πόροι είναι διαθέσιμοι για το ευρύ κοινό από τον φορέα παροχής υπηρεσιών. Οι υπηρεσίες δημόσιου νέφους μπορεί να είναι δωρεάν ή να προσφέρονται βάση του μοντέλου «χρέωση ανάλογα με τη χρήση». Υπάρχουν περιορισμένοι φορείς παροχής στην αγορά όπως την Microsoft και την Google στις οποίες ανήκουν οι υποδομές των κέντρων πληροφοριών τους και η πρόσβαση σε αυτά γίνετε μόνο μέσο του διαδικτύου. Δεν προτείνετε η απευθείας σύνδεση στη αρχιτεκτονική δημοσίου νέφους.

Κοινοτικό νέφος (community cloud), αυτό το μοντέλο μοιράζετε τις υποδομές διαφόρων οργανισμών από μία συγκεκριμένη κοινότητα που μοιράζεται τις ίδιες «ανησυχίες» του τύπου ασφάλειας, συμμόρφωσης, δικαιοδοσίας και άλλα. Το νέφος αυτό τυγχάνει διαχείρισης είτε εσωτερικά είτε από κάποιο τρίτο πρόσωπο, και φιλοξενείτε εσωτερικά ή εξωτερικά. Τα κόστη διασκορπίζοντε στους χρήστες που είναι λιγότεροι από ένα δημόσιο νέφος, αλλά περισσότεροι από ένα ιδιωτικό. Έτσι ώστε μόνο μερικές από τες δυνατότητες εξοικονόμησης των νεφών πραγματοποιούνται.

Υβριδικό νέφος (hybrid cloud), το νέφος αυτό αποτελεί μία σύνθεση δύο ή περισσότερων νεφών (ιδιωτικών, κοινοτικών ή δημοσίων) τα οποία παραμένουν μοναδικές οντότητες, αλλά είναι δεμένα μεταξύ τους και συνεργάζονται, προσφέροντας τα οφέλη των πολλαπλών μοντέλων ανάπτυξης νεφών.

Ιδιωτικό νέφος (private cloud), το ιδιωτικό νέφος υποστηρίζει μια υποδομή που λειτουργεί για ένα και μονό ένα οργανισμό. Το μοντέλο αυτό έχει δεχτεί κριτικές από τους χρήστες επειδή, πρέπει να το αγοράσεις, να το κτήσεις και να το συντηρείς, και ως εκ τούτου δεν επωφελούνται οι οργανισμοί από τη μείωση του προσωπικού για τη διαχείριση. Κατ' ουσία, απουσιάζει το μοντέλο που κάνει το υπολογιστικό νέφος να είναι μια ενδιαφέρουσα οικονομική ιδέα.

Ενοικίαση ιδιωτικού νέφους (private cloud rental), είναι μια αποδοτική επιλογή όταν η ασφάλεια είναι ένα θέμα ανησυχίας. Οι εταιρίες μπορούν να εξετάσουν το υβριδικό νέφος για την αντικατάσταση του παλιού εξοπλισμού του κέντρου πληροφοριών. Η μετακίνηση κρίσιμων και σημαντικών ιδιωτικών δεδομένων της εταιρίας σε ένα δημόσιο νέφος, δεν αποτελεί επιλογή. Χρησιμοποιώντας εικονικές μηχανές, οι εφαρμογές μπορούν να μεταφερθούν από το παλιό κέντρο πληροφοριών στο καινούριο μισθωμένο εξοπλισμό χωρίς καμία αναστάτωση για τους πελάτες. Ο προηγούμενος εξοπλισμός του κέντρου πληροφοριών μπορεί να αφαιρεθεί και να αντικατασταθεί από καινούριο υλικό, και οι εφαρμογές που φιλοξενήθηκαν στο μισθωμένο εξοπλισμό μπορούν να επιστρέψουν στο καινούριο υλικό. Ο μισθωμένος εξοπλισμός μπορεί να χρησιμοποιηθεί σαν αποθήκη δεδομένων ή να επιστραφεί στον ενοικιαστή.

1.2 Ασφάλεια

Καθώς η υπολογιστική νέφους έχει πετύχει τον τελευταίο καιρό αυξημένη δημοτικότητα, ακούγονται ανησυχίες σχετικά με το θέμα της ασφάλειας στην υιοθεσία αυτής της καινούριας τεχνολογίας. Η αποτελεσματικότητα και η αποδοτικότητα των παραδοσιακών μηχανισμών προστασίας έχει αναθεωρηθεί αφού τα χαρακτηριστικά αυτού του καινοτόμου αναπτυξιακού μοντέλου μπορεί να διαφέρει σε μεγάλο βαθμό από τις παραδοσιακές αρχιτεκτονικές. Μια εναλλακτική άποψη για το θέμα της ασφάλειας είναι ότι πρόκειται για ακόμα μία περίπτωση «εφαρμοσμένης ασφάλειας» και πως παρόμοιες αρχές ασφάλειας μπορούν να εφαρμοστούν σε αυτό, που εφαρμοστήκαν σε άλλες περιπτώσεις όπου πολλοί χρήστες θα μπορούσαν να συνδεθούν σε ένα μεγάλο σύστημα υπολογιστή (mainframe). Η σχετική ασφάλεια των υπηρεσιών υπολογιστικής νέφους είναι ένα

επίμαχο ζήτημα που μπορεί να καθυστερεί την διαδικασία υιοθεσίας της τεχνολογίας αυτής [5] . ο Φυσικός έλεγχος του εξοπλισμού του Ιδιωτικού Cloud είναι πιο ασφαλής από το να βρίσκετε ο εξοπλισμός εκτός ιδιωτικού χώρου και υπό τον έλεγχο κάποιου άλλου (παρόχου).

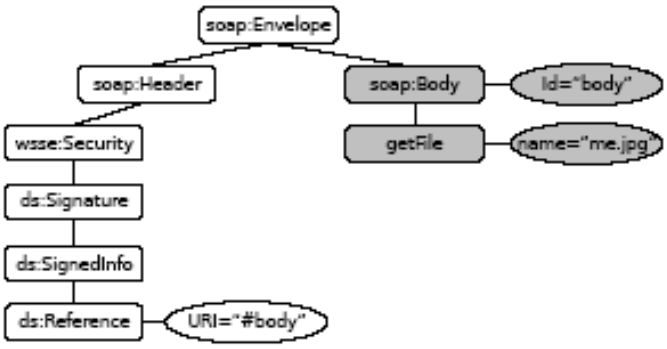
Ο Φυσικός έλεγχος και η δυνατότητα να μπορούν να επιθεωρηθούν οι σύνδεσμοι και οι θύρες πρόσβασης οπτικά, επιβάλετε, προκειμένου να εξασφαλιστεί ότι η σύνδεση των δεδομένων δεν τίθεται σε κίνδυνο. Θέματα φραγής στην υιοθέτηση της υπολογιστικής νέφους οφείλονται, σε μεγάλο βαθμό, η ανησυχίες των ιδιωτικών και δημόσιων τομέων, ποιος, και πως θα υλοποιεί την εξωτερική διαχείριση της ασφάλειας του νέφους. Αυτό προσφέρει μεγάλο κίνητρο για τους παρόχους υπηρεσιών υπολογιστικής νέφους για την κατά προτεραιότητα ανάπτυξη και διατήρηση της ισχυρής διαχείρισης των υπηρεσιών ασφάλειας. Τα θέματα ασφάλειας έχουν κατηγοριοποιηθεί στης πιο κάτω κατηγορίες, πρόσβαση ευαίσθητων δεδομένων, διαχωρισμού των δεδομένων, ιδιωτικότητας, της εκμετάλλευσης των σφαλμάτων, ανακτησιμότητας δεδομένων, διαχείριση κονσόλας ασφαλείας, έλεγχος λογαριασμού, θέματα πολύ-μίσθωσης. Οι λύσεις σε διάφορα ζητήματα ασφάλειας νέφους ποικίλλουν, από την κρυπτογραφία, ιδιαίτερα την Υποδομή Δημοσίου Κλειδιού, να χρησιμοποιείτε από πολλαπλούς παροχείς, τυποποίησης των προγραμματισμένων διεπαφών των εφαρμογών, και τη βελτίωση υποστήριξης της εικονικής μηχανής και επίσης την βελτίωση των νομικών υπηρεσιών.

1.2.1 Θέματα ασφάλειας

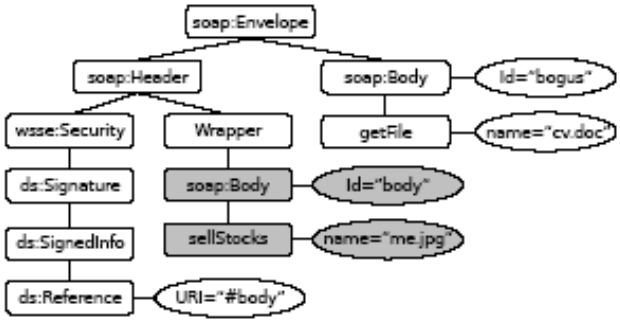
1.2.1.1 XML signature

Μία γνωστή επίθεση σε πρωτόκολλα που χρησιμοποιούν τις XML υπογραφές για αυθεντικοποίηση και προστασία της ακεραιότητας είναι η “XML signature element wrapping” . Αυτή η επίθεση εφαρμόζεται στις υπηρεσίες δικτύου, επομένως μπορεί να λάβει χώρα και στη υπολογιστική νέφους.

Στις εικόνες 1 και 2 απεικονίζετε ένα απλό παράδειγμα “wrapping attack” για να αντιληφθούμε τον τρόπο που δουλεύει η επίθεση αυτή.



Εικόνα 2. Παράδειγμα μηνύματος soap.



Εικόνα 3. Παράδειγμα μηνύματος soap μετά την επίθεση.

Στη πρώτη εικόνα παρουσιάζεται ένα μήνυμα SOAP (simple object access protocol) το οποίο στάλθηκε από ένα νόμιμο πελάτη. Το σώμα του μηνύματος περιέχει ένα αίτημα για λήψη του αρχείου “me.jpg” και είναι υπογεγραμμένο από τον αποστολέα. Η υπογραφή είναι ενσωματωμένη στη κεφαλίδα του SOAP μηνύματος και αναφέρετε στην υπογεγραμμένη δομή του μηνύματος κάνοντας χρήση ενός Xpointer στην ιδιότητα ID με την τιμή “body”. Αν ένας εισβολέας κρυφακούει ένα τέτοιο μήνυμα μπορεί να εκτελέσει την ακόλουθη επίθεση.

Το γνήσιο μήνυμα ενσωματώνετε σε ένα καινούριο μήνυμα, δίνοντας στην επίθεση το όνομά της μέσα στη κεφαλίδα του soap, και δημιουργείτε ένα καινούριο σώμα που ενσωματώνεται σε αυτό η λειτουργία που θέλει να διενεργήσει ο εισβολέας με την ταυτοποίηση του νόμιμου αποστολέα. Σε αυτό το μήνυμα, το αίτημα είναι το αρχείο “cv.doc”. Το προκύπτουσο μήνυμα περιέχει την αυθεντική υπογραφή ενός νόμιμου χρήστη, και για αυτό η υπηρεσία θα εκτελέσει το τροποποιημένο μήνυμα.

Μετά την ανακάλυψη των επιθέσεων wrapping από την McIntosh και την Austel το 2005, παρουσιάστηκε ένας αριθμός από διάφορες παραλλαγές αντιμέτρων. Ωστόσο οι επιθέσεις παράκαμψαν τα αντίμετρα με καινούριες παραλλαγές επιθέσεων. Για παράδειγμα κάποια μέθοδος με το όνομα “inline approach” [6] εισήχθη για να προστατεύσει κάποιες βασικές ιδιότητες τη δομής του

μηνύματος soap παρεμποδίζοντας έτσι τις wrapping επιθέσεις. Λίγο αργότερα, φάνηκε πώς να εκτελείτε μια επίθεση ούτως ή άλλως. [7]

Ωστόσο, λόγω της σπάνιας χρήσης της web service security στις επιχειρηματικές εφαρμογές, αυτές οι επιθέσεις παρέμειναν θεωρητικές και μη ρεαλιστικές στην καθημερινότητα. Οι επιθέσεις wrapping έγιναν γνωστές όταν το 2008 ανακαλύφθηκε ότι οι EC2 υπηρεσία του Amazon ήταν ευάλωτη στις επιθέσεις αυτές [8] . Κάνοντας χρήση μιας παραλλαγής της επίθεσης, όπως πιο πάνω, ένας εισβολέας μπορούσε να εκτελέσει αυθαίρετα EC2 λειτουργίες εκ μέρους ενός νόμιμου χρήστη.

Για να εκμεταλλευτεί το μήνυμα soap επικύρωσης ασφάλειας της ευπαθείς EC2 υπηρεσίας, έπρεπε να υποκλαπεί ένα υπογεγραμμένο μήνυμα soap ενός νόμιμου χρήστη. Δεδομένου ότι η ευπάθεια σε ένα μήνυμα soap απαιτεί επικύρωση , σου επιτρέπει να επέμβεις σε οποιουδήποτε είδους λειτουργίας και να την εκτελέσεις. Δεν έχει σημασία τι τύπου αίτηση έχει στην κατοχή του ο εισβολέας.

Η συγκεκριμενοποίηση ενός πλήθους εικονικών μηχανών για την αποστολή “spam mails” είναι ένα ακόμη παράδειγμα του τι μπορεί να κάνει ένας εισβολέας, χρησιμοποιώντας την νομιμοποιημένη ταυτότητα του χρήστη και επιβαρύνοντας έτσι τον λογαριασμό αυτού.

1.2.1.2 Ασφάλεια προγράμματος περιήγησης

Μέσα σε ένα δίκτυο νέφους, ο υπολογισμός γίνετε σε απομακρυσμένους διακομιστές . Ο υπολογιστής του πελάτη, χρησιμοποιείτε για Εισόδους-Εξόδους (πληροφοριών) και για αυθεντικοποίηση, αδειοδότηση και χρήση των εντολών προς το νέφος. Επομένως δεν έχει νόημα να αναπτύξουμε εφαρμογές για τον πελάτη, αλλά να χρησιμοποιήσουμε ένα καθολικό εργαλείο που είναι ανεξάρτητο από κάποια πλατφόρμα, για εισόδους-εξόδους πληροφορίας. Ένα τέτοιο εργαλείο είναι το πρόγραμμα περιήγησης. Αυτή η τάση έχει παρατηρηθεί κατά την διαρκεί τον τελευταίων χρόνων και έχει κατηγοριοποιηθεί σε διάφορα ονόματα: Web applications, web 2.0, Software-as-a-service(SaaS).

Οι μοντέρνοι περιηγητές, με τις τεχνικές AJAX, είναι ιδανικά κατάλληλοι για το κομμάτι του ”Input/Output”. Τι γίνεται όμως με την ασφάλεια; Μία εν μέρη απάντηση δίνει η Google [9] , όπου διαφορετικές πολιτικές ασφαλείας του περιηγητή συγκρίνονται. Εστιάζοντας στο Same Origin Policy(SOP), αυτό το έγγραφο αποκαλύπτει πολλές ελλείψεις στην ασφάλεια του περιηγητή. Αν

επιπρόσθετα χρησιμοποιήσουμε TLS (Transport layer security) που χρησιμοποιείτε για αυθεντικοποίηση και κρυπτογραφία δεδομένων, αυτές οι ελλείψεις γίνονται ακόμα και πιο εμφανείς.

Οι περιηγητές δικτύου δεν μπορούν να χρησιμοποιήσου XML υπογραφές ή XML κρυπτογράφηση κατευθείαν, τα δεδομένα μπορούν να κρυπτογραφηθούν μέσο του TLS, και οι υπογραφές χρησιμοποιούνται μέσο της χειραψίας TLS. Για όλα τα άλλα κρυπτογραφικά σύνολα δεδομένων μέσα στο web service security ο περιηγητής λειτουργεί μόνο ως μία παθητική αποθήκη δεδομένων. Κάποιες άλλες απλές λύσεις είχαν προταθεί για χρήση π.χ. κρυπτογράφηση TLS αντί για XML , αλλά είχαν περιγραφεί μεγάλα προβλήματα στην ασφάλεια μέσο αυτής της προσέγγισης στην πράξη και για να αποδειχτούν αυτά τέθηκαν σε εφαρμογή οργανωμένες επιθέσεις. Ο στόχος είναι να προτείνουμε αποδεκτές, ασφαλείς λύσεις, χρησιμοποιώντας το TLS, αλλά ταυτόχρονα να ενθαρρύνουμε την κοινότητα του περιηγητή να υιοθετήσει XML κρυπτογραφία για συμπερίληψη στον πυρήνα του περιηγητή.

1.2.1.2.1 Κληροδότημα πολιτικής ίδιας προέλευσης (SOP, same origin policy)

Με τη συμπεριβολή των γλωσσών “scripting” (σεναρίων) μέσα στις ιστοσελίδες, έγινε σημαντικός ο καθορισμός δικαιωμάτων για τα scripts. Μια φυσική επιλογή είναι να επιτρέψεις τις εργασίες γράψε-διάβασε, το περιεχόμενο τις ίδιας προέλευσης και απαγόρευσε οποιαδήποτε πρόσβαση σε περιεχόμενο από διαφορετική προέλευση. Αυτό ακριβώς κάνει το κληροδότημα πολιτικής ίδιας προέλευσης, όπου η προέλευση καθορίζεται από την ίδια την εφαρμογή η οποία μπορεί να προσδιοριστεί από ένα πλαίσιο ιστού και από μία πλειάδα πληροφοριών (domain name, port, protocol). Υπάρχουν πολλές ειδικές περιπτώσεις όπου προκύπτουν προβλήματα με την SOP αλλά αυτά θα λυθούν αν ο βασικός ορισμός των προελεύσεων είναι ορθός και αξιόπιστος. Δυστυχώς για μια διανεμημένη εφαρμογή στο WWW ο ορισμός δεν είναι εύκολο να είναι αξιόπιστος.

Το 2008 ο Dan Kaminski απέδειξε πως οι DNS μνήμες μπορούν εύκολα να “δηλητηριαστούν ” π.χ με ψευδή δεδομένα [10] . Αφού το DNS βασίζεται σε μεγάλο βαθμό στο caching , τα domain name γίνονται αναξιόπιστα. Αυτού του είδους η επίθεση μπορεί να επιδιορθωθεί μόνο από έξω από το πρωτόκολλο DNS με τη χρήση UDP πακέτων τυχαιοποίησης θυρών, για να επιτευχθεί ένα μέτριο επίπεδο αξιοπιστίας. Ένα άλλο σοβαρό πρόβλημα ασφάλειας που έχει παρατηρηθεί με το DNS είναι στη περιοχή των δρομολογητών [11] , και στο τέλος αυτός ο φορέας

επίθεσης καθιστά όλο το περιεχόμενο που φορτώθηκε από ένα URL να είναι αναξιόπιστο, εκτός αν είναι διασφαλισμένο από κάποιο άλλο μέσο.

Για εφαρμογές ιστού που έχουν ψηλές απαιτήσεις στην ασφάλεια, το TLS έχει χρησιμοποιηθεί για αρκετό χρονικό διάστημα για την προστασία των δεδομένων κατά την διάρκεια μιας μεταφοράς, και για να αυθεντικοποιήσει το domain name του διακομιστή. Τα προβλήματα με αυτή την αφελή προσέγγιση έγιναν εμφανή με την έλευση των επιθέσεων “phishing” για τις ηλεκτρονικές τραπεζικές συναλλαγές.

1.2.1.2.2 Επιθέσεις Browser based cloud authentication

Η πραγματοποίηση αυτών των θεμάτων ασφάλειας μέσα από τα βασισμένα-σε-περιηγητή πρωτόκολλα και με την υπολογιστική νέφους μπορεί να υλοποιηθεί με τη χρήση των πρωτοκόλλων FIM(federal, Identity, Management). Εφόσον ο περιηγητής από μόνος του δεν επιτρέπει την παραγωγή έγκυρων κρυπτογραφημένων token για ταυτοποίηση στο νέφος, τότε η ταυτοποίηση πρέπει να γίνετε μέσο κάποιου άλλου, έμπιστου τρίτου.

Το πρωτότυπο αυτών των πρωτοκόλλων είναι το Microsoft passport [12] το οποίο έχει σπάσει ο SLEMKO [13] . Αν δεν μπορεί να υλοποιηθεί απευθείας αυθεντικοποίηση στον διακομιστή επειδή ο περιηγητής δεν έχει τα απαραίτητα διαπιστευτήρια, μια HTTP ανακατεύθυνση στέλνετε στον passport διακομιστή αυθεντικοποίησης όπου ο χρήστης μπορεί να δώσει τα διαπιστευτήριά του (όνομα χρήστη, κωδικό). Ο διακομιστής μεταφράζει την αυθεντικοποίηση σε ένα “Kerberos token ” το οποίο αποστέλλετε στον διακομιστή που ζήτησε την αυθεντικοποίηση εξαρχής, μέσο μιας άλλης ανακατεύθυνσης HTTP. Το κυρίως πρόβλημα με τον κέρβερο είναι το γεγονός ότι δεν είναι συνδεδεμένος με τον περιηγητή και προστατεύετε μόνο από το SOP. Αν ένας εισβολέας μπορεί να έχει πρόσβαση στα tokens τότε μπορεί να έχει πρόσβαση σε όλες τις υπηρεσίες του θύματος.

Τα τρέχων, βασισμένα-σε-περιηγητή πρωτόκολλα αυθεντικοποίησης για το νέφος δεν είναι ασφαλή για δύο λόγους (α) ο περιηγητής είναι ανήμπορος να διαχειριστεί βασισμένα σε XML tokens ασφαλείας από μόνος του και (β) τα συστήματα FIM (Federated Identity Management) έχουν αποθηκεμένα token ασφαλείας μέσα στον περιηγητή όπου προστατεύονται μόνο από τον ανασφαλή SOP .

1.2.1.3 Ασφαλή αυθεντικοποίηση σε περιηγητή

Η κατάσταση ωστόσο δεν είναι απελπιστική. Αν ενσωματώσουμε το TLS και το SOP με ένα καλύτερο τρόπο, μπορούμε να κάνουμε ασφαλέστερα τα πρωτόκολλα FIM. Υπάρχουν τέσσερις τρόποι που μπορούμε να ασφαλίσουμε τα tokens με τη βοήθεια του TLS

- **TLS ομοσπονδία.** Σε αυτή την προσέγγιση τα SALM(security assertion Mark-up Language) token είναι φυλαγμένα σε ένα πιστοποιητικό πελάτη τύπου X509. Το SALM αντικαθιστά άλλα αναγνωριστικά δεδομένα όπως διακεκριμένα ονόματα. Το πιστοποιητικό έχει την ίδια περίοδο εγκυρότητας όπως ένα SALM token.

- **SALM 2.0 Holder-of-key assertion profile** [14] . Εδώ πάλι έχουμε το TLS με αυθεντικοποίηση πελάτη, μόνο που το πιστοποιητικό του πελάτη δεν μεταφέρει καθόλου πληροφορίες εξουσιοδότησης. Αντί αυτού το SALM token είναι δεσμευμένο δημόσιο κλειδί που περιέχετε στο πιστοποιητικό αυτό, συμπεριλαμβάνοντας το κλειδί αυτό σε μια “Holder-of-key assertion” [15] .

- **Καλά κλειδωμένη same origin policy**. Ενώ οι προηγούμενες προσεγγίσεις ήταν βασισμένες στο διακομιστή αυθεντικοποίησης, (σε έναν ανώνυμο τρόπο) για την ταυτοποίηση του πελάτη, σε αυτή την προσέγγιση ενδυναμώσαμε τον πελάτη ούτως ώστε να είναι σε θέση να πάρει αξιόπιστες αποφάσεις ασφαλείας. Αυτό υλοποιήθηκε χρησιμοποιώντας το δημόσιο κλειδί του διακομιστή σαν βάση για την απόφαση του SOP αντί τον ανασφαλή τρόπο του Domain Name System.

- **TLS session binding.** Ενσωματώνοντας το token σε ένα συγκεκριμένο TLS κανάλι, ο διακομιστής μπορεί να συμπεράνει ότι τα δεδομένα που αποστέλλει σε απάντηση του SALM token θα είναι προστατευμένα από το ίδιο το TLS κανάλι, και πως τα δεδομένα θα φτάσουν στον ίδιο ανώνυμο πελάτη ο οποίος προηγουμένως είχε στείλει το token.

1.2.1.4 Επιθέσεις άρνησης εξυπηρέτησης (DoS)

Οι δικτυωμένοι υπολογιστές υλοποιούν ένα συγκεκριμένο πρωτόκολλο για τη μετάδοση των δεδομένων, και αναμένουν από αυτό το πρωτόκολλο να μεταδώσει τις πληροφορίες που περιέχουν κάποια διακριτικά στοιχεία. Όταν το πρωτόκολλο υλοποιείται λανθασμένα και δεν γίνεται αρκετός έλεγχος σφαλμάτων για ανίχνευση του σφάλματος, είναι πιθανό να συμβεί μια επίθεση άρνησης παροχής υπηρεσίας. Σε ορισμένες περιπτώσεις, ο υπολογιστής που υφίσταται την επίθεση θα καταρρεύσει ή θα κρεμάσει. Σε άλλες περιπτώσεις, η υπηρεσία που υφίσταται την επίθεση θα αποτύχει χωρίς να προκαλέσει κατάρρευση του υπολογιστή. [16]

1.2.1.4.1 Επιθέσεις πλημμύρας

Οι πλημμύρες είναι απλές επιθέσεις άρνησης παροχής υπηρεσιών, που εργάζονται χρησιμοποιώντας σπάνιους πόρους, όπως είναι το εύρος ζώνης δικτύου ή την υπολογιστική ισχύ ενός νέφους.

1.2.1.4.1.1 Πλημμύρα SYN

Η πλημμύρα SYN *[17]* εκμεταλλεύεται το μηχανισμό σύνδεσης του TCP. Όταν ανοίγει μια σύνοδος TCP/IP, ο αιτών πελάτης μεταδίδει ένα μήνυμα SYN στην αιτούσα υπηρεσία του ξενιστή και ο παραλήπτης διακομιστής αποκρίνεται με ένα μήνυμα SYN-ACK που δέχεται τη σύνδεση. Ο πελάτης κατόπιν αποκρίνεται με ένα μήνυμα ACK, μετά από το οποίο η κίνηση μπορεί να αρχίσει να ρέει επάνω στην αμφίδρομη σύνδεση TCP.

Όταν ένας διακομιστής δέχεται το αρχικό μήνυμα SYN, συνήθως δημιουργεί ένα νέο νήμα διεργασίας, για να χειριστεί τις αιτήσεις του συνδεδεμένου πελάτη. Αυτή η δημιουργία νήματος διεργασίας απαιτεί χρόνο για την Κεντρική Μονάδα Επεξεργασίας του νέφους και δεσμεύει μια ποσότητα μνήμης. Πλημμυρίζοντας ένα διακομιστή με πακέτα SYN, τα οποία δεν ακολουθούνται ποτέ από ένα ACK, οι εισβολείς μπορούν να κάνουν το διακομιστή να δεσμεύσει μνήμη και χρόνο επεξεργαστή για να τα χειριστεί, και έτσι να απαγορεύει σε νόμιμους χρήστες να χρησιμοποιούν τους ίδιους πόρους. Το πρακτικό αποτέλεσμα μιας πλημμύρας SYN είναι ότι ο διακομιστής που δέχεται την επίθεση γίνεται πολύ αργός και οι νόμιμοι χρήστες δεν μπορούν να συνδεθούν.

1.2.1.4.1.2 Επίθεση χιονοστιβάδας

Η επίθεση χιονοστιβάδας, υφαρπάζει τα χαρακτηριστικά διευθυνσιοδότησης των πρωτοκόλλων επιπέδου δικτύου, π.χ., του IP και του UDP. Αυτό έχει ως αποτέλεσμα μια χιονοστιβάδα αποκρίσεων σε ερωτήματα εκπομπής, τα οποία ανακατευθύνονται σε έναν άλλο ξενιστή και όχι στον εισβολέα. Μια επίθεση χιονοστιβάδας συνεχίζεται με μια πλημμύρα στον ξενιστή του θύματος με πακέτα αιτήσεων ICMP (ping), τα οποία έχουν ως διεύθυνση απάντησης τη διεύθυνση εκπομπής του δικτύου του θύματος. Αυτό έχει ως αποτέλεσμα όλοι οι ξενιστές στο δίκτυο να απαντούν στις αιτήσεις ηχούς ICMP, και έτσι να παράγουν ακόμη περισσότερη κίνηση. Μια παραλλαγμένη επίθεση χιονοστιβάδας συνεχίζεται, όπως περιγράψαμε ήδη, αλλά με τη διεύθυνση προέλευσης IP της αίτησης ηχούς να έχει αλλαχθεί στη διεύθυνση ενός άλλου θύματος, το οποίο δέχεται όλες τις αποκρίσεις ηχούς που παράγονται από το υποδίκτυο ξενιστών στόχου. Αυτή η επίθεση είναι χρήσιμη για τους εισβολείς, επειδή μπορούν να χρησιμοποιήσουν μια σχετικά αργή σύνδεση, π.χ., ένας δρομολογητής, για να προκαλέσουν μια χιονοστιβάδα κίνησης ping προς οποιαδήποτε θέση μέσα στο Internet. Με αυτόν τον τρόπο, ένας εισβολέας με μια πιο αργή σύνδεση προς το Internet, από την σύνδεση του τελικού θύματος μπορεί να πλημμυρίσει την σύνδεση του θύματος, ρίχνοντας μια χιονοστιβάδα σε ένα δίκτυο μεγαλύτερης ταχύτητας.

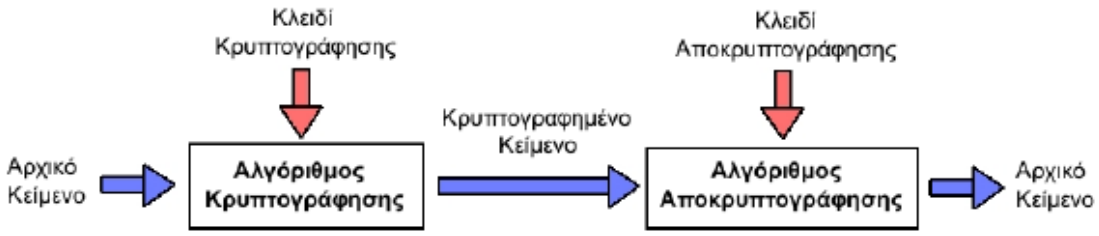
ΚΕΦΑΛΑΙΟ 2. ΚΡΥΠΤΟΓΡΑΦΙΑ, ΚΑΙ ΤΕΧΝΙΚΕΣ ΣΤΟ CLOUD

ΕΙΣΑΓΩΓΗ

Η κρυπτογραφία ασχολείται με την μελέτη της ασφαλούς επικοινωνίας, δηλαδή την μεταφορά κωδικοποιημένων μηνυμάτων αναγνωρίσιμα μόνο από το νόμιμο παραλήπτη. Η διαδικασία μετασχηματισμού ενός μηνύματος σε μια ακατανόητη μορφή με την χρήση κάποιου κρυπτογραφικού αλγορίθμου ονομάζεται κρυπτογράφηση. Οποιαδήποτε επικοινωνία μεταξύ δυο μερών προϋποθέτει κάποια θέματα ασφαλείας και επομένως η κρυπτογραφία παρέχει τους εξής αντικειμενικούς σκοπούς[18]:

- **Εμπιστευτικότητα**: η πληροφορία είναι προσβάσιμη μόνο από τον εξουσιοδοτημένο παραλήπτη.
- **Ακεραιότητα**: διαβεβαίωση ότι το παρεληφθέν μήνυμα δεν έχει αλλοιωθεί ή παραποιηθεί σε σχέση με το αρχικό.
- **Μη απάρνηση**: ο αποστολέας ή ο παραλήπτης της πληροφορίας δεν μπορεί να αρνηθεί την αυθεντικότητα της μετάδοσης ή της δημιουργίας της.
- **Πιστοποίηση**: μηχανισμός που εξακριβώνει τις ταυτότητες του αποστολέα και του παραλήπτη, καθώς και την πηγή και τον προορισμό της πληροφορίας.

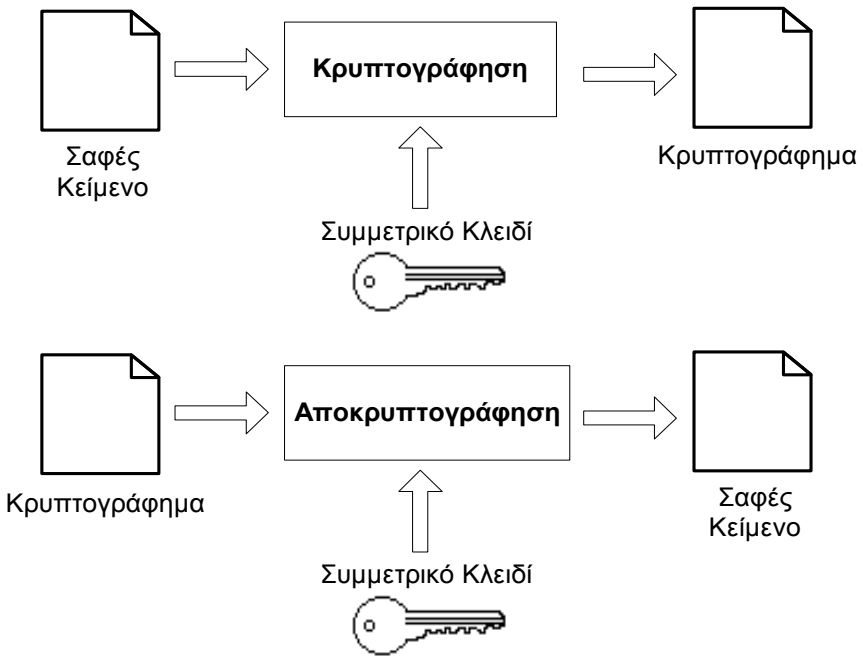
Όσον αφορά στο ψηφιακό περιεχόμενο είναι σημαντικό να διασφαλισθεί η ακεραιότητά του μετά την διανομή του και να είναι ανιχνεύσιμες οι αλλαγές που πιθανόν έχει υποστεί. Η κρυπτογραφία μπορεί να εξασφαλίσει την ακεραιότητα του μεταδιδόμενου μηνύματος, αδυνατεί όμως να συσχετίσει αρραγώς την κρυπτογραφημένη πληροφορία με το αντικείμενο. Θα πρέπει λοιπόν να εμφυτευτεί επιπρόσθετη πληροφορία στο ψηφιακό αντικείμενο που να εγγυάται την παραποίησή του ή την παράνομη διανομή του. Αυτή η ένθεση της επιπρόσθετης πληροφορίας είναι δυνατή μόνο μέσω εξειδικευμένων κρυπτογραφικών συστημάτων.[18]



Εικόνα 4: Διαδικασία κρυπτογράφησης και αποκρυπτογράφησης

2.1. ***Συμμετρική κρυπτογράφηση***

Η συμμετρική κρυπτογραφία βασίζεται στην ύπαρξη ενός μοναδικού κλειδιού, γνωστό ως μυστικό ή συμμετρικό κλειδί (secret key), με το οποίο γίνεται η κρυπτογράφηση και η αποκρυπτογράφηση της πληροφορίας. Ο αποστολέας και ο παραλήπτης είναι οι μοναδικές οντότητες που γνωρίζουν και χρησιμοποιούν το μυστικό κλειδί. Το πιο κάτω σχήμα περιγράφει την διαδικασία της συμμετρικής κρυπτογραφίας. Τα μηνύματα προς κρυπτογράφηση, γνωστά ως το σαφές κείμενο (plaintext), κρυπτογραφούνται με χρήση του συμμετρικού (ή μυστικού) κλειδιού. Η διαδικασία της κρυπτογράφησης έχει ως έξοδο ένα κείμενο σε ακατανόητη μορφή, γνωστό ως κρυπτογράφημα (ciphertext). Η ασφάλεια της μεταδιδόμενης πληροφορίας επιτυγχάνεται ακριβώς επειδή το κρυπτογράφημα μεταδίδεται σε ακατανόητη μορφή. Η διαδικασία της ανάκτησης της αρχικής πληροφορίας με τη χρήση του ίδιου συμμετρικού κλειδιού ονομάζεται αποκρυπτογράφηση.



Εικόνα 5: Διαδικασία συμμετρικής κρυπτογραφίας

Η συμμετρική κρυπτογραφία χρησιμοποιείται εδώ και χιλιάδες χρόνια. Ένας από τους παλιότερους γνωστούς κώδικες κρυπτογραφίας είναι ο αλγόριθμος του Καίσαρα, που αποτελεί έναν απλό κώδικα αντικατάστασης. Άλλοι γνωστοί και πιο σύγχρονοι αλγόριθμοι είναι οι αλγόριθμοι DES [19], IDEA [20], RC520, CAST-128 22 και AES 23.

Στα πλεονεκτήματα της συμμετρικής κρυπτογραφίας συγκαταλέγονται οι υψηλές ταχύτητες κρυπτογράφησης και αποκρυπτογράφησης που μπορούν να υπερβούν τα 100Mbps καθώς επίσης και οι μικρές απαιτήσεις της σε μνήμη και υπολογιστική ισχύ. Έτσι καθίσταται δυνατή η εφαρμογή της σε περιβάλλοντα όπως αυτά ενός κινητού τηλεφώνου ή μιας έξυπνης κάρτας. Επίσης το μέγεθος του κρυπτογραφήματος είναι αρκετά μικρότερο από αυτό του αρχικού κειμένου.

Η ανάγκη της ανταλλαγής του συμμετρικού κλειδιού μεταξύ αποστολέα και παραλήπτη είναι ένας από τους σημαντικότερους περιορισμούς της συμμετρικής κρυπτογραφίας. Η ασφάλεια της συμμετρικής κρυπτογραφίας βασίζεται αποκλειστικά στο γεγονός ότι ο αποστολέας και ο παραλήπτης μοιράζονται το συμμετρικό κλειδί πριν από την αποστολή του μηνύματος. Έτσι κρίνεται απαραίτητη η επίτευξη μιας ασφαλούς ζεύξης για την μεταφορά του συμμετρικού κλειδιού. Κάτι τέτοιο όμως δεν είναι πάντα εφικτό εξαιτίας πρακτικών αλλά και λειτουργικών δυσκολιών. Η διαδικασία της ασφαλούς ανταλλαγής του συμμετρικού κλειδιού γίνεται ακόμα μεγαλύτερη όταν οι δύο οντότητες, ο παραλήπτης και ο αποστολέας , είναι άγνωστες μεταξύ τους. Σε αυτή την περίπτωση προκύπτει η ανάγκη πιστοποίησης της ταυτότητας κάθε οντότητας έτσι ώστε να αποφευχθεί η διαβίβαση του κλειδιού σε κάποια τρίτη, μη εξουσιοδοτημένη οντότητα. Συνήθως στη συμμετρική κρυπτογραφία η μεταφορά του κλειδιού γίνεται είτε μέσω μιας φυσικής ζεύξης (ανταλλαγή κλειδιού πρόσωπο με πρόσωπο) είτε μέσω μίας έμπιστης τρίτης οντότητας, την οποία οι χρήστες εμπιστεύονται για την ασφαλή μεταφορά του κλειδιού

Ένας ακόμη σημαντικός περιορισμός αφορά στη δυσκολία κλιμάκωσης της μεθόδου. Καθώς το πλήθος των χρηστών που θέλουν να επικοινωνήσουν μεταξύ τους μεγαλώνει, γίνεται αυτονόητο ότι μεγαλώνει και το πλήθος των κλειδιών που θα χρησιμοποιηθούν για κάθε επιμέρους επικοινωνία. Για την επίτευξη επικοινωνίας μεταξύ n χρηστών απαιτούνται $n^2/2$ μοναδικά συμμετρικά κλειδιά, συμπεριλαμβανομένου και του κλειδιού που έχει κάθε χρήστης για τον εαυτό του. Τα προβλήματα της διαχείρισης των κλειδιών (key management) γίνονται ακόμα μεγαλύτερα γιατί κάθε κλειδί θα πρέπει περιοδικά να αντικαθίσταται από κάποιο

καινούριο με σκοπό τη μείωση των δεδομένων που κρυπτογραφούνται με το ίδιο κλειδί.

*2.1.1 Ο Αλγόριθμος DES (*Data Encryption Standard*)*

Ο αλγόριθμος DES είναι ο πιο ευρέως διαδεδομένος αλγόριθμος κρυπτογράφησης στον κόσμο. Για πολλά χρόνια και μεταξύ πολλών ανθρώπων η έννοια της δημιουργίας «μυστικού κώδικα» και ο DES ήταν συνώνυμα. Παρά το γεγονός ότι η Electronic Frontier Foundation[24] επένδυσε 220.000 δολάρια στο να κατασκευάσει ένα σύστημα το οποίο σπάει μηνύματα κρυπτογραφημένα σε DES, ο DES θα παραμένει το κυρίαρχο πρότυπο κρυπτογράφησης στις κυβερνητικές επικοινωνίες και διατραπεζικές συναλλαγές για αρκετά χρόνια ακόμα με μια νέα μορφή του, τον triple-DES. Πρέπει να σημειωθεί ότι πολλοί άλλοι σύγχρονοι αλγόριθμοι κρυπτογράφησης βασίζονται πάνω στον DES και κατανοώντας κάποιος τον τρόπο λειτουργίας του DES δεν θα αντιμετωπίσει προβλήματα στο να κατανοήσει και τους υπολοίπους.

Η Data Encryption Standard (DES), είναι το όνομα της Federal Information Processing Standard (FIPS) 46-3, το οποίο περιγράφει τον αλγόριθμο κρυπτογράφησης δεδομένων (DEA). Ο DEA επίσης είναι μια βελτίωση του αλγορίθμου Lucifer[25] που αναπτύχθηκε από την IBM στις αρχές του 1970. Η IBM, η Υπηρεσία Εθνικής Ασφάλειας (NSA) και το Εθνικό Γραφείο Προτύπων (NBS, σήμερα Εθνικό Ινστιτούτο Προτύπων και Τεχνολογίας NIST[26]) ήταν οι υπηρεσίες που ανέπτυξαν τον αλγόριθμο. Το DES είναι 64-bit και χρησιμοποιεί ένα 56-bit κλειδί κατά τη διάρκεια της εκτέλεσης (έχει 8 bits ισοτιμίας από το πλήρες κλειδί 64-bit). Ο DES είναι ένα συμμετρικό κρυπτογραφικό σύστημα, και συγκεκριμένα ένα 16-γύρο cipher Feistel. Όταν χρησιμοποιείται για την επικοινωνία, τόσο αποστολέας και ο παραλήπτης πρέπει να γνωρίζει το ίδιο μυστικό κλειδί, το οποίο μπορεί να χρησιμοποιηθεί για την κρυπτογράφηση και την αποκρυπτογράφηση του μηνύματος, ή για τη δημιουργία και την επαλήθευση ενός κώδικα ταυτότητας μηνυμάτων (MAC). Ο DES μπορεί επίσης να χρησιμοποιηθεί για μεμονωμένους χρήστες κρυπτογράφησης, όπως για την αποθήκευση αρχείων σε έναν σκληρό δίσκο σε κρυπτογραφημένη μορφή.

Ο DES είναι αρχέτυπος block cipher, δηλαδή, ένας πρωτότυπος κρυπαλγόριθμος συμμετρικού κλειδιού, που λαμβάνει μια σειρά από bits απλού κειμένου (plaintext bits) σταθερού μήκους και την μετατρέπει, μέσω μιας σειράς πολύπλοκων ενεργειών, σε μια άλλη σειρά bits, το κρυπτοκείμενο (chiphertext) με

το ίδιο μήκος. Στην περίπτωση του DES το μέγεθος μπλοκ (block size: Η σειρά των bits σταθερού μήκους) είναι 64 bits. Ο DES χρησιμοποιεί, επίσης, ένα κλειδί για να προσαρμόσει την μετατροπή, ώστε η αποκρυπτογράφηση να μπορεί, υποθετικά, να πραγματοποιηθεί μόνο από εκείνους που γνωρίζουν το συγκεκριμένο κλειδί που χρησιμοποιήθηκε για την κρυπτογράφηση. Το κλειδί φαινομενικά αποτελείται από 64 bits. Ωστόσο, στην πραγματικότητα μόνο 56 από αυτά χρησιμοποιήθηκαν από τον αλγόριθμο. Τα υπόλοιπα 8 bits χρησιμοποιούνται αποκλειστικά για τον έλεγχο της ισοτιμίας (parity) και στη συνέχεια απορρίπτονται (αυτά καλούνται parity bits), για αυτό και αναφέρεται συνήθως ως κλειδί μήκους 56 bits. Όπως οι άλλοι block αλγόριθμοι κρυπτογράφησης, έτσι και ο DES από μόνος του δεν είναι ασφαλής τρόπος κρυπτογράφησης. Αντίθετα πρέπει να χρησιμοποιηθεί με ειδικό τρόπο λειτουργίας.

2.1.2 Ο αλγόριθμος Triple-DES

Λόγω της ραγδαίας ανάπτυξης του πεδίου των νέων τεχνολογιών και της πληροφορικής, έγινε απαραίτητο να βρεθεί ένας πιο αξιόπιστος αλγόριθμος κρυπτογράφησης από τον DES. Έτσι λοιπόν προέκυψε ο αλγόριθμος triple-DES. Ο triple-DES είναι απλά ο DES στον οποίο χρησιμοποιούνται δύο κλειδιά μήκους 56 bits το καθένα. Δοθέντος ενός αρχικού μηνύματος προς κρυπτογράφηση, το πρώτο κλειδί χρησιμοποιείται από τον DES για την κρυπτογράφηση του μηνύματος. Το δεύτερο κλειδί χρησιμοποιείται για να αποκρυπτογραφήσει το κρυπτογραφημένο με το πρώτο κλειδί μήνυμα. Επειδή όμως το δεύτερο κλειδί δεν είναι το σωστό κλειδί για την αποκρυπτογράφηση του μηνύματος, το μόνο που επιτυγχάνεται με αυτή τη διαδικασία είναι να μπερδεύεται ακόμα περισσότερο το ήδη κρυπτογραφημένο μήνυμα. Τελικά το μήνυμα ξανά κρυπτογραφείται με το πρώτο κλειδί και έτσι προκύπτει το τελικό κρυπτογραφημένο μήνυμα.
Αυτή λοιπόν η διαδικασία τριών βημάτων αποκαλείται triple-DES.

Ο triple-DES είναι απλά η εφαρμογή του DES τρεις φορές με δύο κλειδιά τα οποία χρησιμοποιούνται με συγκεκριμένη σειρά. Ο triple-DES μπορεί να εφαρμοστεί και με τρία διαφορετικά κλειδιά αντί για δύο. Σε αυτή την περίπτωση, το συνολικό εύρος του κλειδιού είναι 2^112.

2.1.3 Ο αλγόριθμος AES (Advanced Encryption Standard)

Ο αλγόριθμος επιλογής του NIST για κρυπτογράφηση μυστικού κλειδιού ήταν ο DES. Παρόλα αυτά, λόγω της παλαιότητας του αλλά και της αλματώδους

ανάπτυξης της υπολογιστικής ισχύος, έγινε εμφανές ότι ο χώρος είχε ανάγκη από ένα καινούριο, καλύτερο αλγόριθμο. Τον Ιανουάριο του 1997 το NIST ανακοίνωσε ότι αναζητά έναν νέο αλγόριθμο κρυπτογράφησης για τον σχηματισμό ενός νέου προηγμένου προτύπου (AES - Advanced Encryption Standard). Τον Σεπτέμβριο του ίδιου έτους, ανακοινώθηκε η επίσημη διαδικασία πρότασης αλγορίθμων.

Η διαδικασία επιλογής χωρίστηκε σε αρκετούς γύρους με ένα δημόσιο συνέδριο (workshop) στο τέλος κάθε γύρου. Ο πρώτος γύρος ολοκληρώθηκε τον Αύγουστο του 1998, οπότε και επιλέχθηκαν 15 αλγόριθμοι ως υποψήφιοι για το νέο πρότυπο. Η αξιολόγηση των αλγορίθμων στους γύρους που ακολούθησαν αφορούσε τα χαρακτηριστικά τους ως προς την ασφάλεια, το κόστος υλοποίησης και την ταχύτητα (σε s/w και h/w).

Τον Απρίλιο του 1999 (δεύτερος γύρος) ανακοινώθηκε ότι από τους δεκαπέντε υποψήφιους αλγορίθμους επιλέχθηκαν πεντε. Οι αλγόριθμοι αυτοί ήταν : MARS [27](Multiplication, Addition, Rotation and Substitution) από την IBM, Rijndael[28] (από τα αρχικά των δημιουργών του) από τους Βέλγους Joan Daemen και Vincent Rijmen, Serpent[29] από μια ομάδα Βρετανών, Ισραηλινών και Νορβηγών, RC6 [30] από τον Ronald Rivest και TwoFish[31] από τον Bruce Schneier.

Η ανάλυση των αλγορίθμων οδήγησε σε λεπτομερή καταγραφή των ιδιοτήτων καθενός. Έτσι, στο τρίτο workshop για τον AES όπου παρουσιάστηκαν τα αποτελέσματα (Απρίλιος 2000), φάνηκε ότι ο πλέον δημοφιλής αλγόριθμος ήταν ο Rijndael. Η απόφαση αυτή επισημοποιήθηκε τον Οκτώβριο του 2000 οπότε και το NIST ανακοίνωσε ότι αυτός θα ήταν ο αλγόριθμος για το πρότυπο AES.

Τον Φεβρουάριο του 2001, το NIST εξέδωσε τις πρόχειρες προδιαγραφές του AES. Ο AES χρησιμοποιεί ένα υποσύνολο των δυνατοτήτων του Rijndael αλγορίθμου και στις επίσημες προδιαγραφές του NIST μια ελαφρώς διαφορετική ονοματολογία έχει επιλεγεί. Το τελικό κείμενο με τις προδιαγραφές του προτύπου εκδόθηκε προς το τέλος του 2001, ως FIPS-PUB-197[32] (Federal Information Processing Standard Publication).

Το πρότυπο κρυπτογράφησης AES (Advanced Encryption Standard) περιγράφει μια διαδικασία κρυπτογράφησης ηλεκτρονικής πληροφορίας βασισμένη στην λογική της κωδικοποίησης ομάδων δεδομένων με κάποιο μυστικό κλειδί. Έχει προτυποποιηθεί από το NIST (National Institute of Technology) τον Νοέμβριο του 2001, αντικαθιστώντας το πρότυπο DES (Data Encryption Standard) και πλέον αποτελεί τον προτεινόμενο αλγόριθμο για εφαρμογές κρυπτογράφησης.

Η κρυπτογραφία μυστικού κλειδιού (secret key cryptography) βασίζεται στην χρήση μιας μυστικής πληροφορίας που ονομάζεται κλειδί για την κρυπτογράφηση και αποκρυπτογράφηση δεδομένων. Αν κάποιος θέλει να αποστείλει σε ένα τρίτο ένα κωδικοποιημένο μήνυμα, τότε χρησιμοποιεί το μυστικό κλειδί για να κρυπτογραφήσει την πληροφορία (στην ορολογία της κρυπτογραφίας η μη κρυπτογραφημένη πληροφορία αναφέρεται ως plaintext). Το αποτέλεσμα της κρυπτογράφησης είναι να προκύψει μια κωδικοποιημένη πληροφορία (που ονομάζεται ciphertext) η οποία αποστέλλεται. Για να μπορέσει κάποιος να ανακτήσει την αρχική πληροφορία, θα πρέπει να γνωρίζει το ίδιο μυστικό κλειδί που χρησιμοποιήθηκε για την κωδικοποίηση της. Επειδή το ίδιο ακριβώς κλειδί χρησιμοποιείται τόσο για την κρυπτογράφηση όσο και αποκρυπτογράφηση των δεδομένων, η διαδικασία κρυπτογράφησης μυστικού κλειδιού ονομάζεται και συμμετρική κρυπτογραφία (symmetric cipher).

Οι τρόποι κρυπτογράφησης μυστικού κλειδιού τυπικά χωρίζονται σε 2 κατηγορίες. Η πρώτη περιλαμβάνει διαδικασίες κρυπτογράφησης που εφαρμόζονται πάνω σε ένα μοναδικό bit (ή byte ή word) και υλοποιούν κάποιο μηχανισμό ανατροφοδότησης έτσι ώστε το κλειδί να αλλάζει συνεχώς. Για αυτό και ονομάζονται κρυπτογράφοι ροής (stream ciphers). Η δεύτερη κατηγορία (κρυπτογράφοι μπλοκ - block ciphers) αποτελείται από αλγορίθμους κρυπτογράφησης που λειτουργούν πάνω σε ομάδες δεδομένων κάθε χρονική στιγμή χρησιμοποιώντας το ίδιο κλειδί για κάθε ομάδα. Έτσι στην γενική περίπτωση, όταν το ίδιο μυστικό κλειδί χρησιμοποιείται, η ίδια ομάδα δεδομένων ενός plaintext θα κρυπτογραφηθεί στο ίδιο ciphertext όταν η κρυπτογράφηση γίνεται με έναν αλγόριθμο κρυπτογράφησης μπλοκ αλλά σε διαφορετικό ciphertext όταν χρησιμοποιηθεί ένας κρυπτογράφος ροής.

Για τους κρυπτογράφους μπλοκ, έχουν επινοηθεί αρκετοί τρόποι λειτουργίας (modes) ώστε να βελτιωθούν κάποια χαρακτηριστικά τους όπως η ασφάλεια που προσφέρουν ή να γίνουν πιο κατάλληλοι για διάφορες εφαρμογές. Τέσσερις είναι οι κυριότεροι τρόποι λειτουργίας[33] :

Electronic Codebook (ECB)

Αυτός ο τρόπος λειτουργίας είναι ο απλούστερος και ο πλέον προφανής. Το μυστικό κλειδί χρησιμοποιείται για την κρυπτογράφηση κάθε μπλοκ δεδομένων του plaintext. Κατά συνέπεια με την χρήση του ίδιου κλειδιού, το ίδιο plaintext μπλοκ θα μετατρέπεται πάντα στο ίδιο ciphertext μπλοκ. Είναι ο πλέον κοινός τρόπος

λειτουργίας των κρυπτογράφων μπλοκ γιατί είναι ο απλούστερος και άρα ο πιο εύκολα υλοποιήσιμος και συνάμα ο πιο γρήγορος καθώς δεν χρησιμοποιείται κάποιου είδους ανατροφοδότηση. Μειονέκτημα του είναι ότι είναι ο πιο ευάλωτος τρόπος κρυπτογράφησης σε επιθέσεις τύπου brute-force.

Cipher Block Chaining (CBC)

Χρησιμοποιώντας την CBC λειτουργία, προστίθεται σε έναν κρυπτογράφο μπλοκ ένας μηχανισμός ανατροφοδότησης. Ο τρόπος αυτός λειτουργίας ορίζει ότι προτού να γίνει η κρυπτογράφηση ενός νέου μπλοκ plaintext, γίνεται XOR (αποκλειστικό-Ή) του μπλοκ αυτού και του ciphertext μπλοκ που μόλις πριν έχει παραχθεί. Με τον τρόπο αυτό, δύο ταυτόσημα μπλοκ plaintext δεν κρυπτογραφούνται ποτέ στο ίδιο ciphertext. Σε σχέση με τον ECB προσφέρεται μεγαλύτερη ασφάλεια, με κόστος όμως κυρίως στην ταχύτητα κρυπτογράφησης καθώς για να ξεκινήσει η επεξεργασία ενός μπλοκ plaintext είναι απαραίτητο να έχει ολοκληρωθεί πλήρως η κρυπτογράφηση του προηγούμενου μπλοκ. Αποτρέπεται έτσι η χρήση τεχνικών pipelining (software ή hardware) που μπορούν να επιταχύνουν την διαδικασία.

Cipher Feedback (CFB)

Ο τρόπος αυτός λειτουργίας επιτρέπει σε έναν κρυπτογράφο μπλοκ να συμπεριφερθεί σαν ένας κρυπτογράφος ροής. Αυτό είναι θεμιτό όταν πρέπει να κρυπτογραφούνται δεδομένα που μπορεί να έχουν μέγεθος μικρότερο από ένα μπλοκ. Παράδειγμα τέτοιας εφαρμογής μπορεί να είναι η διαδικασία κρυπτογράφησης ενός terminal session. Περιληπτικά, κατά την CFB λειτουργία χρησιμοποιείται ένας shift καταχωρητής στο μέγεθος του block μέσα στον οποίο τοποθετούνται τα δεδομένα προς κρυπτογράφηση. Όλος ο καταχωρητής κρυπτογραφείται και αυτό που προκύπτει είναι το ciphertext. Η ποσότητα των δεδομένων που μπαίνουν μέσα στον shift καταχωρητή καθορίζεται από την εφαρμογή.

Output Feedback (OFB)

Στόχος και αυτού του τρόπου λειτουργίας των μπλοκ κρυπτογράφων είναι να εξασφαλίσει ότι το ίδιο plaintext μπλοκ δεν μπορεί να παράγει το ίδιο ciphertext μπλοκ. Σε σχέση με το CBC, χρησιμοποιείται και εδώ ένας μηχανισμός

ανατροφοδότησης παρόλα αυτά είναι εσωτερικός και ανεξάρτητος από τα plaintext και ciphertext δεδομένα.

Το πρότυπο υποστηρίζει την χρήση κλειδιών μήκους 128, 192 και 256 bits. Ανάλογα με το ποιο μήκος κλειδιού χρησιμοποιείται, συνήθως χρησιμοποιείται η συντόμευση AES-128, AES-192 και AES-256 αντίστοιχα. Ανεξάρτητα από το μήκος κλειδιού, ο αλγόριθμος επενεργεί πάνω σε μπλοκ δεδομένων μήκους 128 bits. Η διαδικασία κρυπτογράφησης είναι επαναληπτική. Αυτό σημαίνει ότι σε κάθε μπλοκ δεδομένων γίνεται μια επεξεργασία η οποία επαναλαμβάνεται πολλές φορές ανάλογα με το μήκος κλειδιού και τις επαναλήψεις που θα προκύψουν. Κάθε επανάληψη ονομάζεται γύρος (round). Στον πρώτο γύρο επεξεργασίας ως είσοδος είναι ένα plaintext μπλοκ και το αρχικό κλειδί, ενώ στους γύρους που ακολουθούν ως είσοδος είναι το μπλοκ που έχει προκύψει από τον προηγούμενο γύρο καθώς και ένα κλειδί που έχει παραχθεί από το αρχικό με βάση κάποια διαδικασία που ορίζει ο αλγόριθμος. Το τελικό προϊόν της επεξεργασίας είναι το κρυπτογραφημένο μπλοκ (ciphertext). Το μπλοκ αυτό πρέπει να σημειωθεί ότι έχει ακριβώς το ίδιο μέγεθος (128 bits) με το plaintext μπλοκ.

2.2 Ασύμμετρη Κρυπτογραφία ή Κρυπτογραφία Δημοσίου Κλειδιού

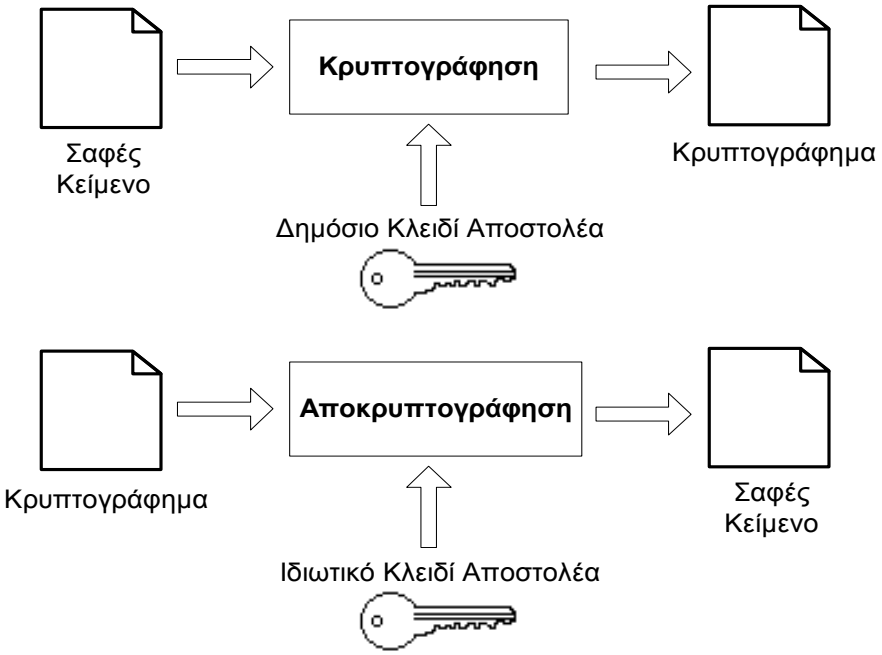
Στα μέσα της δεκαετίας του ‘70 οι Whitfield Diffie και Martin Hellman [60] πρότειναν μια νέα τεχνική για τον περιορισμό των προβλημάτων της συμμετρικής κρυπτογραφίας. Η τεχνική αυτή, γνωστή ως *κρυπτογραφία δημοσίου κλειδιού* ή *ασύμμετρη κρυπτογραφία*, βασίζεται στην ύπαρξη ενός *ζεύγους κλειδιών (key pair)*. Σε αυτό το ζεύγος, τα κλειδιά, αν και σχετίζονται μεταξύ τους με κάποια μαθηματική σχέση, είναι επαρκώς διαφορετικά έτσι ώστε η γνώση του ενός να μην επιτρέπει την παραγωγή ή τον υπολογισμό του άλλου. Αυτό σημαίνει ότι το ένα από τα κλειδιά μπορεί να είναι δημόσια γνωστό και διαθέσιμο. Το κλειδί αυτό ονομάζεται *δημόσιο κλειδί (public key)* και χρησιμοποιείται για την κρυπτογράφηση των δεδομένων. Το δεύτερο κλειδί είναι απαραίτητο να μένει κρυφό, για αυτό και ονομάζεται *ιδιωτικό κλειδί (private key)* και χρησιμοποιείται για την αποκρυπτογράφηση των δεδομένων.

Τα βασικά χαρακτηριστικά του δημόσιου και του ιδιωτικού κλειδιού είναι:

- Κάθε κλειδί είναι ένα δυαδικό αλφαριθμητικό.
- Τα κλειδιά, δημόσια και ιδιωτικά, παράγονται ταυτόχρονα από ειδικό πρόγραμμα λογισμικού.

- Τα κλειδιά δεν είναι ταυτόσημα, αλλά σχετίζονται μοναδικά έτσι ώστε να είναι δυνατή η χρήση τους για κρυπτογράφηση και αποκρυπτογράφηση δεδομένων. Η διαδικασία μέσω της οποίας παράγεται το ζεύγος των κλειδιών εξασφαλίζει ότι κάθε κλειδί σχετίζεται μοναδικά με το ταίρι του και κανένα κλειδί δεν μπορεί να παραχθεί από το άλλο.
- Τα κλειδιά, δημόσια και ιδιωτικά, που ανήκουν σε ένα ζεύγος είναι συμπληρωματικά, δηλαδή οι πληροφορίες που κρυπτογραφούνται με το ένα κλειδί μπορούν να αποκρυπτογραφηθούν μόνο με το άλλο και αντίστροφα. Με άλλα λόγια, ένα μήνυμα που έχει κρυπτογραφηθεί χρησιμοποιώντας ένα δημόσιο κλειδί μπορεί να αποκρυπτογραφηθεί μόνο χρησιμοποιώντας το αντίστοιχο ιδιωτικό κλειδί.
- Κάθε οντότητα που συμμετέχει σε ένα σύστημα επικοινωνίας δημοσίου κλειδιού έχει το δικό της ζεύγος δημόσιου και ιδιωτικού κλειδιού.
- Το ιδιωτικό κλειδί:
  Προστατεύεται από τον ιδιοκτήτη του.
  Χρησιμοποιείται για την ψηφιακή υπογραφή μηνυμάτων.
- Το δημόσιο κλειδί:
  Διανέμεται ελεύθερα και είναι προσβάσιμο σε οποιονδήποτε
  Χρησιμοποιείται για την πιστοποίηση ψηφιακών υπογραφών
  Χρησιμοποιείται για την κρυπτογράφηση μηνυμάτων
  Αποθηκεύεται μέσα σε «ψηφιακά πιστοποιητικά» που παρέχουν την ακεραιότητα και την αυθεντικότητα του ιδιοκτήτη του κλειδιού
- Παρόλο που τα δημόσια κλειδιά μπορούν να διανέμονται ελεύθερα, τα ιδιωτικά κλειδιά δε θα πρέπει ποτέ να γίνονται γνωστά σε μη εξουσιοδοτημένες οντότητες.

Η εικόνα 6 περιγράφει τη διαδικασία κρυπτογράφησης δημοσίου κλειδιού. Ο αποστολέας κρυπτογραφεί με το δημόσιο κλειδί του παραλήπτη, το οποίο είναι ελεύθερα διαθέσιμο, το μήνυμα που θέλει να του στείλει. Το κρυπτογραφημένο μήνυμα φτάνει στον παραλήπτη ο οποίος το αποκρυπτογραφεί με το ιδιωτικό κλειδί του.



Εικόνα 6: Διαδικασία Κρυπτογράφησης Δημοσίου Κλειδιού

Η κρυπτογραφία δημοσίου κλειδιού χρησιμοποιείται για την κρυπτογράφηση και αποκρυπτογράφηση δεδομένων καθώς και για την ψηφιακή υπογραφή τους. Η ασφάλεια της κρυπτογραφίας δημοσίου κλειδιού βασίζεται ακριβώς στο γεγονός ότι είναι υπολογιστικά αδύνατη η παραγωγή του ιδιωτικού από το δημόσιο κλειδί. Θεωρητικά βέβαια, το ιδιωτικό κλειδί μπορεί πάντα να υπολογιστεί αλλά το κόστος σε χρόνο, μνήμη και υπολογιστική ισχύ για κάτι τέτοιο είναι τόσο μεγάλο που καθίσταται πρακτικά αδύνατο.

Το σημαντικότερο πλεονέκτημα της ασύμμετρης κρυπτογραφίας είναι ότι δεν απαιτείται ανταλλαγή μυστικού κλειδιού. Το δημόσιο κλειδί είναι ελεύθερα διαθέσιμο, πράγμα που κάνει τη διαχείριση των κλειδιών (key management) πολύ ευκολότερη, ενώ το ιδιωτικό κλειδί είναι γνωστό μόνο στον ιδιοκτήτη του, καθιστώντας έτσι δυσκολότερη την παραποίησή του. Επίσης με την κρυπτογραφία δημοσίου κλειδιού καθίσταται δυνατή η υλοποίηση μιας πολύ σημαντικής κρυπτογραφικής λειτουργίας, αυτή της ψηφιακής υπογραφής δεδομένων.

Η κρυπτογραφία δημοσίου κλειδιού έχει μεγάλες απαιτήσεις σε υπολογιστική ισχύ, σχεδόν 100 φορές παραπάνω από αυτή που απαιτείται στην συμμετρική κρυπτογραφία. Επίσης, όπως έχει ήδη αναφερθεί, είναι αρκετά αργή ειδικά όταν πρόκειται για μεγάλα μηνύματα.

2.2.1 Αλγόριθμος Rivest Shamir Adleman (RSA)

Το 1978 οι Ron Rivest, Adi Shamir και Len Adleman πρότειναν έναν αλγόριθμο, γνωστό ως RSA [34], ο οποίος είναι ένας από τους πιο διαδεδομένους και περισσότερο χρησιμοποιημένους αλγόριθμους στην κρυπτογραφία δημοσίου κλειδιού. Αυτός ο αλγόριθμος είναι κατάλληλος για κρυπτογράφηση/αποκρυπτογράφηση δεδομένων, για την δημιουργία ψηφιακών υπογραφών και την επαλήθευσή τους καθώς και για την ασφαλή μεταφορά κλειδιών. Χρησιμοποιείται ως βάση για τη δημιουργία μιας ασφαλούς γεννήτριας ψευδοτυχαίων αριθμών καθώς και για την ασφάλεια σε ορισμένα ηλεκτρονικά παιχνίδια. Ο RSA βασίζεται στις αρχές της θεωρίας αριθμών. Στη συνέχεια αναφέρονται συνοπτικά τα βήματα που ακολουθούνται για την υλοποίηση του αλγορίθμου:

1. Επιλέγονται δύο μεγάλοι πρώτοι αριθμοί, p και q (συνήθως πολύ μεγαλύτεροι από 10^{100})
2. Υπολογίζεται $n = p \times q$ και $z = (p-1) \times (q-1)$. Ο *n* ονομάζεται *υπόλοιπο RSA (RSA modulus)*
3. Επιλέγεται ένας πρώτος αριθμός ως προς τον z ο οποίος ονομάζεται e
4. Υπολογίζεται ο d έτσι ώστε $d \times e = 1 \bmod z$
5. Το δημόσιο κλειδί αποτελείται από το ζευγάρι (e, n) και το ιδιωτικό κλειδί από το ζευγάρι (d, n).

Έχοντας υπολογίσει εκ των προτέρων τις παραπάνω παραμέτρους ξεκινά η κρυπτογράφηση. Τα βήματα που ακολουθούνται για την κρυπτογράφηση ενός κειμένου m περιγράφονται παρακάτω:

1. Το κείμενο το οποίο θα κρυπτογραφηθεί (που θεωρείται ως συρμός bit) διαιρείται σε μπλοκ, έτσι ώστε κάθε μήνυμα κειμένου, m, να πέφτει στο διάστημα $0 \le m < n$. Αυτό μπορεί να γίνει με ομαδοποίηση του κειμένου σε μπλοκ των *k* bit, όπου το *k* είναι ο μεγαλύτερος ακέραιος για τον οποίο η σχέση $2^k < n$ είναι αληθής.
2. Υπολογίζεται το $c = m^e \pmod n$ όπου το *c* είναι το κρυπτογραφημένο κείμενο.

Η ασφάλεια της μεθόδου οφείλεται στην δυσκολία της παραγοντοποίησης μεγάλων αριθμών. Εάν ο κρυπτοαναλυτής μπορούσε να παραγοντοποιήσει το δημόσια γνωστό *n*, θα μπορούσε στη συνέχεια να βρει τα *p* και *q* και από αυτά το *z*. Αν διαθέτει τα *z* και *e* μπορεί να βρει το *d* με τη βοήθεια του αλγορίθμου του Ευκλείδη. Σύμφωνα τον Rivest και τους συναδέλφους του, η παραγοντοποίηση ενός αριθμού με 200 ψηφία απαιτεί 4 δισεκατομμύρια χρόνια υπολογιστικού

χρόνου θεωρώντας ότι γίνεται χρήση ενός υπολογιστή με χρόνο εντολής 1 μsec. Ακόμα και αν οι υπολογιστές συνεχίσουν να γίνονται ταχύτεροι κατά μία τάξη μεγέθους ανά δεκαετία θα χρειαστούν αιώνες για να γίνει δυνατή η παραγοντοποίηση αριθμών με 500 ψηφία, αλλά και τότε θα μπορούμε απλά να επιλέγουμε μεγαλύτερα *p* και *q* . Το σημερινό επίπεδο της έρευνας πάνω στην παραγοντοποίηση των αριθμών απαιτεί τα κλειδιά που παράγονται με τον αλγόριθμο RSA να έχουν μήκος τουλάχιστον 1024 bits έτσι ώστε να παρέχεται ικανοποιητική ασφάλεια στις επικοινωνίες μέσα στα επόμενα χρόνια.

2.2.1.1 Πλεονεκτήματα του RSA

Ο RSA παρέχει μερικά πλεονεκτήματα τα οποία βοήθησαν στην υλοποίηση πιο ασφαλών και ευκολότερα διαχειρίσιμων συναλλαγών. Τα πλεονεκτήματα αυτά περιλαμβάνουν:

- *Απλοποίηση του προβλήματος της διαχείρισης κλειδιών*: στην συμμετρική κρυπτογραφία ο αριθμός των κλειδιών που απαιτείται για την επικοινωνία *n* οντοτήτων σε ένα κρυπτοσύστημα είναι ανάλογος του n^2. Στην ασύμμετρη κρυπτογραφία όμως κάθε χρήστης χρειάζεται δύο κλειδιά, έτσι ο απαιτούμενος αριθμός κλειδιών είναι απλά $2n$. Γίνεται κατανοητό λοιπόν ότι σε ένα κρυπτοσύστημα δημοσίου κλειδιού η σχέση που συνδέει τον αριθμό των χρηστών με τον αριθμό των κλειδιών είναι γραμμική και για αυτό το λόγω καθίστανται εύκολα διαχειρίσιμη ακόμα και όταν ο αριθμός των χρηστών είναι πολύ μεγάλος.
- *Ενισχυμένη ασφάλεια των συναλλαγών*: κάθε χρήστης παράγει μόνος του και για δική του χρήση ένα ζεύγος κλειδιών. Το ιδιωτικό κλειδί θα πρέπει να μένει μυστικό και κρυφό από οποιαδήποτε μη εξουσιοδοτημένη οντότητα εξαλείφοντας έτσι όχι μόνο το πρόβλημα της μεταφοράς του αλλά και την απαίτηση για την εγκατάσταση ενός ασφαλούς διαύλου επικοινωνίας. Το δημόσιο κλειδί από την άλλη είναι ευρέως διαθέσιμο και άρα μπορεί να μεταφερθεί με οποιαδήποτε βολική μέθοδο σε ένα δίκτυο χωρίς να τίθεται θέμα για την διατήρηση της μυστικότητάς του.

Ο αλγόριθμος RSA είναι κάτι παραπάνω από δεδομένο στην κρυπτογραφία δημοσίου κλειδιού σε σημείο μάλιστα που οι δύο έννοιες να θεωρούνται ταυτόσημες. Η ισχύς του RSA είναι τόσο μεγάλη που η κυβέρνηση των ΗΠΑ έχει περιορίσει σημαντικά την εξαγωγή του αλγορίθμου σε ξένες χώρες.

2.2.1.2 Πιθανές επιθέσεις στον RSA

Αν και ο αλγόριθμος RSA είναι ο επικρατέστερος στο χώρο της κρυπτογραφίας δημοσίου κλειδιού έχει κάποιες αδυναμίες. Μερικά από τα πιο σημαντικά προβλήματα που θα μπορούσε να αντιμετωπίσει ο αλγόριθμος αυτός είναι τα παρακάτω:

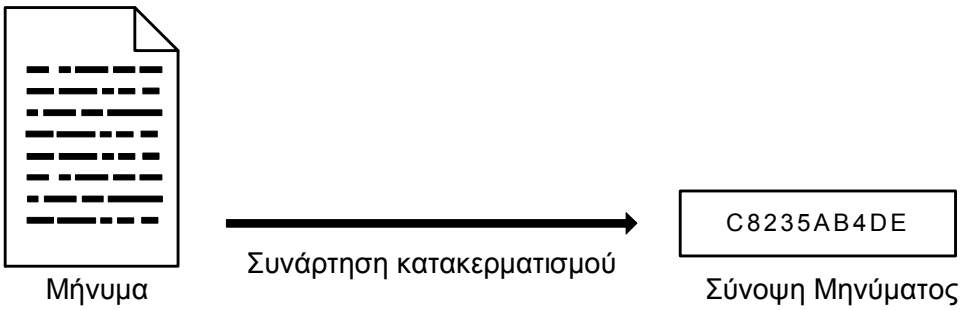
- *Παραγοντοποίηση του δημοσίου κλειδιού*: αυτό που θα ήθελε φυσικά να πετύχει κάθε εισβολέας σε ένα δίκτυο επικοινωνίας είναι να ανακτήσει το αρχικό κείμενο *m* από το αντίστοιχο κρυπτοκείμενο *c*, δοσμένου του δημοσίου κλειδιού (e,n) του παραλήπτη. Το παραπάνω είναι γνωστό και ως *πρόβλημα RSA (RSA problem, RSAP)*. Μια πιθανή προσέγγιση που θα μπορούσε να υλοποιήσει ο εισβολέας για να λύσει το πρόβλημα RSA είναι αρχικά να παραγοντοποιήσει το *n* και στη συνέχεια να υπολογίσει τα *z* και *d* ακριβώς όπως έκανε και ο χρήστης του δικτύου. Από τη στιγμή που ο εισβολέας καταφέρει να υπολογίσει το d τότε μπορεί να αποκρυπτογραφήσει οποιαδήποτε πληροφορία στέλνεται σε αυτόν το χρήστη. Το παραπάνω σενάριο πρόκειται για την πιο μεγάλη απειλή που μπορεί να αντιμετωπίσει ο αλγόριθμος αυτός. Προς το παρόν ο RSA φαίνεται να είναι εξαιρετικά δυνατός και ασφαλής. Αν και, όπως αναφέρθηκε παραπάνω, η παραγοντοποίηση του υπολοίπου RSA *n* είναι πολύ δύσκολη, σε περίπτωση που κάτι τέτοιο επιτευχθεί όλα τα μηνύματα που κρυπτογραφούνται με το δημόσιο κλειδί θα μπορούν να αποκρυπτογραφηθούν.
- *Επίθεση επανάληψης (cycle attack):* σε αυτό το είδος επίθεσης το κρυπτοκείμενο αποκρυπτογραφείται επαναλαμβανόμενα μέχρι να προκύψει το αρχικό κείμενο. Ένας μεγάλος αριθμός επαναλήψεων μπορεί να καταφέρει να αποκρυπτογραφήσει οποιοδήποτε κρυπτοκείμενο. Παρόλα αυτά, η μέθοδος αυτή είναι εξαιρετικά αργή και στην περίπτωση που το μήκος του κλειδιού είναι μεγάλο, μη πρακτική.
- *Επίθεση στο υπόλοιπο RSA, n*: η επίθεση στο υπόλοιπο *n* μπορεί να εφαρμοσθεί σε περιπτώσεις όπου υπάρχει μια ομάδα επικοινωνούντων που έχουν κλειδιά των οποίων το n είναι ίδιο. Όταν γίνεται χρήση του RSA είναι απαραίτητο κάθε οντότητα να διαλέγει το δικό της υπόλοιπο RSA *n*. Ο λόγος εξηγείται στη συνέχεια. Κάποιες φορές προτείνεται μια έμπιστη κεντρική αρχή για να διαλέγει το υπόλοιπο RSA *n*. Η ίδια αρχή στη

συνέχεια διανέμει σε κάθε χρήστη του δικτύου ένα ζεύγος (e_i, d_i). Όμως, μπορεί να αποδειχθεί ότι η γνώση οποιουδήποτε ζεύγους (e_i, d_i) επιτρέπει την παραγοντοποίηση του *n.* Έτσι, οποιαδήποτε οντότητα θα μπορούσε να υπολογίσει το *d* για οποιοδήποτε άλλη οντότητα μέσα στο δίκτυο. Συνεπώς, αν ένα κρυπτογραφημένο μήνυμα στελνόταν σε μία ή παραπάνω οντότητες μέσα στο δίκτυο, υπάρχει τεχνική με την οποία ένας εισβολέας έχει μεγάλες πιθανότητες να ανακτήσει το αρχικό μήνυμα χρησιμοποιώντας μόνο μια πληροφορία που είναι δημόσια διαθέσιμη (δηλαδή το δημόσιο κλειδί (e, n))

Παρόλες τις οποιεσδήποτε αδυναμίες του, ο RSA συνεχίζει να θεωρείται ως το de facto δεδομένο για την κρυπτογράφηση δημοσίου κλειδιού, ιδιαίτερα όταν πρόκειται για δεδομένα που μεταφέρονται στο Internet.

2.3 Συναρτήσεις Κατακερματισμού (Hash Functions)

Ο όρος *συνάρτηση κατακερματισμού* (hash function) υποδηλώνει ένα μετασχηματισμό *H* ο οποίος παίρνει ως είσοδο ένα μήνυμα *m* ανεξαρτήτου μήκους και δίνει ως έξοδο μία ακολουθία χαρακτήρων h, είναι δηλαδή $h = H(m)$. Η έξοδος *h* μιας συνάρτησης κατακερματισμού ονομάζεται *τιμή κατακερματισμού (hash value)* ή *σύνοψη μηνύματος (message digest)* και έχει συγκεκριμένο μήκος ανάλογα με το είδος του αλγόριθμου κατακερματισμού που χρησιμοποιείται, συνήθως πολύ μικρότερο από αυτό του αρχικού μηνύματος (εικόνα 7). Μπορούμε να φανταστούμε την σύνοψη μηνύματος ως το “ψηφιακό αποτύπωμα” (“digital fingerprint”) του εγγράφου.



Εικόνα 7: Συνάρτηση κατακερματισμού

Οι σημαντικότερες ιδιότητες των συναρτήσεων κατακερματισμού με μορφή $y = H(x)$ είναι:

- Η είσοδος *x* μπορεί να έχει οποιοδήποτε μήκος
- Η έξοδος *y* έχει περιορισμένο μήκος
- Δεδομένου του x και της συνάρτησης Η είναι εύκολος ο υπολογισμός του *H*(*x*)
- Η *H*(*x*) είναι μονόδρομη (one way function)
- Η *H*(*x*) είναι αμφιμονοσήμαντη (συνάρτηση ένα προς ένα)

Μια μονόδρομη συνάρτηση κατακερματισμού είναι μία συνάρτηση κατακερματισμού για την οποία είναι υπολογιστικά ανέφικτο να υπολογιστεί η αντιστροφή της, δηλαδή το αρχικό μήνυμα δεν μπορεί να ανακτηθεί από τη σύνοψή του. Όταν επιπλέον η συνάρτηση είναι αμφιμονοσήμαντη, τότε είναι πολύ δύσκολο να βρεθούν δύο διαφορετικά μηνύματα με την ίδια σύνοψη. Στην περίπτωση που κάτι τέτοιο συμβεί τότε υπάρχει *σύγκρουση* (collision) .
Οι πιο γνωστοί αλγόριθμοι κατακερματισμού είναι ο SHA-1[35] με σύνοψη 160 bits, οι MD5[36] με σύνοψη 128 bit και ο RIPEMD-160 [37] με σύνοψη 160 bits. Οι νέες εκδόσεις του αλγορίθμου SHA, SHA-256, SHA-384 και SHA-512 [38] δίνουν σύνοψη μηνύματος 256, 384 και 512 bits αντίστοιχα.

2.4 Υβριδική Κρυπτογραφία Ψηφιακού Φακέλου

Ιδιαίτερο ενδιαφέρον για την επίτευξη ασφαλούς επικοινωνίας μεταξύ δύο μερών παρουσιάζει η υβριδική κρυπτογραφία που είναι γνωστή και ως *ψηφιακός φάκελος [39] (digital envelope)* και αξιοποιεί ταυτόχρονα τις τεχνικές συμμετρικής και ασύμμετρης κρυπτογραφίας. Η υβριδική αυτή κρυπτογραφία μπορεί να χρησιμοποιηθεί για πολλούς παραλήπτες ταυτόχρονα. Τα βήματα που ακολουθούνται για τη δημιουργία ενός ψηφιακού φακέλου είναι τα εξής:

1. Δημιουργείται ένα συμμετρικό κλειδί με χρήση ενός αλγορίθμου συμμετρικής κρυπτογραφίας (π.χ. του DES).
2. Η αρχική πληροφορία κρυπτογραφείται με το συμμετρικό κλειδί που έχει δημιουργηθεί.
3. Το συμμετρικό κλειδί κρυπτογραφείται με το δημόσιο κλειδί του παραλήπτη.
4. Τα δύο κρυπτογραφημένα κείμενα αποτελούν τον ψηφιακό φάκελο του παραλήπτη.

Ο παραλήπτης ανοίγει τον ψηφιακό του φάκελο αποκρυπτογραφώντας με το ιδιωτικό κλειδί του το κρυπτογραφημένο συμμετρικό κλειδί. Με χρήση του συμμετρικού κλειδιού ο παραλήπτης αποκρυπτογραφεί το αρχικό κείμενο. Μετά

την επίτευξη μιας ασφαλούς επικοινωνίας μεταξύ αποστολέα και παραλήπτη το συμμετρικό κλειδί καταστρέφεται.

Η χρήση της υβριδικής κρυπτογραφίας βοήθα στο να ξεπεραστούν κάποιες σημαντικές αδυναμίες της κρυπτογραφίας δημοσίου κλειδιού. Συγκεκριμένα η κρυπτογραφία δημοσίου κλειδιού είναι αρκετά αργή σε σύγκριση με την συμμετρική κρυπτογραφία, ειδικά όταν πρόκειται να κρυπτογραφηθούν μεγάλα μηνύματα. Ακόμα όμως και στην περίπτωση που ο όγκος των προς κρυπτογράφηση δεδομένων είναι μικρός, έχει καθιερωθεί να χρησιμοποιείται η κρυπτογραφία ψηφιακού φακέλου. Με αυτόν τον τρόπο αποφεύγεται οποιαδήποτε σύγχυση ως προς το αν το αποτέλεσμα της αποκρυπτογράφησης είναι δεδομένα ή συμμετρικό κλειδί.

2.5 Πρωτόκολλο Ασφάλειας SSL

Το SSL [40] (Secure Socket Layer) είναι ένα ευέλικτο, γενικού σκοπού σύστημα κρυπτογράφησης για την προστασία της επικοινωνίας μέσω του Παγκόσμιου Ιστού, το οποίο είναι ενσωματωμένο και στα προγράμματα πλοήγησης τις Netscape και της Microsoft.

Το πρωτόκολλο SSL έχει σχεδιαστεί για να παρέχει απαραίτητη επικοινωνία μεταξύ δυο συστημάτων, από τα οποία το ένα λειτουργεί σαν πελάτης (client) και το άλλο σαν εξυπηρετητής (server). Δηλαδή το πρωτόκολλο αυτό μπορεί να παρέχει απόρρητη επικοινωνία μεταξύ εμπόρου και πελάτη σε μια συναλλαγή πληρωμής και για το λόγω αυτό, το SSL αποτελεί το κύριο πρωτόκολλο ασφαλείας για το ηλεκτρονικό εμπόριο. Συγκεκριμένα, το πρωτόκολλο SSL παρέχει κρυπτογράφηση της μεταδιδόμενης πληροφορίας (data encryption), υποχρεωτική πιστοποίηση της ταυτότητας του εξυπηρετητή (server authentication) και προαιρετική πιστοποίηση της ταυτότητας του πελάτη (client authentication) μέσω έγκυρων πιστοποιητικών που έχουν εκδοθεί από έμπιστες Αρχές Πιστοποίησης (Certificates Authorities). Υποστηρίζει πληθώρα μηχανισμών κρυπτογράφησης και ψηφιακών υπογραφών για την αντιμετώπιση όλων των διαφορετικών αναγκών. Επιπλέον εξασφαλίζει την ακεραιότητα των δεδομένων (data integrity), εφαρμόζοντας την τεχνική των Messages Authentication Codes (MACs), ώστε κανείς να μην μπορεί να αλλοιώσει την πληροφορία χωρίς να γίνει αντιληπτός. Για κάθε κρυπτογραφημένη συναλλαγή δημιουργείται ένα κλειδί συνόδου (session key) το μήκος του οποίου μπορεί να είναι 40 bits ή 128 bits. Είναι γνωστό ότι όσο

μεγαλύτερο είναι το μήκος του κλειδιού, τόσο πιο ασφαλής είναι η κρυπτογραφημένη επικοινωνία.

Το πρωτόκολλο SSL αναπτύχθηκε από την Netscape Communications Corporation για την ασφαλή επικοινωνία ευαίσθητων πληροφοριών όπως προσωπικά στοιχεία και αριθμούς πιστωτικών καρτών. Έχουν υπάρξει τρεις εκδόσεις του SSL. Η ιστορία του SSL έχει ως εξής:

Τον Ιούλιο του 1994 κυκλοφόρησε η πρώτη έκδοση v.1.0 του πρωτοκόλλου από την Netscape, η οποία χρησιμοποιήθηκε μόνο για εσωτερικές ανάγκες της εταιρείας.

Τον Δεκέμβριο του 1994 κυκλοφόρησε η δεύτερη έκδοση v.2.0 του πρωτοκόλλου, η οποία ενσωματώθηκε στο web browser της Netscape, τον Netscape Navigator.

Τον Ιούλιο του 1995 εκδόθηκε ο αντίστοιχος web browser της Microsoft, ο internet explorer, ο οποίος υποστηρίζει και αυτός την έκδοση v.2.0 του SSL, με κάποιες όμως επεκτάσεις της Microsoft.

Το πρωτόκολλο SSL, στην έκδοση v.2.0, καθιερώθηκε ως de facto πρότυπο για κρυπτογραφική προστασία της HTTP κυκλοφορίας δεδομένων. Το HTTP(Hyper Text Transfer Protocol) είναι ένα πρωτόκολλο που φροντίζει τη μεταφορά και τον τρόπο μετάδοσης δεδομένων στο διαδίκτυο. Ωστόσο το SSL v.2.0 είχε αρκετούς περιορισμούς τόσο ως προς την κρυπτογραφική ασφάλεια όσο και ως προς την λειτουργικότητά του. Για το λόγο αυτό υπήρχε η ανάγκη για βελτίωση της έκδοσης v.2.0. Έτσι το πρωτόκολλο αναβαθμίστηκε σε SSL v.3.0 με δημόσια αναθεώρηση και σημαντική συνεισφορά από τη βιομηχανία.

Το Νοέμβριο του 1995 κυκλοφόρησε επισήμως η έκδοση v.3.0 του SSL, ενώ λίγους μήνες πιο πριν εφαρμοζόταν σε προϊόντα της εταιρίας, όπως τον Netscape Navigator.

Το Μάιο του 1996 το SSL περνά στη δικαιοδοσία του Internet Engineering Task Force-IETF, ο οποίος δημιουργεί την ειδική ομάδα εργασίας TLS group και μετονομάζει την νέα έκδοση του SSL, σε TLS(Transport Layer Security)[41].

Η ομάδα εργασίας TLS group καθιερώθηκε το 1996 για να τυποποιήσει το πρωτόκολλο Transport Layer Security. Η TLS group εργάστηκε πάνω στο SSL v.3.0 πρωτόκολλο. Η ομάδα αυτή έχει ολοκληρώσει μια σειρά από προδιαγραφές που περιγράφουν τις εκδόσεις 1.0 και 1.1 του TLS πρωτοκόλλου, και ετοιμάζει την έκδοση 1.2.

Τον Ιανουάριο του 1999 εκδίδεται η πρώτη έκδοση του πρωτοκόλλου TLS, η οποία μπορεί να θεωρείται και ως έκδοση v.3.1 του SSL.

Το Δεκέμβριο του 2005 δημοσιεύεται η έκδοση 1.1 του TLS πρωτοκόλλου από την TLS group.

Η Τρίτη έκδοση του πρωτοκόλλου SSL κάλυψε πολλές αδυναμίες της δεύτερης έκδοσης. Οι σημαντικότερες αλλαγές αφορούν:

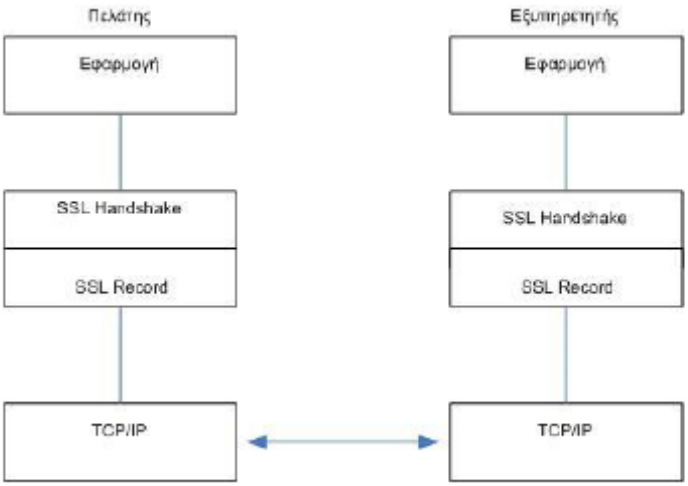
1. Τη μείωση των απαραίτητων μηνυμάτων κατά το στάδιο εγκαθίδρυσης της σύνδεσης (χειραψία-handshake),
2. Τη επιλογή των αλγορίθμων συμπίεσης και κρυπτογράφησης από τον εξυπηρετητή
3. Την εκ νέου διαπραγμάτευση του κυρίως κλειδιού (master key) και του αναγνωριστικού συνόδου (session-id).

Ακόμη αυξάνονται οι διαθέσιμοι αλγόριθμοι κρυπτογράφησης και προστίθενται νέες τεχνικές για τη διαχείριση των κλειδιών.

Γενικά, η Τρίτη έκδοση του SSL είναι πιο ολοκληρωμένη σχεδιαστικά από τη δεύτερη, με μεγαλύτερο εύρος υποστήριξης και λιγότερες ατέλειες. Επειδή η Netscape επιθυμούσε την παγκόσμια υιοθέτηση του πρωτοκόλλου SSL, γεγονός που ερχόταν σε σύγκρουση με την τότε νομοθεσία των Η.Π.Α περί εξαγωγής κρυπτογραφικών αλγόριθμων, αναγκάστηκε να επιτρέψει την χρήση αλγόριθμών κρυπτογράφησης με κλειδί των 40 bits στις προς εξαγωγή εφαρμογές SSL,τη στιγμή που η κανονική έκδοση του χρησιμοποιεί κλειδί των 128 bits.

2.5.1 Η αρχιτεκτονική του SSL

Η αρχιτεκτονική του SSL protocol απεικονίζεται στη εικόνα 8



Εικόνα 8: αρχιτεκτονική SSL

Το SSL μπορεί να λειτουργήσει πάνω από οποιοδήποτε πρωτόκολλο μεταφοράς. Δεν εξαρτάται από την ύπαρξη του TCP/IP (Transmission Control Protocol/ Internet Protocol) και υποστηρίζει πρωτόκολλα εφαρμογών όπως τα HTTP, FTP και TELNET. Το TCP/IP είναι το πρωτόκολλο επικοινωνίας (communication protocol) για την επικοινωνία ανάμεσα σε υπολογιστές που είναι συνδεδεμένοι στο διαδίκτυο. Τα αρχικά TCP/IP αναφέρονται σε δύο από τα σημαντικότερα πρωτόκολλα που χρησιμοποιούνται στο διαδίκτυο, δηλ. στο TCP και στο IP. Το FTP (File Transfer Protocol) είναι ένα πρωτόκολλο μεταφοράς αρχείων, το οποίο φροντίζει για τη διακίνηση αρχείων μέσα στο διαδίκτυο, και το TELNET είναι ουσιαστικά μια υπηρεσία του διαδικτύου με την οποία οι χρήστες αποκτούν απευθείας πρόσβαση σε άλλους υπολογιστές στο διαδίκτυο.

Είναι σημαντικό κάθε καινούργιο πρωτόκολλο επικοινωνίας να συμμορφώνεται με το μοντέλο διασύνδεσης ανοικτών συστημάτων (Open System Interconnection, OSI), έτσι ώστε να μπορεί να αντικαταστήσει εύκολα κάποιο υπάρχον πρωτόκολλο ή να ενσωματωθεί στην υπάρχουσα δομή πρωτοκόλλων. Το SSL λειτουργεί προσθετικά σε σχέση με την υπάρχουσα δομή του OSI και όχι ως πρωτόκολλο αντικατάστασης. Επιπλέον η χρήση του SSL δεν αποκλείει τη χρήση άλλου μηχανισμού ασφαλείας που λειτουργεί σε υψηλότερο επίπεδο, όπως για παράδειγμα το HTTP/S που εφαρμόζεται στο επίπεδο εφαρμογής πάνω από το SSL. Το HTTP/S(HTTP Secure) πρωτόκολλο φροντίζει για την ασφαλή μεταφορά δεδομένων στο διαδίκτυο.

Ένα σημαντικό πλεονέκτημα της ασφάλειας επιπέδου μεταφοράς γενικά και του SSL ειδικότερα είναι η ανεξαρτησία από την εφαρμογή, που σημαίνει ότι

μπορεί να χρησιμοποιηθεί για να παρέχει ασφάλεια σε οποιαδήποτε TCP/IP εφαρμογή στρωματοποιείτε στην κορυφή του.

Το πρωτόκολλο SSL παρέχει TCP/IP ασφάλεια σύνδεσης η οποία έχει τρεις βασικές ιδιότητες:

1. Οι επικοινωνούντες μπορούν να αυθεντικοποιούνται αμοιβαία χρησιμοποιώντας κρυπτογραφία δημοσίου κλειδιού.
2. Επιτυγχάνεται εμπιστευτικότητα των μεταδιδόμενων δεδομένων αφού η σύνδεση κρυπτογραφείται διαφανώς μετά από μια αρχική χειραψία και τον καθορισμό ενός κλειδιού συνόδου.
3. Προστατεύεται η ακεραιότητα των μεταδιδόμενων δεδομένων, καθώς τα μηνύματα αυθεντικοποιούνται διαφανώς και ελέγχονται ως προς την ακεραιότητα τους κατά τη μετάδοση με χρήση MACs.

Για τη γενική λειτουργία του πρωτοκόλλου SSL υπάρχουν δύο βασικές οντότητες, η σύνοδος SSL και η σύνδεση SSL.

2.5.1.1 Σύνοδος SSL

Η **σύνοδος SSL** αποτελεί τη δημιουργία μιας σχέσης μεταξύ ενός πελάτη και ενός εξυπηρετητή. Οι σύνοδοι δημιουργούνται από το SSL Handshake Protocol και είναι ομάδες παραμέτρων ασφάλειας, οι οποίες μπορούν να διαμοιραστούν ταυτόχρονα σε πολλές συνδέσεις. Ο κύριος λόγος για αυτό είναι η αποφυγή χρονοβόρων διαπραγματεύσεων νέων παραμέτρων ασφάλειας για κάθε νέα σύνδεση.

Οι παράμετροι που περιέχονται και μοιράζονται σε μια σύνοδο είναι οι ακόλουθοι:

- *Αναγνωριστικό συνόδου*: επιλέγεται από τον εξυπηρετητή για αναγνώριση μιας ενεργούς ή επαναληπτικής κατάστασης συνόδου.
- *Ψηφιακό πιστοποιητικό* (μεταξύ ομότιμων οντοτήτων).
- *Μέθοδος συμπίεσης των δεδομένων:* Αλγόριθμος που χρησιμοποιείται γιο συμπίεση δεδομένων πριν την κρυπτογράφηση.
- *Αλγόριθμος κρυπτογράφησης των δεδομένων*.
- *Κύριο μυστικό* (master secret): Μοναδικός αριθμός μήκους 48-byte, κοινό μυστικό μεταξύ εξυπηρετητή και πελάτη.
- *Δυνατότητα επανεκκίνησης της συνόδου.*

2.5.1.2 Σύνδεση SSL

Σύνδεση SSL είναι η μεταφορά των πληροφοριών μεταξύ δύο οντοτήτων. Στο SSL σι συνδέσεις αυτές είναι σχέσεις μεταξύ ομότιμων οντοτήτων και είναι παροδικές.

Οι παράμετροι που περιέχονται σε μια σύνδεση είναι σι ακόλουθοι:

Τυχαίοι αριθμοί μεταξύ πελάτη και εξυπηρετητή, σι οποίοι είναι διαφορετικοί για κάθε σύνδεση.

Μυστικό MAC εξυπηρετητή: Μυστικό που χρησιμοποιείται για MAC λειτουργίες σε δεδομένα εγγεγραμμένα από τον εξυπηρετητή.

Μυστικό MAC πελάτη: Μυστικό που χρησιμοποιείται για MAC λειτουργίες σε δεδομένα εγγεγραμμένα από τον πελάτη.

Κλειδί που χρησιμοποιείται για κρυπτογράφηση δεδομένων στον εξυπηρετητή και αποκρυπτογράφηση από τον πελάτη.

Κλειδί που χρησιμοποιείται για κρυπτογράφηση δεδομένων στον πελάτη και ο αποκρυπτογράφηση από τον εξυπηρετητή.

Διανύσματα αρχικοποίησης της σύνδεσης

Αριθμοί ακολουθίας: Κάθε μέλος (εξυπηρετητής, πελάτης) διατηρεί ξεχωριστούς αριθμούς ακολουθίας για αποστολή και λήψη μηνυμάτων σε κάθε σύνδεση.

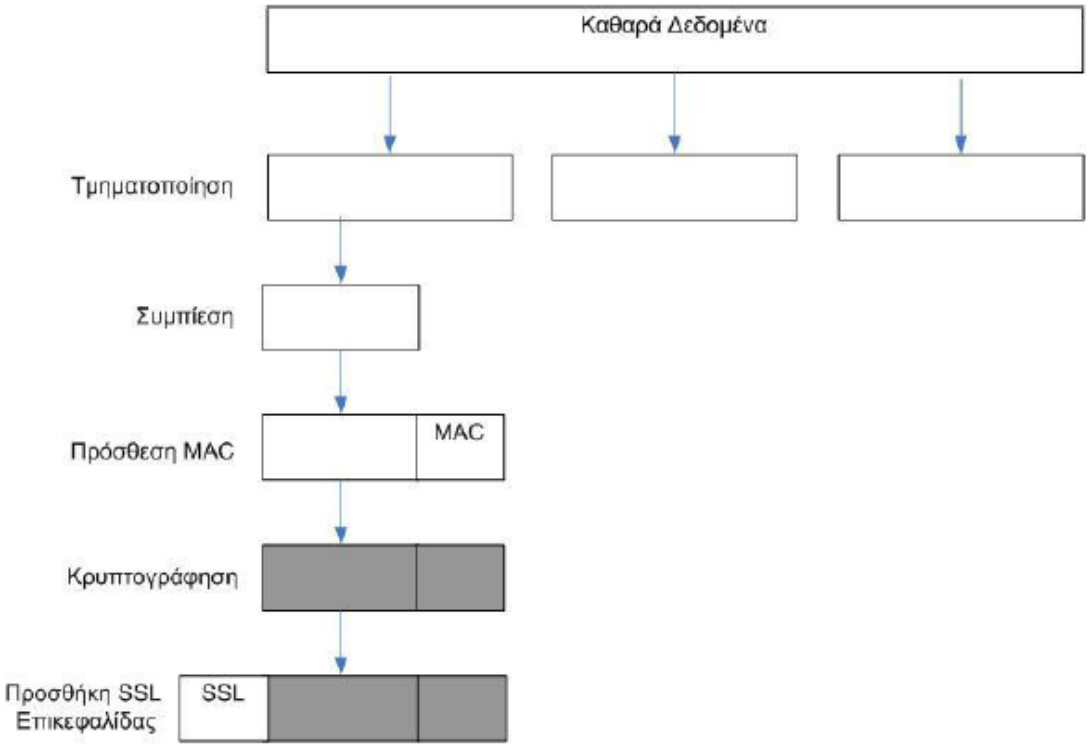
Το πρωτόκολλο SSL αποτελείται από δύο επιμέρους πρωτόκολλα, το SSL Record protocol και το SSL Handshake protocol. Το SSL record protocol παρέχει υπηρεσίες αυθεντικοποίησης εμπιστευτικότητας και ακεραιότητας δεδομένων, καθώς επίσης και προστασία από επιθέσεις με επανεκπομπή μηνυμάτων. Συγκεκριμένα το πρωτόκολλο αυτό τοποθετεί τα δεδομένα σε πακέτα και αφού τα κρυπτογραφήσει τα μεταδίδει. Επίσης εκτελεί την αντίστροφη διαδικασία γιο τα παραλαμβανόμενα πακέτα. Το SSL handshake protocol είναι ένα πρωτόκολλο αυθεντικοποίησης και ανταλλαγής κλειδιών το οποίο επίσης διαπραγματεύεται, αρχικοποιεί και συγχρονίζει τις παραμέτρους ασφάλειας. Συγκεκριμένα το πρωτόκολλο αυτό διαπραγματεύεται τους αλγόριθμους κρυπτογράφησης που θα χρησιμοποιηθούν και πραγματοποιεί την πιστοποίηση της ταυτότητας του εξυπηρετητή και του πελάτη αν αυτό ζητηθεί. Μετά την ολοκλήρωση του SSL handshake protocol, τα δεδομένα των εφαρμογών μπορούν να αποστέλλονται μέσω του SSL record protocol ακολουθώντας τις προσυμφωνημένες παραμέτρους ασφάλειας. Τα SSL record protocol και SSL handshake περιγράφονται αναλυτικά παρακάτω.

2.5.2 SSL Record Protocol

Το SSL Record Protocol παρέχει δύο υπηρεσίες για SSL συνδέσεις:

Εμπιστευτικότητα: Το Handshake Protocol ορίζει ένα Κοινό μυστικό κλειδί, το οποίο χρησιμοποιείται για την κρυπτογράφηση των δεδομένων του SSL.
Ακεραιότητα: Το Handshake Protocol επίσης ορίζει ένα κοινό μυστικό κλειδί που χρησιμοποιείται για τη δημιουργία MAC όλων των μηνυμάτων που ανταλλάσσονται.
Το SSL Record Protocol λαμβάνει δεδομένα από πρωτόκολλα υψηλότερων επιπέδων και ασχολείται με τον κατακερματισμό (fragmentation), τη συμπίεση, την αυθεντικοποίηση και την κρυπτογράφηση δεδομένων. Ουσιαστικά το πρωτόκολλο αυτό μετατρέπει τα προς μετάδοση δεδομένα σε SSL πακέτα.

Συγκεκριμένα το Record Protocol παίρνει το μήνυμα της εφαρμογής που θα μεταδοθεί, τμηματοποιεί τα δεδομένα σε εύχρηστα blocks, προαιρετικά συμπιέζει τα δεδομένα με κατάλληλους μηχανισμούς που επιλέγονται κατά τη «χειραψία» και μετά εφαρμόζει ένα MAC πάνω από τα συμπιεσμένα δεδομένα. Στη συνέχεια κρυπτογραφεί το αποτέλεσμα χρησιμοποιώντας συμμετρική κρυπτογράφηση, προσθέτει μια επικεφαλίδα SSL και στο τέλος μεταδίδει το πακέτο. Η μέθοδος συμπίεσης και ο αλγόριθμος κρυπτογράφησης καθορίζονται κατά τη διάρκεια εκτέλεσης του SSL Handshake Protocol.



Εικόνα 9: Διαδικασία SSL

Το SSL Record Protocol εκτελεί και την αντίστροφη διαδικασία για τα παραλαμβανόμενα πακέτα. Συγκεκριμένα τα δεδομένα που λαμβάνονται αποκρυπτογραφούνται, επιβεβαιώνονται, αποσυμπιέζονται, επανασυγκεντρώνονται και διανέμονται στους χρήστες των ανώτερων επιπέδων. Διάφορα πρωτόκολλα SSL μπορούν να στρωματοποιούνται στην κορυφή του SSL Record Protocol. Οι προδιαγραφές του SSL 3.0 καθορίζουν τα ακόλουθα πρωτόκολλα SSL:

Πρωτόκολλο προειδοποίησης (SSL Alert Protocol).

Πρωτόκολλο χειραψίας (SSL Handshake Protocol).

Πρωτόκολλο Αλλαγής Προδιαγραφών Κρυπτογραφίας (SSL Change Cipher Spec Protocol)

Το SSL Alert Protocol χρησιμοποιείται για να μεταφέρει προειδοποιήσεις (alerts) μέσω του SSL Record Protocol. Οι προειδοποιήσεις είναι συνήθως μηνύματα προβλημάτων και λαθών (π.χ. “λάθος MAC”, “μη αναμενόμενο μήνυμα” κλπ.) που αφορούν τόσο τη σύνδεση όσο και τη μετάδοση των μηνυμάτων μεταξύ δύο ομότιμων οντοτήτων. Με τον τρόπο αυτό ειδοποιεί το SSL να διακόψει τη σύνδεση ή να προβεί σε όποιες άλλες ενέργειες έχουν καθοριστεί.

Το πρωτόκολλο αλλαγής προδιαγραφών κρυπτογραφίας είναι το απλούστερο από τα πιο πάνω πρωτόκολλα. Χρησιμοποιείται για την αλλαγή μιας προδιαγραφής κρυπτογραφίας με μια άλλη. Κανονικά μια προδιαγραφή κρυπτογραφίας αλλάζει στο τέλος μιας SSL χειραψίας. Μπορεί όμως να τροποποιηθεί και σε οποιοδήποτε άλλη στιγμή.

2.5.3 SSL handshake protocol

Ο πελάτης και εξυπηρετητής του SSL μπαίνουν σε κατάσταση σύνδεσης χρησιμοποιώντας μία διαδικασία χειραψίας (handshaking). Κατά τη διάρκεια αυτής της διαδικασίας, ο πελάτης και ο server συμφωνούν σε διάφορες παραμέτρους που αφορούν στην ασφάλεια της σύνδεσης. Η χειραψία ξεκινάει όταν ένας πελάτης συνδέεται σε έναν server SSL και αιτείται μιας ασφαλούς σύνδεσης παρουσιάζοντας μία λίστα υποστηριζόμενων κρυπταλγορίθμων και συναρτήσεων κατακερματισμού (hash functions). Από τη λίστα αυτή, ο server επιλέγει τις

ισχυρότερες διαθέσιμες συναρτήσεις hash και κρυπταλγόριθμους, που υποστηρίζει και αυτός, ενημερώνοντας το πρόγραμμα πελάτη για την επιλογή του. Ο server στέλνει πίσω την ταυτότητά του σε μορφή ψηφιακού πιστοποιητικού. Το πιστοποιητικό συνήθως περιέχει το όνομα του εξυπηρετητή, την αρχή έκδοσης έμπιστων πιστοποιητικών και το δημόσιο κλειδί κρυπτογράφησής του. Το πρόγραμμα του πελάτη μπορεί να επικοινωνήσει με την αρχή έκδοσης του πιστοποιητικού για να διασφαλίσει τη γνησιότητά του πριν προχωρήσει στα επόμενα στάδια. Προκειμένου να δημιουργήσει τα κλειδιά συνόδου (session keys) που θα χρησιμοποιηθούν για την ασφαλή σύνδεση, ο πελάτης κρυπτογραφεί ένα τυχαίο αριθμό με το δημόσιο κλειδί του server και στέλνει το αποτέλεσμα στον server. Μόνο ο server θα πρέπει να είναι σε θέση να το αποκρυπτογραφήσει με το ιδιωτικό του κλειδί . Αυτός είναι ο ένας παράγοντας που καθιστά τα κλειδιά κρυφά από τρίτα μέρη, καθόσον μόνο ο server και ο πελάτης έχουν πρόσβαση σε αυτά τα δεδομένα. Ο πελάτης ξέρει το δημόσιο κλειδί και το τυχαίο αριθμό, ενώ ο server γνωρίζει το ιδιωτικό του κλειδί και μετά την αποκρυπτογράφηση του μηνύματος του πελάτη, το τυχαίο αριθμό. Ένα τρίτο μέρος μπορεί να μάθει το τυχαίο αριθμό μόνο αν έχει υποκλέψει ή ανακτήσει το ιδιωτικό κλειδί. Από τον τυχαίο αριθμό και τα δύο μέρη δημιουργούν τα κλειδιά για την κρυπτογράφηση και αποκρυπτογράφηση. Στο σημείο αυτό τελειώνει η διαδικασία της χειραψίας και ξεκινάει η ασφαλής σύνδεση, η οποία κρυπτογραφείται με τα κλειδιά μέχρι τη λήξη της σύνδεσης.

Το SSL ανταλλάσσει πεδία, τα οποία ενθυλακώνουν τα δεδομένα που πρέπει να μεταφερθούν. Κάθε πεδίο μπορεί να συμπιεστεί, να προστεθούν σε αυτό bit γεμίσματος, καθώς και μήνυμα κώδικα αυθεντικοποίησης (MAC) ή να κρυπτογραφηθεί, αναλόγως της κατάστασης της σύνδεσης. Κάθε πεδίο έχει ένα τμήμα τύπου περιεχομένου (content type) που προσδιορίζει το πεδίο, ένα τμήμα μήκους πεδίου και ένα τμήμα που ορίζει την έκδοση του TLS. Όταν ξεκινάει η σύνδεση, το πεδίο ενθυλακώνει ένα άλλο πρωτόκολλο μηνυμάτων χειραψίας (handshaking messaging protocol) το οποίο έχει τιμή 22 στο τμήμα τύπου περιεχομένου.

2.5.4 Αντοχή του SSL σε Γνωστές Επιθέσεις

Επίθεση Λεξικού (Dictionary Attack) *[42]*

Κατά την επίθεση αυτή, ένα τμήμα του μη κρυπτογραφημένου κειμένου βρίσκεται στην κατοχή κακόβουλων προσώπων. Το τμήμα αυτό κρυπτογραφείται

με χρήση κάθε πιθανού κλειδιού και έπειτα ερευνάται ολόκληρο το κρυπτογραφημένο μήνυμα μέχρι να βρεθεί ένα κομμάτι που να ταιριάζει με κάποιο από τα προϋπολογισμένα. Σε περίπτωση που η έρευνα έχει επιτυχία, τότε το κλειδί που χρησιμοποιήθηκε για την κρυπτογράφηση ολόκληρου του κειμένου έχει βρεθεί.

Το SSL δεν απειλείται από αυτήν την επίθεση αφού τα κλειδιά των αλγορίθμων του είναι πολύ μεγάλα (128 bits). Ακόμα και οι αλγόριθμοι σε εξαγόμενα προϊόντα, υποστηρίζουν 128 bits κλειδιά και παρόλο που τα 88 bits αυτών μεταδίδονται χωρίς κρυπτογράφηση, ο υπολογισμός 240 διαφορετικών ακολουθιών καθιστά την επίθεση εξαιρετικά δύσκολη.

Επίθεση Επανάληψης (Replay Attack)*[43]*

Όταν ένας τρίτος καταγράφει την ανταλλαγή μηνυμάτων μεταξύ πελάτη-εξυπηρετητή και προσπαθεί να χρησιμοποιήσει ξανά τα μηνύματα του πελάτη για να αποκτήσει πρόσβαση στον εξυπηρετητή, έχουμε επίθεση τύπου replay attack. Όμως το SSL κάνει χρήση του αναγνωριστικού συνόδου (Connection-ID). Το οποίο παράγεται από τον εξυπηρετητή με τυχαίο τρόπο και διαφέρει για κάθε σύνδεση. Έτσι δεν είναι δυνατόν πότε να υπάρχουν δυο ίδια αναγνωριστικά σύνδεσης.

Επίθεση Παρεμβολής (Man-In-The-Middle-Attack)*[44]*

Η επίθεση Man-In-The-Middle-Attack συμβαίνει όταν ένας τρίτος είναι σε Θέση να παρεμβάλλεται στην επικοινωνία μεταξύ του εξυπηρετητή και του πελάτη. Αφού επεξεργαστεί τα μηνύματα του πελάτη και το τροποποιήσει όπως αυτός επιθυμεί, τα προωθεί στον εξυπηρετητή. Ομοίως πράττει για τα μηνύματα που προέρχονται από τον εξυπηρετητή. Δηλαδή, προσποιείται στον πελάτη ότι είναι ο εξυπηρετητής και αντίστροφα.
Το SSL υποχρεώνει τον εξυπηρετητή να αποδεικνύει την ταυτότητα του με την χρήση έγκυρου πιστοποιητικού του οποίου η τροποποίηση είναι αδύνατη.

2.6 Transport Layer Security Protocol (TLS)

Το Μάιο του 1996 Το IETF(Internet Engineering Task Force) δημιούργησε το πρωτόκολλο Transport Layer Security WG [45] για την ασφάλεια του επιπέδου μεταφοράς. Το Δεκέμβριο του 1996 ένα Πρώτο έγγραφο TLS 1.0 κυκλοφόρησε ως Internet draft. Το έγγραφο ήταν ουσιαστικά το ίδιο με τις προδιαγραφές του SSL 3.0. Γενικά η ομάδα εργασίας για τη δημιουργία του TLS είχε σαν στρατηγική της οι

προδιαγραφές του TLS 1.0 να βασίζονται κυρίως στο SSL 3.0, παρά σε άλλα πρωτόκολλα ασφάλειας επιπέδου μεταφοράς όπως SSL 2.0. Το Δεκέμβριο του 2005, η ομάδα εργασίας TLS group δημοσίευσε την έκδοση 1.1 του TLS.

Στο TLS 1.0 ενσωματώθηκε στο SSL 3.0 με κάποιες μικρές τροποποιήσεις. Οι τροποποιήσεις αυτές αφορούσαν περισσότερο σημεία αποσαφήνισης. Η κύρια τροποποίηση που υποδείχθηκε για το SSL 3.0 ώστε να ενσωματωθεί στο TLS 1.0 είναι:

- Το TLS Record Protocol το TLS Handshake Protocol θα έπρεπε να διαχωρίζονται εντελώς και να καθορίζονται σαφώς σε σχετικά έγγραφα.Οι διαφορές μεταξύ του TLS 1.0 και του SSL 3.0 δεν είναι ιδιαίτερα σημαντικές αλλά είναι αρκετά κρίσιμες ώστε τα TLS 1.0 και SSL 3.0 να μη συνεργάζονται εύκολα. Ωστόσο το TLS 1.0 ενσωματώνει ένα μηχανισμό μέσω του οποίου μια υλοποίηση TLS μπορεί να γίνει συμβατή με Το SSL 3.0.
- Το πρωτόκολλο TLS είναι από μόνο του ένα στρωματοποιημένο πρωτόκολλο. Στο χαμηλότερο επίπεδο, το TLS record protocol λαμβάνει τα προς μετάδοση μηνύματα, κατακερματίζει τα δεδομένα σε διαχειρίσιμα τμήματα, προαιρετικά τα συμπιέζει, υπολογίζει και προσαρτά ένα MAC σε κάθε τμήμα, κρυπτογραφεί το αποτέλεσμα και το αποστέλλει. Επιπλέον το πρωτόκολλο αυτό εκτελεί την αντίστροφη διαδικασία για τα παραλαμβανόμενα πακέτα.
- Το TLS record protocol όταν λάβει ένα πακέτο το αποκρυπτογραφεί, το επιβεβαιώνει, το αποσυμπιέζει, και το επανασυναρμολογεί πριν το μεταδώσει. Μια κατάσταση TLS σύνδεσης αποτελεί το λειτουργικό περιβάλλον του TLS record protocol. Αυτή καθορίζει τους αλγόριθμους συμπίεσης κρυπτογράφησης και αυθεντικοποίησης μηνυμάτων, καθώς και τα κλειδιά που χρησιμοποιούνται για κρυπτογράφηση και αυθεντικοποίηση. Η σύνδεση αυτή (σύνδεση TLS), δημιουργείται κατά την εκτέλεση του TLS handshake protocol.

Στο υψηλότερο επίπεδο, το TLS handshake protocol χρησιμοποιείται για να συμφωνηθεί μια κατάσταση συνόδου μεταξύ του εξυπηρετητή και του εξυπηρετούμενου (πελάτη). Συγκεκριμένα προσδιορίζεται μια ταυτότητα συνόδου, μια προδιαγραφή κρυπτογραφίας, μια μέθοδος συμπίεσης και ένα κύριο κλειδί. Τα στοιχεία αυτά χρησιμοποιούνται για τη δημιουργία παραμέτρων ασφάλειας που θα

χρησιμοποιηθούν από το TLS record protocol κατά την προστασία δεδομένων εφαρμογών. Συγκεκριμένα το TLS handshake protocol αποτελείται από τρία υπό-πρωτόκολλα:

1. Το TLS change cipher spec control αποτελείται από ένα απλό μήνυμα Change_Cipher_Spec, το οποίο αποστέλλεται από τον πελάτη στον εξυπηρετητή κατά τη διάρκεια της χειραψίας αφού έχουν συμφωνηθεί ο παράμετροι ασφάλειας.
2. Το TLS alert protocol χρησιμοποιείται για να αποστέλλει μηνύματα προειδοποίησης τα οποία μεταβιβάζουν τη σημαντικότητα ενός μηνύματος προειδοποίησης και μια περιγραφή της προειδοποίησης αυτής. Οι προειδοποιήσεις είναι συνήθως μηνύματα προβλημάτων και λαθών που αφορούν κυρίως τη μετάδοση των μηνυμάτων.
3. Το TLS handshake protocol χρησιμοποιείται για να συμφωνηθεί μια κατάσταση συνόδου. Όταν ένας πελάτης και ένας εξυπηρετητής αρχίζουν για πρώτη φορά να επικοινωνούν επιλέγουν αλγόριθμους κρυπτογράφησης προαιρετικά αυθεντικοποιούνται αμοιβαία και χρησιμοποιούν κρυπτογραφία δημοσίου κλειδιού για να παράγουν ένα κύριο μυστικό και τα αντίστοιχα κλειδιά συνόδου. Η ροή των μηνυμάτων που ο πελάτης και ο εξυπηρετητής ανταλλάσσουν μεταξύ τους είναι ουσιαστικά η ίδια, όπως του SSL handshake protocol.

Μετά την εκτέλεση του TLS handshake protocol ο πελάτης και ο εξυπηρετητής μπορούν να ανταλλάσσουν μηνύματα δεδομένων εφαρμογών με ασφάλεια. Τα μηνύματα αυτά μεταφέρονται μέσω του SSL record protocol, αφού πρώτα κατακερματιστούν, συμπιεστούν, αυθεντικοποιηθούν και κρυπτογραφηθούν.

2.7 Το πρωτόκολλο SSH (secure shell)

Το Secure Shell (SSH) [46] είναι ένα δικτυακό πρωτόκολλο που επιτρέπει τη μεταφορά δεδομένων, χρησιμοποιώντας ένα ασφαλές κανάλι μεταξύ δύο δικτυακών συσκευών. Οι δύο κύριες εκδόσεις του αναφέρονται ως SSH1 και SSH2. Χρησιμοποιήθηκε αρχικά σε συστήματα Linux και Unix ως εργαλείο πρόσβασης σε λογαριασμούς του συστήματος, αντικαθιστώντας το Telnet και άλλα ανασφαλή εργαλεία πρόσβασης στη γραμμή εντολών του λειτουργικού, τα οποία στέλνουν ευαίσθητες πληροφορίες μη κρυπτογραφημένες, καθιστώντας τα πρωτόκολλα αυτά

ευάλωτα στην ανάλυση σε επίπεδο πακέτου. Η κρυπτογράφηση που εφαρμόζεται από το SSH παρέχει εμπιστευτικότητα και ακεραιότητα των δεδομένων που διακινούνται σε ένα ανασφαλές δίκτυο όπως το δίκτυο νέφους.

Το SSH χρησιμοποιεί κρυπτογραφία δημοσίου κλειδιού για να αυθεντικοποιήσει τον απομακρυσμένο υπολογιστή και να επιτρέψει σε αυτόν να αυθεντικοποιήσει τον χρήστη, αν αυτό είναι απαραίτητο. Το SSH χρησιμοποιείται συνήθως για να συνδεθεί κάποιος σε απομακρυσμένο υπολογιστή και να εκτελέσει εντολές, αλλά επίσης υποστηρίζονται λειτουργία ενθυλάκωσης (tunneling), προώθηση θυρών TCP και συνδέσεων X11. Μπορεί επίσης, να μεταφέρει αρχεία χρησιμοποιώντας τα πρωτόκολλα SFTP και SCP(secure copy). Το SSH χρησιμοποιεί το μοντέλο εξυπηρετητή – πελάτη. Η θύρα 22 του TCP έχει αντιστοιχηθεί στους εξυπηρετητές SSH.

Ένα πρόγραμμα πελάτη χρησιμοποιείται συνήθως για να εδραιωθούν συνδέσεις στο πρόγραμμα του εξυπηρετητή που δέχεται αιτήματα συνδέσεων. Και τα δύο προγράμματα αυτά υπάρχουν στα περισσότερα λειτουργικά συστήματα, όπως είναι το Mac OS X, Linux, FreeBSD, Solaris και OPENVMS. Υπάρχουν εκδόσεις προγραμμάτων εμπορικές, ελεύθερου λογισμικού και ανοικτού κώδικα, διαφορετικής πολυπλοκότητας και με διαφορετικό βαθμό ολοκλήρωσης του προτύπου. Η πρώτη έκδοση του πρωτοκόλλου(SSH-1) σχεδιάστηκε το 1995 από τον Tatu Ylönen, έναν ερευνητή του Πανεπιστημίου Τεχνολογίας του Ελσίνκι στη Φινλανδία, παροτρυνόμενος από μια επίθεση στους κωδικούς πρόσβασης (password sniffing attacks) στο πανεπιστήμιό του. Ο στόχος του SSH ήταν να αντικαταστήσει τα πρωτόκολλα rlogin, TELNET και rsh, τα οποία δεν παρείχαν ισχυρή αυθεντικοποίηση και εγγύηση της εμπιστευτικότητας. Ο Ylönen έδωσε την υλοποίησή του ως ελεύθερο λογισμικό τον Ιούλιο του 1995 και το εργαλείο αυτό γρήγορα έγινε δημοφιλές. Προς το τέλος του 1995, η βάση χρηστών αριθμούσε περί τους 20.000 χρήστες σε 50 χώρες. Στα τέλη του 1995, ο Ylönen ίδρυσε την εταιρεία SSH Communication Security για να προωθήσει και να αναπτύξει το SSH. Η αρχική έκδοση χρησιμοποίησε διάφορα κομμάτια ελεύθερου λογισμικού, αλλά οι μετέπειτα εκδόσεις εστιάστηκαν περισσότερο στο εμπορικό λογισμικό. Εκτιμάται ότι το 2000 το πρωτόκολλο είχε δύο εκατομμύρια χρήστες.

Το 1999, οι προγραμματιστές που θέλανε μια έκδοση ελεύθερου λογισμικού, μεταβήκανε στην παλιά έκδοση 1.2.12 του αρχικού προγράμματος SSH, η οποία ήταν η τελευταία έκδοση ανοικτού κώδικα. Από αυτόν τον κώδικα εξελίχθηκε το

OSSH από τον Björn Grönvall. Λίγο αργότερα, οι προγραμματιστές του OpenBSD πήραν τον κώδικα του Grönvall και έκαναν εκτεταμένη δουλειά σε αυτόν, δημιουργώντας το OpenSSH, το οποίο ενσωματώθηκε στην έκδοση 2.6 του OpenBSD. Από αυτή την έκδοση δημιουργήθηκε ένας κλάδος μεταφοράς του OpenSSH σε άλλα λειτουργικά συστήματα. Μέχρι το 2005, το OpenSSH ήταν η πιο δημοφιλής υλοποίηση του πρωτοκόλλου, που ήταν ενσωματωμένη σε πολλά λειτουργικά συστήματα. Τώρα συνεχίζει να αναπτύσσεται και υποστηρίζει τις εκδόσεις 1.x και 2.x. Το επίσημο όνομα της ομάδας εργασίας της IETF πάνω στη δεύτερη έκδοση του προτύπου, είναι Secsh. Το 2006 μια αναθεωρημένη έκδοση του SSH-2, έγινε αποδεκτή ως πρότυπο. Η έκδοση αυτή δεν ήταν συμβατή με την SSH-1, παρείχε όμως βελτιώσεις στα χαρακτηριστικά λειτουργίας και ασφάλειας.

Το πρότυπο του SSH-2 για χρήση στο Internet περιγράφεται σε μια σειρά από κείμενα RFC(Requests for Comments) που εκδόθηκαν από την IETF. Το πρωτόκολλο μπορεί να χρησιμοποιηθεί σε πολλές εφαρμογές σε διάφορες πλατφόρμες, όπως τα Microsoft Windows, Apple Mac OS και Linux. Μερικές από τις εφαρμογές μπορεί να απαιτούν λειτουργίες που είναι διαθέσιμες σε συγκεκριμένες μόνο υλοποιήσεις εξυπηρετητών ή πελατών SSH. Από τις υλοποιήσεις του SSH, μόνο η OpenSSH μέχρι σήμερα, υποστηρίζει την υλοποίηση ενός Εικονικού Ιδιωτικού Δικτύου (VPN).

Συνοπτικά το SSH μπορεί να χρησιμοποιηθεί για:

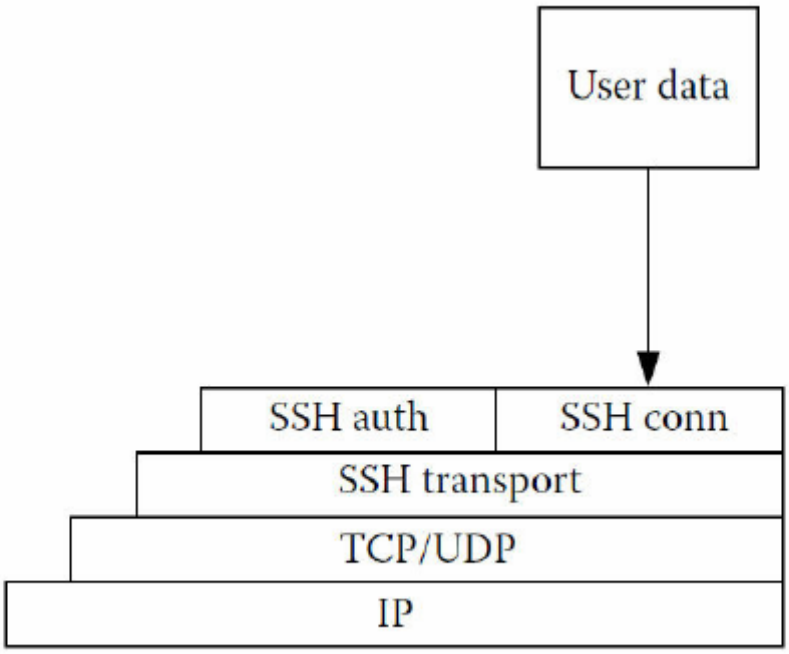
- Σύνδεση σε λογαριασμό απομακρυσμένου υπολογιστή, αντικαθιστώντας τα Telnet και rlogin
- Εκτέλεση μιας εντολής σε απομακρυσμένο ξενιστή, αντικαθιστώντας το Rsh
- Ασφαλή μεταφορά αρχείων
- Λήψη αντιγράφων ασφαλείας, αντιγραφή και αντικατοπτρισμό αρχείων (mirroring) αποτελεσματικά και με ασφάλεια, σε συνδυασμό με το rsync
- Προώθηση της κίνησης σε μια θύρα ή τη λειτουργία ενθυλάκωσης (tunneling).
- Δημιουργία Εικονικών Ιδιωτικών Δικτύων (VPN).
- Προώθηση του περιβάλλοντος X από έναν απομακρυσμένο υπολογιστή.
- Φυλλομετρητές (web browsers) διαμέσου μιας κρυπτογραφημένης σύνδεσης proxy που υποστηρίζει το πρωτόκολλο SOCKS.
- Να προσαρτηθεί με ασφάλεια ένας φάκελος σε έναν απομακρυσμένο εξυπηρετητή σαν ένα σύστημα αρχείων σε τοπικό υπολογιστή χρησιμοποιώντας το SSHFS(SSH file system).

- Αυτόματη απομακρυσμένη επίβλεψη και διαχείριση (monitoring and management) σε server με μία από τις παραπάνω τεχνικές και μηχανισμούς.

Πέρα των ανωτέρω χρήσεων, υπάρχουν διάφοροι μηχανισμοί μεταφοράς αρχείων με χρήση των πρωτοκόλλων Secure Shell. Το SSH File Transfer Protocol (SFTP) είναι μια ασφαλέστερη εναλλακτική στο FTP. Το SCP έχει εξελιχθεί από το RCP σε συνδυασμό με το SSH. Τέλος, το πρωτόκολλο FISH (Files transferred over Shell protocol) εκδόθηκε το 1998 και ήταν εξέλιξη της χρήσης εντολών του φλοιού του UNIX πάνω σε ένα ασφαλές κανάλι SSH. Για τα παραπάνω πρωτόκολλα δεν έχουν αναπτυχθεί πρότυπα της IETF. Μια σειρά από προτάσεις της IETF έχουν γίνει για το SFTP, αλλά σταμάτησαν το 2006, εξαιτίας του προβλήματος ότι το SFTP είναι κατ' ουσία ένα σύστημα αρχείων.

2.7.1 Αρχιτεκτονική του πρωτοκόλλου

Η έκδοση 2 του SSH έχει μια εσωτερική αρχιτεκτονική που ορίζεται στο κείμενο RFC 4251, με σαφώς διαχωρισμένα επίπεδα (layers), τα οποία φαίνονται στο παρακάτω σχήμα και είναι:



Εικόνα 10: Αρχιτεκτονική SSH

1. Το επίπεδο μεταφοράς (transport layer) που περιγράφεται στο RFC 4253. Αυτό το επίπεδο αναλαμβάνει την ανταλλαγή του αρχικού κλειδιού και αυθεντικοποίηση του server και ορίζει την επαλήθευση κρυπτογράφησης, συμπίεσης και ακεραιότητας. Παρέχει στο ανώτερο επίπεδο μία διεπαφή (interface) λήψης και αποστολής πακέτων απλών κειμένων (plaintexts) μέχρι και 32768 bytes το καθένα (παραπάνω μπορεί να το επιτρέπει η συγκεκριμένη υλοποίηση). Το επίπεδο μεταφοράς φροντίζει επίσης, για επαναπαλλαγή των κλειδιών συνήθως μετά από μμεταφορά 1 GB δεδομένων ή μετά την πάροδο μίας ώρας (όποιο από τα δύο συμβεί νωρίτερα).
2. Το επίπεδο αυθεντικοποίησης χρήστη (user authentication layer). Το επίπεδο αυτό αναλαμβάνει την αυθεντικοποίηση του πελάτη (client authentication) και παρέχει έναν αριθμό από μεθόδους αυθεντικοποίησης.Η διαδικασία αυθεντικοποίησης εκκινείται από τον πελάτη, όταν καλείται ο χρήστης να εισάγει τον κωδικό πρόσβασης. Συνήθως αυτό γίνεται από το πρόγραμμα SSH του πελάτη και όχι από τον server. Ο server σπάνια απαντά σε αιτήσεις αυθεντικοποίησης του πελάτη. Οι συνήθεις μέθοδοι αυθεντικοποίησης περιλαμβάνουν τα παρακάτω:
   - Κωδικός πρόσβασης (password): Μέθοδος απευθείας αυθεντικοποίησης με κωδικό, που παρέχει δυνατότητα αλλαγής του password. Αυτή η μέθοδος δεν υλοποιείται από όλα τα προγράμματα.
   - Δημόσιο κλειδί (public key): Μέθοδος αυθεντικοποίησης δημοσίου κλειδιού, που συνήθως υποστηρίζει ζευγάρια κλειδιών DSA και RSA, ενώ κάποιες υλοποιήσεις και πιστοποιητικά X.509.
   - Διαδραστικά (keyboard interactive): Ευπροσάρμοστη μέθοδος στην οποία ο server στέλνει μία ή περισσότερες προσκλήσεις για παροχή πληροφοριών και ο πελάτης τις δείχνει στον χρήστη και στέλνει πίσω τις πληροφορίες που πληκτρολογεί αυτός. Χρησιμοποιείται για να παρέχει αυθεντικοποίηση με κωδικό μίας χρήσης, όπως το S/key και SecurID. Μερικές φορές οδηγεί σε αδυναμία σύνδεσης με τον πελάτη που υποστηρίζει μόνο την απλή μέθοδο αυθεντικοποίησης κωδικού πρόσβασης (password authentication), σε ρυθμίσεις του OpenSSH όπου ο μηχανισμός αυθεντικοποίησης του ξενιστή είναι ο PAM (Pluggable Authentication Modules).

- GSSAPI[47]: Παρέχουν ένα εκτεταμένο σχήμα που παρέχει αυθεντικοποίηση SSH, χρησιμοποιώντας εξωτερικούς μηχανισμούς όπως το Kerberos 5 ή το NTLM, δίδοντας δυνατότητα υπογραφής στις συνεδρίες του SSH. Αυτές οι μέθοδοι συνήθως υλοποιούνται από εμπορικό λογισμικό SSH για χρήση σε οργανισμούς, παρόλο που και το OpenSSH έχει μια υλοποίηση που υποστηρίζει το GSSAPI.

3. Το επίπεδο σύνδεσης (connection layer). Το επίπεδο αυτό ορίζει την έννοια των καναλιών, αιτημάτων για κανάλι (channel request) και γενικών αιτημάτων (global request), χρησιμοποιώντας τις υπηρεσίες του SSH που παρέχονται. Μια σύνδεση SSH μπορεί να εξυπηρετεί πολλαπλά κανάλια ταυτόχρονα, καθένα από τα οποία μεταφέρει δεδομένα προς δύο κατευθύνσεις. Τα αιτήματα για κανάλια χρησιμοποιούνται για την αναμετάδοση εκτός πλαισίου δεδομένων που αφορούν το κανάλι, όπως είναι η αλλαγή του μεγέθους του παραθύρου ενός τερματικού ή ο κώδικας εξόδου μιας διαδικασίας από την πλευρά του εξυπηρετητή (server). Το πρόγραμμα του πελάτη του SSH ζητάει την προώθηση της θύρας στην πλευρά του server χρησιμοποιώντας ένα γενικό αίτημα. Οι συνήθεις τύποι καναλιών περιλαμβάνουν:
   - Φλοιός Shell για τερματικά, αιτήματα SFTP και exec (περιλαμβανομένου μεταφορές SCP).
   - Απευθείας TCP-IP για προωθημένες συνδέσεις πελάτη σε εξυπηρετητή. Προωθημένες TCP-IP για προωθημένες συνδέσεις εξυπηρετητή σε πελάτη.
   - Το νέο πεδίο που ονομάζεται SSHFP στο DNS παρέχει ένα αποτύπωμα του δημοσίου κλειδιού του ξενιστή (host) με σκοπό την επαλήθευση της ταυτότητας του ξενιστή.

Η ανοικτή αρχιτεκτονική παρέχει σημαντική ευελιξία, επιτρέποντας το SSH να χρησιμοποιηθεί σε μια πλειάδα εφαρμογών πέρα από τον ασφαλή φλοιό (secure shell). Η λειτουργικότητα του επιπέδου μεταφοράς είναι συγκρίσιμη με αυτή του TLS (Transport Layer Security). Το επίπεδο αυθεντικοποίησης του χρήστη είναι αρκετά επεκτάσιμο με διάφορες μεθόδους αυθεντικοποίησης. Το επίπεδο σύνδεσης παρέχει τη δυνατότητα πολυπλεξίας πολλών δευτερευουσών συνεδριών (sessions) σε μία μοναδική σύνδεση SSH, ένα χαρακτηριστικό αντίστοιχο του πρωτοκόλλου BEEP (Blocks Extensible Exchange Protocol) που

δεν διατίθεται στο TLS. Το BEEP είναι ένα πλαίσιο για τη δημιουργία δικτυακών πρωτοκόλλων εφαρμογών. Περιέχει ένα πυρήνα για την επίτευξη ασύγχρονων συνδέσεων και μπορεί να χρησιμοποιηθεί τόσο για μηνύματα κειμένου, όσο και για δυαδικά (binary), τα οποία ανταλλάσσουν οντότητες εφαρμογών ενός χρήστη.

2.7.2 Ασφάλεια – γνωστές επιθέσεις

Καθόσον το SSH-1 έχει ενδογενή προβλήματα στον σχεδιασμό του που το καθιστούν ευάλωτο σε επιθέσεις τύπου άνθρωπος στη μέση (man-in-themiddle attacks), είναι πλέον γενικώς αποδεκτό ότι θα πρέπει να αποφεύγεται η χρήση του και να μην επιτρέπεται η μετάπτωση στο SSH-1 για χάρη συμβατότητας παλαιότερων συστημάτων. Παρόλο που οι περισσότεροι σύγχρονοι servers υποστηρίζουν SSH-2, μερικοί οργανισμοί χρησιμοποιούν ακόμη λογισμικό που δεν το υποστηρίζει και έτσι η χρήση του SSH-1 δεν μπορεί να αποφευχθεί. Σε όλες τις εκδόσεις του SSH είναι σημαντικό να επαληθευτούν τα δημόσια κλειδιά των ξενιστών (hosts) πριν αυτά γίνουν αποδεκτά από τον χρήστη, διότι σε αντίθετη περίπτωση ο επιτιθέμενος μπορεί να καταφέρει να υποκλέψει το μεταδιδόμενο κωδικό πρόσβασης επιτρέποντας επιθέσεις τύπου man-in-the-middle. Μια σειρά από επιθέσεις αποκάλυψης απλού κειμένου στην υλοποίηση OpenSSH δημοσιεύτηκαν από τους Albrecht, Paterson και Watson το 2009 Στις επιθέσεις αυτές μπορεί να αποκαλυφθούν και να επαληθευτούν 14 bits του απλού κειμένου από ένα επιλεγμένο block του κρυπτοκειμένου με πιθανότητα 2^(-14) και 32 bits του απλού κειμένου με πιθανότητα 2^(-18).

Οι επιθέσεις αυτές γίνονται στην προεπιλεγμένη ρύθμιση των 128 bit κρυπταλγορίθμου τμήματος σε λειτουργία CBC (cipher block chaining). Η δυνατότητα αυτών των επιθέσεων προέρχεται από ένα κενό ασφαλείας στον σχεδιασμό του πρωτοκόλλου στο κείμενο RFC που περιγράφει το SSH BPP (Binary Packet Protocol). Το BPP είναι το τμήμα του πρωτοκόλλου SSH που είναι υπεύθυνο για να παρέχει υπηρεσίες εμπιστευτικότητας και ακεραιότητας σε όλα τα μηνύματα που ανταλλάσσονται σε μια σύνδεση SSH. Καταρχήν, το SSH έχει ένα κρυπτογραφημένο πεδίο μήκους στο πρώτο τμήμα του κρυπτοκειμένου, το οποίο χρησιμοποιείται για να προσδιορίσει πόσα δεδομένα αναμένονται για ένα συγκεκριμένο πακέτο και πρέπει να ληφθούν υπόψη στον υπολογισμό της τιμής MAC, πριν αυτή είναι δυνατό να επαληθευτεί. Εν συνεχεία, η εξάρτηση από τον τρόπο λειτουργίας του CBC (ακόμα και με αλυσίδα αρχικών διανυσμάτων IV),

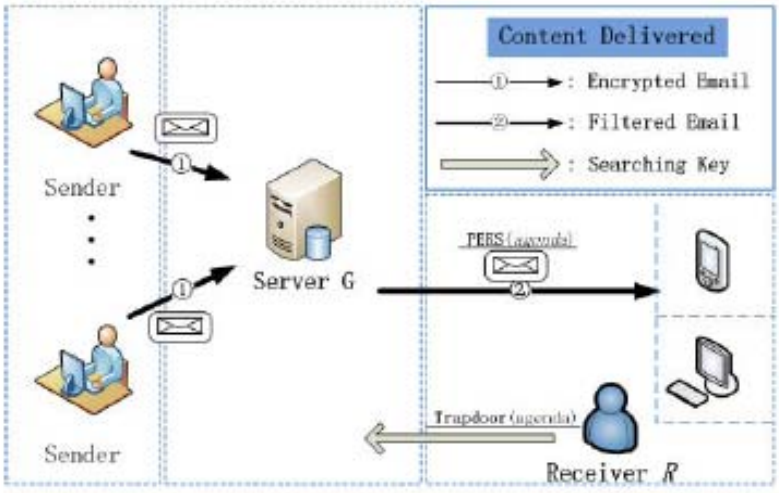
επιτρέπει στον επιτιθέμενο να εισάγει ένα block επιλεγμένου κρυπτοκειμένου σε ένα νέο πακέτο BPP ως το πρώτο block του πακέτου και η αποκρυπτογράφηση αυτού του επιλεγμένου τμήματος στην αρχική του θέση να σχετίζεται με ένα γνωστό τρόπο με την αποκρυπτογράφηση του νέου πακέτου. Αυτοί οι δύο παράγοντες, σε συνδυασμό με την ικανότητα του επιτιθέμενου να εισάγει δεδομένα ανά block σε μια σύνδεση SSH και να ανιχνεύσει πότε συμβαίνει ένα λάθος στην επαλήθευση της τιμής MAC, επιτρέπουν στον επιτιθέμενο να αποκαλύψει κάποια bits του απλού κειμένου που σχετίζονται με ένα επιλεγμένο κρυπτοκείμενο, παρατηρώντας πόσα blocks χρειάζονται για να προκαλέσουν ένα σφάλμα στην τιμή του MAC. Οι έλεγχοι μήκους του OpenSSH σπάνια είναι ένας παράγοντας που μειώνει τις πιθανότητες επιτυχίας αυτών των επιθέσεων.

Οι επιθέσεις αυτές έγιναν σε μια υλοποίηση του SSH BPP, η οποία είχε αποδειχθεί ασφαλής, σε προγενέστερη εργασία των M. Bellare,T. Kohno και C. Namprempre (2004). Ωστόσο, στις παραδοχές του μοντέλου το οποίο χρησιμοποιήθηκε για την απόδειξη της ασφάλειας, δεν ελήφθησαν ορισμένοι παράγοντες υπόψη. Για παράδειγμα, ενώ αναγνωρίζεται ότι η λειτουργία αποκρυπτογράφησης στο SSH δεν είναι μονοδιάστατη και μπορεί να αποτύχει με διάφορους τρόπους, το μοντέλο ασφαλείας δεν κάνει διάκριση μεταξύ των διάφορων τύπων σφάλματος, όταν αυτά αναφέρονται στο άλλο μέρος της σύνδεσης. Επιπλέον, το μμοντέλο δεν λαμβάνει σαφώς υπόψη του το γεγονός ότι το ποσό δεδομένων που χρειάζεται για να συμπληρωθεί η λειτουργία αποκρυπτογράφησης προσδιορίζεται από τα δεδομένα που πρέπει να αποκρυπτογραφηθούν (το μήκος πεδίου). Δυστυχώς, φαίνεται ότι οι κρυπτογραφικές υλοποιήσεις πραγματικών συστημάτων είναι πιο πολύπλοκες από τα σημερινά μοντέλα ασφαλείας που περιγράφουν το SSH. Το 2010 παρουσιάστηκε από τους Keneth G. Paterson και Gaven J. Watson, ένα εκτεταμένο μοντέλο ανάλυσης που αποδεικνύει την ασφάλεια του SSH σε λειτουργία CTR (counter mode). Το μοντέλο αυτό περιγράφει το SSHCTR σχετιζόμενο στενά με τις προδιαγραφές των κειμένων RFC και την υλοποίηση OpenSSH. Η προσέγγιση του μοντέλου λαμβάνει υπόψη της και τις επιθέσεις που αναφέρθηκαν στην προηγούμενη παράγραφο και το γεγονός αυτό βοηθά στην εξάλειψη του κενού ανάμεσα στην τυπική ανάλυση ασφάλειας του SSH και στον τρόπο που θα έπρεπε να λειτουργεί το SSH και υλοποιείται στην πράξη. Επίσης, η προσέγγιση αυτή που παρουσιάστηκε στην προαναφερόμενη δημοσίευση, είναι μια προσπάθεια να μεγεθύνει το εύρος της αποδείξιμης ασφάλειας, ώστε να

ενοποιήσει τις λεπτομέρειες των κρυπτογραφικών υλοποιήσεων. Θεωρεί δε ένα μεγαλύτερο και πιο ρεαλιστικό σύνολο από τρόπους αλληλεπίδρασης του επιτιθέμενου με το πρωτόκολλο, από ότι η προηγούμενη ανάλυση. Η προσέγγιση αυτή καταγράφει περισσότερα από τα κρυπτογραφικά χαρακτηριστικά του SSH BBP, όπως αυτά που σχετίζονται με την εξάρτηση από το απλό κείμενο, τη σειρά αποκρυπτογράφησης των bytes και τον τρόπο μοντελοποίησης των σφαλμάτων που μπορεί να προκύψουν κατά την επεξεργασία αποκρυπτογράφησης.

2.8 Κρυπτογραφία δημοσίου κλειδιού με αναζήτηση

Public-key Encryption with Keyword Search(PEKS)[48] είναι το πρώτο, πρακτικό, ασύμμετρο, σύστημα κρυπτογράφησης με αναζήτηση. Είχε σχεδιαστεί αρχικά για το σκοπό της έξυπνης δρομολόγησης ηλεκτρονικών μηνυμάτων (e-mail). Στη εικόνα 11 απεικονίζετε η υλοποίηση της τεχνικής αυτής. Ο αποστολέας στέλνει ένα κρυπτογραφημένο κείμενο, ο διακομιστής θα προωθήσει το μήνυμα στο παραλήπτη. Ο παραλήπτης όμως θέλει, αν στο μήνυμα αναφέρετε μια λέξη π.χ «επείγων» το μήνυμα να προωθείτε στο κινητό του τηλέφωνο χωρίς όμως ο διακομιστής να αποκρυπτογραφά ολόκληρο το κρυπτογραφημένο μήνυμα. Έτσι ο παραλήπτης κρυπτογραφεί και αποθηκεύει, με τέτοιο τρόπο την λέξη που θέλει να ερευνάτε αν αναφέρεται στο κείμενο, στο διακομιστή, ούτως ώστε να μπορεί να παραδώσει σωστά το μήνυμα και ταυτόχρονα να μην γνωρίζει τι αναγράφετε σε ολόκληρο το κρυπτογράφημα.



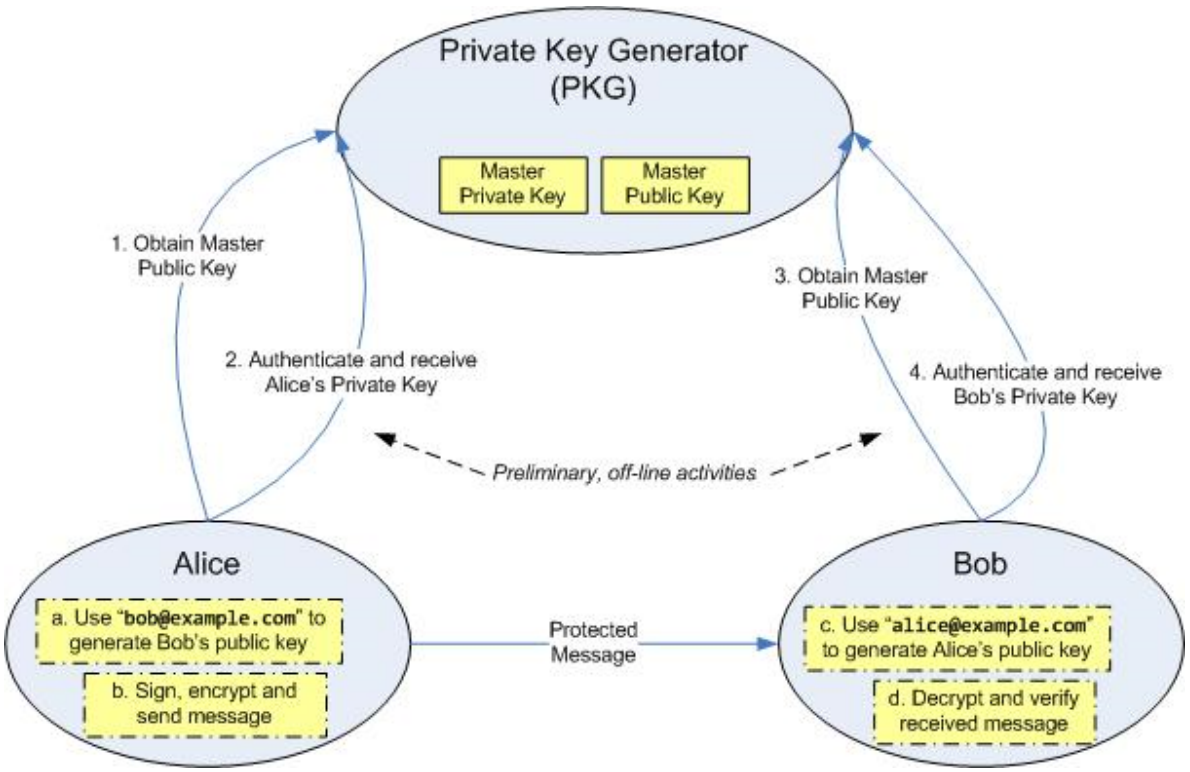
Εικόνα 11: Διαδικασία κρυπτογράφησης με αναζήτηση

2.9 Κρυπτογραφία βασισμένη στην ταυτότητα

Identity based encryption (IBE) [49,50] είναι ένας τύπος κρυπτογραφίας δημοσίου κλειδιού, όπου το δημόσιο κλειδί ενός χρήστη, μπορεί να είναι κάποια

μοναδική πληροφορία σχετικά με την ταυτότητά του, όπως το ονοματεπώνυμο ή την ταχυδρομική του διεύθυνση. Τη πρώτη υλοποίηση ενός συστήματος κρυπτογραφίας δημοσίου κλειδιού, με βάση την ηλεκτρονική διεύθυνση, παρουσίασε ο Adi Shamir [51] το 1984, στο οποίο επιτρεπόταν στους χρήστες να επαληθεύουν ψηφιακές υπογραφές χρησιμοποιώντας μόνο τις δημόσιες πληροφορίες τους.

Στην εικόνα 12 παρουσιάζονται τα βήματα που περιλαμβάνει η τεχνική αυτή. Το σύστημα επιτρέπει σε κάθε συμβαλλόμενο μέρος να αναπαράγει ένα δημόσιο κλειδί από μία γνωστή τιμή ταυτότητας όπως είναι μια ακολουθία ASCI. Για να λειτουργήσει η τεχνική, ένα έμπιστο τρίτο μέρος που ονομάζετε γεννήτρια ιδιωτικού κλειδιού (Private Key Generator PKG) δημιουργεί τα αντίστοιχα ιδιωτικά κλειδιά. Γνωρίζοντας το δημόσιο κλειδί, οποιοσδήποτε συμβαλλόμενος, μπορεί να υπολογίσει ένα δημόσιο ειδικό κλειδί που να αντιστοιχεί στην ταυτότητα συνδυάζοντας το δημόσιο κλειδί με την τιμή της ταυτότητας, της οντότητας. Για να ληφθεί ένα αντίστοιχο ειδικό ιδιωτικό κλειδί, η αρχή που είναι αρμόδια να χρησιμοποίηση την ταυτότητα, επικοινωνεί με την γεννήτρια, η οποία χρησιμοποιεί το ιδιωτικό κλειδί για να δημιουργήσει το ειδικό ιδιωτικό κλειδί της οντότητας. Ως αποτέλεσμα, οι συμβαλλόμενοι μπορούν να κρυπτογραφήσουν μηνύματα χωρίς την προηγουμένως διανομή κλειδιών μεταξύ των συμμετεχόντων. Αυτό είναι εξαιρετικά χρήσιμο σε περιπτώσεις κατά τις οποίες η προ-διανομή επικυρωμένων κλειδιών είναι ανασφαλή ή μη εφικτή λόγω τεχνικών περιορισμών. Ωστόσο για την αποκρυπτογράφηση μηνυμάτων, ο εξουσιοδοτημένος χρήστης πρέπει να αποκτήσει το κατάλληλο ιδιωτικό κλειδί από την γεννήτρια. Το σημαντικότερο μειονέκτημα αυτής της προσέγγισης είναι ότι η γεννήτρια ιδιωτικών κλειδιών πρέπει να είναι εξαιρετικά έμπιστη καθώς έχει τη δυνατότητα με το ιδιωτικό κλειδί που μοιράζει, να αποκρυπτογραφήσει μήνυμα οποιοδήποτε χρήστη χωρίς να χρειάζεται εξουσιοδότηση.



Εικόνα 12 Υλοποίηση διαδικασίας κρυπτογραφίας με ταυτότητα

2.10 Format-preserving encryption (FPE)

Στη κρυπτογραφία η μέθοδος FPE[52] αναφέρετε στη κρυπτογράφηση με τέτοιο τρόπο ώστε η μορφή των δεδομένων που πρόκειται να κρυπτογραφηθούν να μην μεταβάλλετε. Η έννοια του 'μορφή' ποικίλη, πρακτικά έχουν εξεταστεί μόνο πεπερασμένοι τομείς π.χ Η κρυπτογράφηση ενός δεκαεξαψήφιου αριθμού πιστωτικής κάρτας έτσι ώστε το κρυπτογράφημα να είναι ένας άλλος δεκαεξαψήφιος αριθμός. Η κρυπτογράφηση μίας αγγλικής λέξης, με αποτέλεσμα μία άλλη αγγλική λέξη. Η κρυπτογράφηση ενός v-οστού αριθμού με αποτέλεσμα έναν άλλο v-οστό όρο.

Ένα από τα κίνητρα που ώθησαν στην ανάπτυξη του FPE είναι τα προβλήματα που σχετίζονται με την ενσωμάτωση της κρυπτογράφησης σε ήδη υπάρχουσες εφαρμογές. Υποθέτουμε ότι θέλουμε να κρυπτογραφήσουμε ένα δεκαεξαψήφιο αριθμό πιστωτικής κάρτας 0000-0000-0000-0000.Κάνοντας χρήση του αλγορίθμου AES ο αριθμός αυτός θα μεταμορφωθεί σε μία μεγάλη ακολουθία προκαθορισμένου μήκους δυαδικής τιμής του τύπου 0x96a45cbcf9c2a9425cde9e274948cb67, η οποία περιέχει πολλά bytes τα οποία θεωρούνε άκυρα όταν συγκρίνονται με ένα συνηθισμένο αριθμό πιστωτικής κάρτας. Αν ένας αριθμός πιστωτικής κάρτας πρόκειται να αποθηκευτεί σε μία στήλη μιας βάσης δεδομένων τις οποίας οι καταχωρίσεις είναι *char* ή *varchar*

δεδομένα, τότε το κρυπτογραφημένα δεδομένα δεν θα μπορούν να αποθηκεύονται στην ιδία στήλη χωρίς να αλλάζει η μορφή της στήλης. Σε τέτοιες περιπτώσεις, οι εφαρμογές που χειρίζονται αριθμούς πιστωτικών καρτών δεν θα μπορούν να χειριστούν κρυπτογραφημένες τιμές χωρίς κάποια τροποποίηση.

2.11 Αξιολόγηση τεχνικών κρυπτογράφησης στο cloud

Στις ποιο πάνω ενότητες αναφέρθηκαν οι σημαντικότερες τεχνικές κρυπτογράφησης καθώς και οι ευπάθειες τους, σε διάφορους τύπους επιθέσεων. Στο cloud όμως κάποιες από αυτές τις τεχνικές δεν μπορούν να χρησιμοποιηθούν για την επικοινωνία και την δημιουργία μιας ασφαλούς σύνδεσης μεταξύ πελάτη και εξυπηρετητή. Οι αλγόριθμοι κρυπτογράφησης όπως AES, DES, RSA, και οι συναρτήσεις κατακερματισμού, μπορούν να χρησιμοποιηθούν για την κρυπτογράφηση των data at rest (τα δεδομένα που δεν τυγχάνουν συνεχής χρήσης). Οι τεχνικές κρυπτογραφίας FPE, IBE, και PEKS επινοήθηκαν και κατασκευάστηκαν για συγκεκριμένους σκοπούς διευκόλυνσης και αντιμετώπισης κάποιων εξειδικευμένων λειτουργιών, δεν είναι σε θέση από μόνες τους να διαφυλάξουν εξολοκλήρου την ασφάλεια της επικοινωνίας σε ένα σύστημα cloud. Για την διασφάλιση της ασφαλούς επικοινωνίας μεταξύ του νέφους και του πελάτη χρησιμοποιείτε μία σύνδεση SSL/TLS για να δημιουργηθεί ένα ασφαλές κανάλι μέσα στο οποίο θα μεταφερθούν οι πληροφορίες. Ανταλλάσσονται κρυπτογραφημένα κάποιες πληροφορίες μεταξύ των οποίων τα κλειδιά και ο αλγόριθμός κρυπτογράφησης με τον οποίο θα κρυπτογραφηθούν για να ταξιδέψουν οι πληροφορίες. Ο αλγόριθμος θα ήταν καλό να επιλεχθεί με κριτήρια του τύπου, ανάλυσης του ρίσκου υποκλοπής, της σημαντικότητας των δεδομένων όπως επίσης και του χρόνου κρυπτογράφησης και μεταφοράς των δεδομένων.

Αλγόριθμοι	**Πλεονεκτήματα**	**Μειονεκτήματα**	**Κατάλληλο για cloud**
DES	Γρήγορος, απλός αλγόριθμος εύκολος σε υλοποίηση, χαμηλή κατανάλωση υπολογιστικής ισχύς.	Χαμηλή ασφάλεια, ανταλλαγή κλειδιών, ανάγκη ασφαλούς καναλιού.	**ΜΗ ΚΑΤΑΛΛΗΛΟ**

Triple-DES	Πολύ καλή ασφάλεια.	Υψηλή κατανάλωση υπολογιστικής ισχύς, πολύ αργός αλγόριθμος, ανταλλαγή κλειδιών, ανάγκη ασφαλούς καναλιού.	**ΜΗ ΚΑΤΑΛΛΗΛΟ**
AES	Χρήση διάφορου μήκους κλειδιών (128, 192, 256 bits), πολύ καλή ασφάλεια, γρήγορος αλγόριθμος με χαμηλή κατανάλωση υπολογιστικής ισχύς.	Ανταλλαγή κλειδιών, ανάγκη ασφαλούς καναλιού.	**ΚΑΤΑΛΛΗΛΟ**
RSA	Αποφυγή διανομής κλειδιών, χρήση ψηφιακής υπογραφής, καλή ασφάλεια.	Η απώλεια ιδιωτικού κλειδιού είναι σοβαρό πρόβλημα, πιο αργός από τους προηγούμενους αλγόριθμους με περισσότερη κατανάλωση υπολογιστικής ισχύς.	**ΜΗ ΚΑΤΑΛΛΗΛΟ**
SSL	Πολύ καλό επίπεδο ασφάλειας, δημιουργία ασφαλούς καναλιού για επικοινωνία. Απαραίτητη για χρηματικές συναλλαγές.	Αργή επικοινωνία, ακριβή εγκατάσταση και συντήρηση, περίπλοκη υλοποίηση.	**ΚΑΤΑΛΛΗΛΟ**
SSH	Πολύ καλό επίπεδο ασφάλειας, δημιουργία ασφαλούς καναλιού για επικοινωνία. Απαραίτητη για χρηματικές συναλλαγές.	Αργή επικοινωνία, ακριβή εγκατάσταση και συντήρηση, περίπλοκη υλοποίηση.	**ΚΑΤΑΛΛΗΛΟ**

MD5	Σχετικά γρήγορος για το επίπεδο ασφάλειας που παρέχει. Παράγει ένα μοναδικό κρυπτογραφημένο αρχείο που αν αλλάξει έστω και ένα bit ανιχνεύεται.	Έχει γνωστά τρωτά σημεία.	**ΚΑΤΑΛΛΗΛΟ**
PEKS	Εξυπηρετεί την αναζήτηση κρυπτογραφημάτων για συγκεκριμένες λέξεις.	Δεν μπορεί από μόνη της σαν τεχνική να υποστηρίξει ένα σύστημα	**ΚΑΤΑΛΛΗΛΟ**
IBE	Χρήσιμη τεχνική όταν δεν μπορεί να υπάρξει προ-διανομή κλειδιών. Χρησιμοποιείτε για ταυτοποίηση χρήστη βάση των στοιχείων του.	Η γεννήτρια κλειδιών πρέπει να είναι έμπιστη. Δεν έχει τη δυνατότητα να διαφυλάξει την ασφάλεια σε ένα δίκτυο.	**ΚΑΤΑΛΛΗΛΟ**
FPE	Χρήσιμη τεχνική για κρυπτογράφηση δεδομένων σταθερού τύπου π.χ. αρ. πιστωτικών καρτών, ταυτοτήτων.	Δεν μπορεί από μόνη της σαν τεχνική να υποστηρίξει ένα σύστημα.	**ΚΑΤΑΛΛΗΛΟ**

ΚΕΦΑΛΑΙΟ 3. ΜΟΝΤΕΛΑ ΚΑΙ ΤΕΧΝΙΚΕΣ ΔΙΑΝΟΜΗΣ ΚΛΕΙΔΙΩΝ ΣΤΟ CLOUD

Εισαγωγή

Στη κρυπτογραφία συμμετρικού κλειδιού και τα δύο μέλη πρέπει έχουν στη κατοχή τους ένα μυστικό κλειδί, το οποίο πρέπει να ανταλλάξουν πριν να γίνει χρήση οποιουδήποτε κρυπτογραφικού αλγόριθμου. Η διανομή των κλειδιών ήταν προβληματική μέχρι και πρόσφατα, επειδή περιλαμβάνει συνάντηση πρόσωπο με πρόσωπο, χρήση ενός έμπιστου μεταφορέα, ή στέλνοντας το κλειδί μέσο ενός υπάρχοντος κρυπτογραφικού καναλιού. Οι δύο πρώτοι τρόποι δεν είναι πρακτικοί και πάντα ασφαλής, ενώ ο τρίτος εξαρτάται από την ασφάλεια της προηγούμενης ανταλλαγής κλειδιών.

Στη κρυπτογραφία δημοσίου κλειδιού, η διανομή των δημοσίων κλειδιών γίνεται μέσω διακομιστών δημοσίου κλειδιού. Όταν μια οντότητα δημιουργήσει ένα ζεύγος κλειδιών, το ιδιωτικό-κρυφό κλειδί το αποθηκεύει η οντότητα, και το δημόσιο κλειδί αποθηκεύετε σε ένα διακομιστή όπου μπορεί να το δει ο οποιοσδήποτε θέλει, και να στείλει στην οντότητα ένα προσωπικό κρυπτογραφημένο μήνυμα.

Στην ανταλλαγή μυστικών(secret sharing), ένα μυστικό χρησιμοποιείτε για την αρχική δημιουργία ενός αριθμού διακριτών μυστικών, και τα κομμάτια διανέμονται έτσι ώστε κάποιο υποσύνολο των αποδεκτών να μπορεί να πιστοποιήσει τον εαυτό τους από κοινού, και να μπορεί να χρησιμοποιήσει τις μυστικές πληροφορίες χωρίς να ξέρει τι είναι.

Στο κεφάλαιο αυτό θα περιγραφούν κάποια είδη υπάρχουσα μοντέλα διανομής κλειδιών, για την επίλυση κάποιων προβλημάτων που παρουσιάζονται στο δίκτυο του νέφους.

3.1 Υποδομή δημοσίου κλειδιού

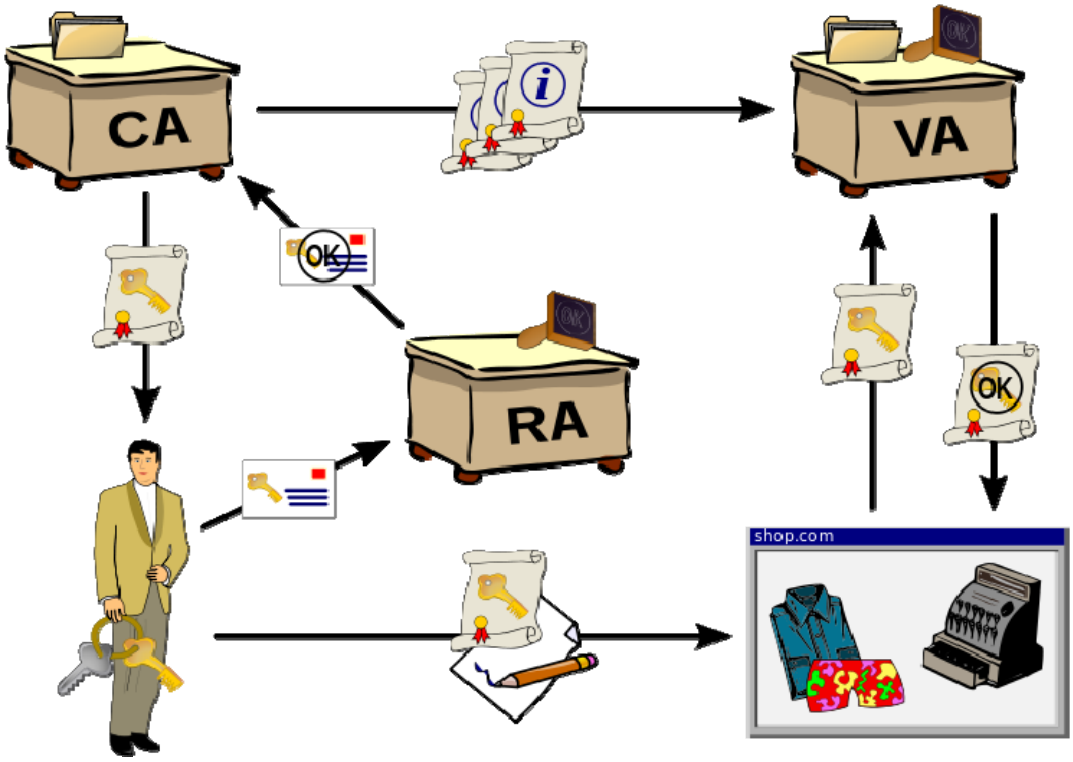
Η Υποδομή Δημόσιου Κλειδιού (Public Key Infrastructure) [53] αποτελεί ένα συνδυασμό λογισμικού, τεχνολογιών ασύμμετρης κρυπτογραφίας και διαδικασιών, ο οποίος πιστοποιεί την εγκυρότητα του κάθε εμπλεκόμενου σε μια ψηφιακή συναλλαγή. Η Υποδομή Δημοσίου κλειδιού πιστοποιεί την ταυτότητα μιας πιστοποιημένης οντότητας υπογράφοντας το δημόσιο κλειδί της και δημοσιεύοντας το, μαζί με πληροφορίες σχετικά με την ταυτότητα της οντότητας, σε ένα

πιστοποιητικό. Παράλληλα διατηρεί καταλόγους με τα έγκυρα, τα ληγμένα αλλά και τα ανακληθέντα πιστοποιητικά.

Οι κυριότεροι μηχανισμοί ασφάλειας τους οποίους καλύπτει η Υποδομή Δημόσιου Κλειδιού είναι η εξής:

- ***Απόρρητο της επικοινωνίας***: Τα δεδομένα προστατεύονται από μη εξουσιοδοτημένη πρόσβαση με μηχανισμούς ελέγχου πρόσβασης στην περίπτωση αποθήκευσης δεδομένων και μέσω κρυπτογράφησης κατά την αποστολή τους.
- ***Ακεραιότητα***: Τα δεδομένα προστατεύονται από μη εξουσιοδοτημένη τροποποίηση μέσω μηχανισμών κρυπτογράφησης όπως οι ηλεκτρονικές υπογραφές.
- ***Πιστοποίηση***: Πραγματοποιείται επιβεβαίωση της ταυτότητας ενός ατόμου ή της πηγής αποστολής δεδομένων.
- ***Μη άρνηση Αποδοχής***: Η Μη άρνηση αποδοχής συνδυάζει τις υπηρεσίες της Πιστοποίησης και της Ακεραιότητας και διασφαλίζει ότι μία συναλλαγή η οποία πραγματοποιήθηκε ηλεκτρονικά δε μπορεί να αμφισβητηθεί από τα συμβαλλόμενα μέρη.

Στη πιο κάτω εικόνα απεικονίζετε η σειρά των διεργασιών που πρέπει να υλοποιηθούν στη υποδομή αυτή. Η οντότητα πρέπει πρώτα να επικοινωνήσει με την αρχή πιστοποίησης (Registry Authority RA). Η αρχή πιστοποίησης εφόσον εξακριβώσει τα στοιχεία της οντότητας αποστέλλει τις πληροφορίες αυτές στην αρχή πιστοποιητικών (Certificate Authority CA).Η CA ενημερώνει για την καινούρια οντότητα την αρχή πιστοποίησης (Validation Authority VA) και αποστέλλει στον χρήστη το ιδιωτικό του κλειδί. Η οντότητα κωδικοποιεί ένα κείμενο με το ιδιωτικό κλειδί της, και αποστέλλει το κρυπτογραφημένο κείμενο με το πιστοποιητικό του σε αυτόν που θέλει να επικοινωνήσει. Το πιστοποιητικό του επαληθεύετε από την αρχή πιστοποίησης η οποία επιστρέφει το από-κρυπτογραφημένο μήνυμα για να ολοκληρωθεί η επικοινωνία.



Εικόνα 13: Διαδικασία υποδομής δημοσίου κλειδιού

3.1.1 Πρότυπο X.509

Το X.509[54] είναι ένα διεθνές πρότυπο που καθορίζει τον τρόπο λειτουργίας των Υποδομών Δημοσίου Κλειδιού (Public Key Infrastructure/PKI). Προδιαγράφει τις μορφές διάθεσης της σχετικής πληροφορίας (κλειδιά, πιστοποιητικά, λίστες ανάκλησης), καθώς και τους αλγορίθμους επαλήθευσης του κύρους ενός πιστοποιητικού. Στο X.509, η Αρχή Πιστοποίησης εκδίδει ένα πιστοποιητικό, το οποίο περιλαμβάνει πληροφορίες για μια οντότητα. Η οντότητα του πιστοποιητικού μπορεί να είναι ένα όνομα (common name) ή κάποια άλλη οντότητα (alternative name), όπως μια διεύθυνση e-mail, μια διεύθυνση IP ή ένα όνομα DNS.

Ένα πιστοποιητικό X.509 περιέχει τις ακόλουθες πληροφορίες:

- Πιστοποιητικό
  - Έκδοση X.509 (π.χ. 3)
  - Αύξων Αριθμός
  - Αλγόριθμος
  - Εκδοθείσα Αρχή Πιστοποίησης
  - Περίοδος εγκυρότητας:
    - Όχι νωρίτερα από

Όχι αργότερα από

Υποκείμενο

Πληροφορίας δημόσιου κλειδιού υποκειμένου:

Αλγόριθμος δημόσιου κλειδιού

Τιμή δημόσιου κλειδιού

Προαιρετικές πληροφορίες

Επεκτάσεις (extensions)

- Αλγόριθμος υπογραφής πιστοποιητικού
- Υπογραφή πιστοποιητικού

Η Αρχή Πιστοποίησης υπογράφει ψηφιακά τις πληροφορίες του πιστοποιητικού, ώστε όποιος διαθέτει το δημόσιο κλειδί της, να μπορεί να επαληθεύσει την ισχύ των πληροφοριών που φέρει το πιστοποιητικό. Κάθε Αρχή Πιστοποίησης διαθέτει με τη σειρά της ένα δικό της πιστοποιητικό, υπογεγραμμένο από κάποια άλλη αρχή πιστοποίησης ή από τον εαυτό της. Επομένως η επαλήθευση της εγκυρότητας ενός πιστοποιητικού μπορεί να γίνει ακολουθώντας μια αλυσίδα Αρχών Πιστοποίησης, μέχρι να φτάσουμε σε μια έμπιστη Αρχή Πιστοποίησης (trusted CA). Κάθε λειτουργικό σύστημα ή πρόγραμμα περιήγησης του διαδικτύου (browser), διανέμεται με έναν κατάλογο έμπιστων , πράγμα το οποίο σημαίνει ότι μπορεί να επαληθεύσει την εγκυρότητα ενός μεγάλου αριθμού πιστοποιητικών.

3.2 Internet Key Exchange protocol (IKE)

Το πρωτόκολλο IKE[55] όπως περιγράφετε στην αναφορά RFC2409, είναι ένα πρωτόκολλο διαχείρισης προτύπων το οποίο χρησιμοποιείτε σε μία σύνδεση με το πρότυπο IPsec.

Το IKE είναι ένα υβριδικό πρωτόκολλο βασισμένο στο **Internet security Association and Key Management Protocol (ISAKMP)**, το οποίο περιγράφετε στην αναφορά RCF 2408. Το πρωτόκολλο IKE υλοποιεί κομμάτια από δύο άλλα πρωτόκολλα διαχείρισης κλειδιών, το **Oakley** το οποίο περιγράφετε στην αναφορά RCF 2412 και το πρωτόκολλο **SKEME**[56]. Η ασφάλεια του κρυπτογραφικού αυτού πρωτοκόλλου πηγάζει από τη δυσκολία υπολογισμού διακριτών λογαρίθμων σε ένα πεπερασμένο σώμα. Η πολιτική προστασίας μέσα στο IPsec διαπραγματεύεται και διεκπεραιώνεται με τη βοήθεια του πρωτοκόλλου ISAKMP, τα κλειδιά συνόδου

για κρυπτογράφηση και αυθεντικοποίηση ορίζονται και ανταλλάσσονται με τη χρήση των πρωτοκόλλων Oakley και SKEME.

3.2.1 Το πρωτόκολλο IPsec

Το **IPsec** (Internet Protocol Security) [57] είναι ένα σύνολο πρωτοκόλλων που εξασφαλίζει την ασφάλεια της επικοινωνίας, αυθεντικοποιώντας και κρυπτογραφώντας κάθε πακέτο IP. Το IPsec περιλαμβάνει επίσης και πρωτόκολλα για την επίτευξη αμοιβαίας αυθεντικοποίησης μεταξύ των επικοινωνούντων μερών στην αρχή κάθε συνεδρίας (session) και συμφωνίας κρυπτογραφικών κλειδιών που θα χρησιμοποιηθούν στη διάρκεια της. Το IPsec υλοποιεί ένα σχήμα ασφαλείας(**S**ecurity **A**ssociation **SA**) μεταξύ δύο σημείων (end to end) που μπορεί να χρησιμοποιηθεί για να προστατέψει ροές δεδομένων μεταξύ δύο ξενιστών (host to host), μεταξύ δύο πυλών ασφαλείας (network to network) ή μεταξύ μιας πύλης ασφαλείας και ενός ξενιστή (network to host). Οι εφαρμογές δεν χρειάζεται να είναι ειδικά σχεδιασμένες ώστε να μπορούν να αξιοποιήσουν το IPsec, καθόσον αυτό λειτουργεί στο Internet Layer του πρωτοκόλλου IP, και όχι στα ανώτερα στρώματα (layers) της στοίβας πρωτοκόλλων, όπως στην περίπτωση του SSL-TLS και του SSH. Το IPsec χρησιμοποιεί διάφορα πρωτόκολλα που εκτελούν αντίστοιχες λειτουργίες, για αυτό και θεωρείται ως ανοικτό πρότυπο.

3.2.2 Internet security Association and Key Management Protocol (ISAKMP)

Το ISAKMP είναι διεκπεραιώνει τη διαχείριση μιας ασφαλούς συνόδου μεταξύ των IPsec peers, η οποία χρησιμοποιείτε για τη διαπραγμάτευση των IPsec SAs(σχήμα ασφαλείας). Το ISAKMP παρέχει τα μέσα για την υλοποίηση των παρακάτω:

- Αυθεντικοποίηση της απομακρυσμένης επικοινωνίας
- Παρέχει κρυπτογραφική προστασία στη διαχείριση της συνόδου
- Ανταλλάζει πληροφορίες για την ανταλλαγή των κλειδιών
- Διαπραγματεύεται όλες τις παραμέτρους προστασίας της κυκλοφορίας χρησιμοποιώντας ρυθμισμένες πολιτικές ασφαλείας

Επομένως ο στόχος του ISAKMP είναι η διεκπεραίωση ενός ανεξαρτήτου ασφαλούς καναλιού μεταξύ αυθεντικοποιημένων συνδέσεων προκειμένου να υλοποιηθεί μία ασφαλής ανταλλαγή κλειδιών μεταξύ των IPsec SAs.

3.2.3 Πρωτόκολλο OAKLEY

Το πρωτόκολλο Oakley ένα πρωτόκολλο ελεύθερης μορφής το οποίο επιτρέπει στα επικοινωνούντα μέρη να προχωρούν στην ανταλλαγή μηνυμάτων με την δική τους ταχύτητα. Το πρωτόκολλο IKE δανείστηκε αυτή την ιδέα από το Oakley και με αυτό το πρωτόκολλο ορίζει τους μηχανισμούς για ανταλλαγή κλειδιών με διάφορους τρόπους στη σύνοδο IKE(ISAKMP). Κάθε πρωτόκολλο παράγει ένα παρόμοιο αποτέλεσμα-μία αυθεντικοποιημένη ανταλλαγή κλειδιών, αποδίδοντας έμπιστο υλικό που χρησιμοποιείτε στα IPsec SAs. Το Oakley μέσα στο πρωτόκολλο IKE προσδιορίζει μέσο του Authentication Header(AH[58]) και του Encapsulation Security Payload (ESP[58]) την αυθεντικοποίηση και κρυπτογράφηση των κλειδιών συνόδου για κάθε IPsec SA αυτόματα, και από επιλογή χρησιμοποιεί τον αλγόριθμο Diffie-Hellman για να το πετύχει.

3.2.4 Ο αλγόριθμος Diffie-Hellman

Ο αλγόριθμος Diffie-Hellman[59] ανακαλύφθηκε από τους Whitfield Diffie and Martin Hellman το 1976. Εξασφαλίζει το επίπεδο της ασφάλειάς του από τη δυσκολία του υπολογισμού των διακριτών αλγορίθμων των πολύ μεγάλων αριθμών. Ο αλγόριθμος αυτός χρησιμοποιείτε για την ασφαλή ανταλλαγή κλειδιών σε ανασφαλή κανάλια επικοινωνίας και επίσης γίνετε εκτεταμένη χρήση του στη μοντέρνα διαχείριση κλειδιών για να παρέχει κρυπτογραφικό υλικό για άλλους συμμετρικούς αλγόριθμους όπως τον DES ή τον MD5-με-κλειδί [60](HMAC).

Προκειμένου να ξεκινήσει ο Diffie-Hellman τα δύο μέρη πρέπει να συμφωνήσουν και να ανταλλάξουν δύο μη μυστικούς αριθμούς. Ο πρώτος αριθμός είναι ο *g* (generator) η γεννήτρια και ο δεύτερος ο *p* (modulus). Αυτοί οι αριθμοί μπορούν να δημοσιευτούν και συνήθως επιλέγονται από ένα πίνακα με γνωστές τιμές. Η γεννήτρια είναι συνήθως ένας πολύ μικρός αριθμός (π.χ 2,3…), και το *p* είναι ένας πολύ μεγάλος πρώτος αριθμός. Κάθε μέρος υπολογίζει τη δική του ιδιωτική τιμή. Μετά με βάση το *g*, *p* και τη ιδιωτική τιμή τους, το κάθε μέρος ξεχωριστά υπολογίζει την δημόσια τιμή(κλειδί) του. Το κλειδί υπολογίζετε από την παρακάτω φόρμουλα Y=g^x mod p όπου το χ το ιδιωτικό κλειδί τις οντότητας, και το Y το δημόσιο κλειδί. Μετά από αυτό τα δύο μέρη ανταλλάζουν τα δημόσια κλειδιά τους. Στη συνέχεια τα δύο μέρη υψώνουν στη ιδιωτική του τιμή την δημόσια τιμή του άλλου μέρους για να υπολογίσουν και να κατέχουν ένα κοινό μυστικό. Όταν ο αλγόριθμος ολοκληρωθεί και τα δύο μέρη έχουν το ίδιο κοινό μυστικό το οποίο υπολόγισαν με τις ιδιωτικές του τιμές και τις δημόσιες του αντίθετου μέρους.

Οποιοσδήποτε εισβολέας και αν παρακολουθούσε το κανάλι αυτό δεν μπορεί να υπολογίσει το κοινό μυστικό αφού θα γνώριζε μόνο το *g*, το *p* και το *Y* και των δύο μελών αλλά δεν θα γνώριζε τη κρυφή τους τιμή, αφού δεν την αντάλλαξαν και δεν ταξιδέψε στο κανάλι επικοινωνίας, και χωρίς αυτό δεν μπορεί να υπολογίσει το κοινό μυστικό.

3.2.5 Σύνοδος IKE

Μια σύνοδος IKE τρέχει στο πρωτόκολλο UDB στην προεπιλεγμένη θύρα 500. Όταν ξεκινήσει η IKE διαπραγμάτευση, το IKE ψάχνει για μια πολιτική IKE η οποία είναι ίδια και για τα δύο μέρη. Κόμβος ο οποίος εκκινεί την διαπραγμάτευση θα στείλει όλες τις ρυθμισμένες πολιτικές στον απομακρυσμένο κόμβο, ο οποίος θα προσπαθήσει να ταιριάξει τις πολιτικές των 2 μελών συγκρίνοντας τις ψηλότερες σε προτεραιότητα πολιτικές του, με τις πολιτικές που λήφθηκαν από τον άλλο κόμβο. Ο απομακρυσμένος κόμβος ελέγχει τις πολιτικές έως ότου βρεθεί μια αντιστοιχία. Η αντιστοιχία γίνετε όταν οι πολιτικές και από τους δύο κόμβους περιέχουν τον ίδιο αλγόριθμο κρυπτογράφησης, κατακερματισμού, αυθεντικοποίησης, και παραμετρικές τιμές Diffie-Hellman, και όταν η πολιτική του απομακρυσμένου κόμβου προσδιορίζει ένα χρόνο εγκυρότητας μικρότερο ή ίσο στη περίοδο εγκυρότητας της πολιτικής τις οποίας γίνετε η σύγκριση. Αν δεν βρεθεί μια αποδεκτή αντιστοιχία, το IKE θα αρνηθεί τη διαπραγμάτευση και το IPsec SA(σχήμα ασφάλειας) δεν θα τύχει διαπραγμάτευσης, ούτε θα εγκατασταθεί. Αν βρεθεί αντιστοιχία το IKE θα ολοκληρώσει τις διαπραγματεύσεις και θα δημιουργήσει μια ασφαλή σύνοδο IKE βασισμένη επάνω στη προσυμφωνηθέντα πολιτική, και θα διαπραγματευτεί τις IPsec συσχετίσεις ασφάλειας πάνω στην ασφαλή σύνοδο της IKE.

3.2.6 Προστασία συνόδου IKE

Η σύνοδος IKE προστατεύετε από κρυπτογραφικούς αλγόριθμους. Το IKE παρέχει αυθεντικοποίηση κόμβου, ακεραιότητα συνόδου και ιδιωτικότητα για τη διαχείριση της συνόδου. Η πολιτική που ακολουθεί το πρωτόκολλο IKE, καθορίζει πως η σύνοδος IKE θα έπρεπε να είναι προστατευμένη και έχει διάφορους παραμέτρους οι οποίοι θα πρέπει να συμφωνηθούν κατά τη διάρκεια της αρχικής διαπραγμάτευσης μεταξύ των κόμβων. Εφόσον κάποια μηνύματα IKE είναι κρυπτογραφημένα και αυθεντικοποιημένα, οι κόμβοι πρέπει να συμφωνήσουν, ένα τρόπο, με τον οποίο θα γίνετε από κοινού η κρυπτογράφηση και η

αυθεντικοποίηση των μηνυμάτων. Για όλες αυτές τις διαπραγματεύσιμες παραμέτρους το IKE καθορίζει τα χαρακτηριστικά και το διάστημα των τιμών που μπορούν να πάρουν. Οι κόμβοι πρέπει να συμφωνούν εξολοκλήρου σε μια δέσμη αλγορίθμων και πρωτοκόλλων για να προστατεύσουν τη σύνοδο. Αυτή η δέσμη που αποτελείτε από την κρυπτογράφηση, αλγόριθμους κατακερματισμού, μέθοδος αυθεντικοποίησης, αλγόριθμος Diffie-Hellman, και σχήμα ασφαλείας IKE με χρονοδιάγραμμα, ονομάζετε πακέτο προστασίας IKE (protection suites).

3.2.7 Φάσεις IKE

Το IKE έχει δύο φάσεις προετοιμασίας, κάθε μία από την οποία μπορεί να υλοποιηθεί με ειδικό τρόπο. Στη πρώτη φάση χρησιμοποιείτε μια κύρια ή μια επιθετική ανταλλαγή λειτουργιών που χρησιμοποιείτε για τη διαπραγμάτευση των σχημάτων ασφαλείας του IKE. Στη δεύτερη φάση γίνετε χρήση μιας γρήγορης διαπραγμάτευσης με σκοπό την ανταλλαγή και τη δημιουργία των σχημάτων ασφαλείας (IPsec SAs).

3.2.7.1 Πρώτη Φάση IKE

Στη εικόνα 14 απεικονίζετε ένα παράδειγμα της πρώτης φάσης του IKE. Παρατηρούμαι ότι τα δύο μέλη θέλουν να επικοινωνήσουν. Για να γίνει αυτό πρέπει από κοινού να συμφωνήσουν σε ένα πακέτο προστασίας IKE. Ο εκκινητής προτείνει διάφορα πακέτα προστασίας στον επικοινωνούντα κόμβο ο οποίος επιλέγει ένα από τα προτεινόμενα πακέτα. Η επιλογή της πολιτικής ασφάλειας πραγματοποιείτε από τον αποδέκτη κόμβο σύμφωνα με τις προτεραιότητες ασφάλειας που έχει προαποφασίσει. Στο παράδειγμα, ο εκκινητής κόμβος προτείνει τρία πακέτα προστασίας και ο δέκτης διαλέγει το δεύτερο βασιζόμενος πάντα στις τοπικές ρυθμίσεις προτεραιοτήτων. Οι κόμβοι πρέπει να συμφωνήσουν με ακρίβεια στο θέμα του πακέτου προστασίας. Αν δεν συμβεί αυτό(να υπάρχουν κοινές πολιτικές), τότε η σύνοδος IKE τερματίζετε.

Εικόνα 14: Πρώτη φάση IKE

3.2.7.2 Δεύτερη φάση IKE

Η δεύτερη φάση του IKE χρησιμοποιείτε για να γίνει η διαπραγμάτευση των υπόλοιπων σχημάτων ασφαλείας των άλλων πρωτοκόλλων (IPsec's AH, IP PCP(payload comparison protocol) και ESP). Ο εκκινητής προτείνει μια λίστα με προτάσεις πολιτικών IPsec στον επικοινωνών κόμβο, ο οποίος επιλέγει μια από τις προτεινόμενες. Η επιλογή της πολιτικής ασφάλειας πραγματοποιείτε από τον αποδέκτη κόμβο σύμφωνα με τις προτεραιότητες ασφάλειας που έχει προαποφασίσει. Όταν η πολιτική συμφωνηθεί, επιτυγχάνετε η συμφωνία του κρυπτογραφικού υλικού και τα IPsec SAs εγκαθίστανται. Στην εικόνα 15, ο εκκινητής προτείνει δύο πολιτικές για τη ασφαλή διακίνηση των πληροφοριών στο ανασφαλές δίκτυο, και ο επικοινωνών κόμβος επιλέγει την δεύτερη επιλογή. Μετά την επιτυχή διαπραγμάτευση το κρυπτογραφικό υλικό ανταλλάσσετε και επιτυγχάνετε η ασφαλή διακίνηση των πληροφοριών.

Εικόνα 15: Δεύτερη φάση IKE

3.2.8 Επιπρόσθετη ασφάλεια

Ένας από του πιο σημαντικούς παράγοντες της διαπραγμάτευσης των IKE SAs είναι η αμοιβαία αυθεντικοποίηση των κόμβων. Κάθε κόμβος πρέπει να είναι σίγουρος ότι απευθύνετε στο σωστό κόμβο, πριν διαπραγματευτεί τις πολιτικές προστασίας διακίνησης δεδομένων μαζί του. Αυτή η αμοιβαία αυθεντικοποίηση πραγματοποιείτε με μεθόδους αμφίδρομης αυθεντικοποίησης. Το IKE παρέχει τρεις μεθόδους για αμφίδρομη αυθεντικοποίηση:

- Αυθεντικοποίηση με χρήση προαποφασισμένου κλειδιού
- Αυθεντικοποίηση με χρήση κρυπτογραφημένων στοιχείων RSA
- Αυθεντικοποίηση με χρήση υπογραφών RSA

Εκτός από την αυθεντικοποίηση των κόμβων, το IKE παρέχει και διαφυλάσσει επίσης την ακεραιότητα και την ιδιωτικότητα στη σύνοδο IKE. Η σύνοδος μπορεί να κρυπτογραφηθεί με τη χρήση του DES ή του 3DES αλγόριθμου. Τα κλειδιά για την κρυπτογράφηση δημιουργούνται και προέρχονται από την αρχική ανταλλαγή Diffie-Hellman η οποία προκύπτει μεταξύ των κόμβων στην αρχή της IKE συνόδου. Όταν το IKE διαπραγματεύεται στην κύρια λειτουργία του, η ταυτότητα των κόμβων κρυπτογραφείτε, ενώ στην επιθετική λειτουργία του IKE οι ταυτότητες δεν υποκρύπτονται.

Επίσης το IKE χρησιμοποιεί τις λειτουργίες HMAC[60] (**H**ash-Based **M**essage **A**uthentication **C**ode) για να διαφυλάξει την ακεραιότητα μιας συνόδου IKE. Συνήθως είναι διαθέσιμη η επιλογή μεταξύ keyed SHA-1 και MD5. Αυτά τα

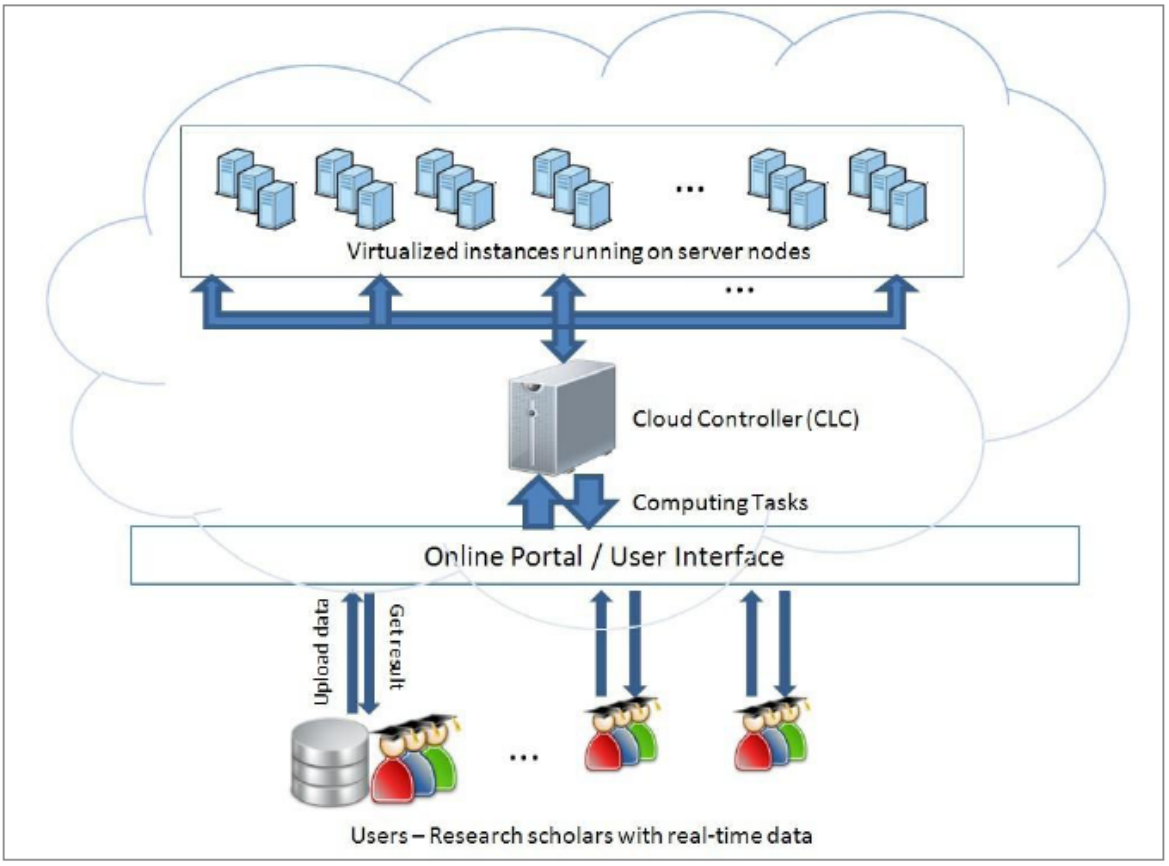
κρυπτογραφικά κλειδιά για τις λειτουργίες του IMAC δημιουργούνται εκ των προτέρων από την αρχική ανταλλαγή Diffie-Hellman.

Το IKE αρχικά διενεργεί μια εφήμερη ανταλλαγή Diffie-Hellman. Για την ανταλλαγή, οι κόμβοι μοιράζονται κρυπτογραφικά κλειδιά, τα οποία θα χρησιμοποιηθούν αργότερα για την κρυπτογράφηση και την ακεραιότητα των λειτουργιών. Η δύναμη των κλειδιών μπορεί να τροποποιηθεί, για την ταχύτερη ή ασφαλέστερη επικοινωνία, από τον χρήστη, επιλέγοντας ένα μικρότερου μεγέθους κλειδί για τις ανταλλαγές Diffie-Hellman. Όταν η διάρκεια ζωής μιας IKE συνόδου λήξει, μια νέα ανταλλαγή Diffie-Hellman πραγματοποιείτε μεταξύ των κόμβων και το σχήμα ασφαλείας IKE ξαναδημιουργείτε.

3.3 Cloud Computing Background Key Exchange (CCBKE)

Το IKE είναι αποδεκτά αποτελεσματικό για τη χρήση του στο νέφος για την ανταλλαγή κλειδιών, αλλά έρευνες έδειξαν ότι, το σχήμα IKE πάσχει από χαμηλή αποδοτικότητα λόγω των χαμηλών επιδόσεων του στις κρυπτογραφικές λειτουργίες ασύμμετρου κλειδιού, πάνω σε μεγάλου όγκου δεδομένων και στις πράξεις υψηλής πυκνότητας. Αυτά ακριβώς τα χαρακτηριστικά διακατέχουν τις επιστημονικές εφαρμογές. Επομένως το CCBKE προτάθηκε για την επίλυση αυτού του προβλήματος, των επιστημονικών εφαρμογών που σε έργα όπως το γιγάντιο τηλεσκόπιο εξήντα και τεσσάρων μέτρων που χρησιμοποιούν οι αυστραλοί αστροφυσικοί [61] το IKE έρχεται αντιμέτωπο, με μια συνεχόμενη παραγωγή δεδομένων, γύρο στο 1GB το δευτερόλεπτο. Σε πολύ λίγα λεπτά το σύνολο τον δεδομένων που θα τύχουν επεξεργασίας είναι τεράστιο σε σχέση με τον χρόνο δημιουργίας τους. Από αυτό το παράδειγμα μπορούμε να συμπεράνουμε ότι, λόγω της εντατικής χρήσης των δεδομένων, η ανάπτυξη των επιστημονικών εφαρμογών στο νέφος θα επιτελέσει μεγάλη καινοτομία στη μείωση των κοστών. Επίσης τα δεδομένα που παράγονται από τέτοιες επιστημονικές έρευνες, είναι ιδιοκτησία διανοούμενων που μπορεί να οδηγήσουν σε μια σημαντική επιστημονική ανακάλυψη, ως εκ τούτου, η αξία των δεδομένων μπορεί να καθίστανται ανεκτίμητη. Για αυτό, η ασφάλεια των δεδομένων, και συγκεκριμένα η εμπιστευτικότητα και η ακεραιότητα των δεδομένων τυγχάνει εξαιρετικής σημασίας. Ένα άλλο σχετικό πρόβλημα είναι ότι, οι επιστήμονες συνήθως χρειάζονται πρόσβαση στα δεδομένα τους το συντομότερο δυνατών. Επομένως ένα αργοπορημένο αποτέλεσμα μιας έρευνας, μπορεί να επιφέρει μεγάλη χρηματική σπατάλη ή μια απώλεια επιστημονικής ανακάλυψης. Έχοντας αυτά τα δεδομένα

σαν κίνητρα, μια επιστημονική ομάδα ανέπτυξε το CCBKE στοχεύοντας στον αποτελεσματικό προγραμματισμό της ασφάλειας των επιστημονικών εφαρμογών. Το σχήμα του CCBKE είναι βασισμένο στη στρατηγική της τυχαίας επαναχρησιμοποίησης και στο σχήμα του πρωτοκόλλου IKE. Στην εικόνα 16 απεικονίζεται η βασική δομή ενός συστήματος νέφους για επιστημονικές εφαρμογές, για να έχουμε πιο παραστατική αντίληψη.



Εικόνα 16: Βασική δομή τυπικού συστήματος νέφους για επιστημονικές εφαρμογές

Σε μια τυπική υποδομή συστήματος νέφους ένας κεντρικός server χρησιμοποιείτε για τη λήψη και επεξεργασία των αιτημάτων του χρήστη, αλλά και για το προγραμματισμό και διαμοιρασμό των διεργασιών μέσα από το MapReduce[62]. Αυτός ο server ονομάζετε και Cloud Controller (CLC). Τα μέρη (instances) του server που τρέχουν στα clusters, είναι υπεύθυνα για την επεξεργασία των διαμοιρασμένων διεργασιών σε ένα παράλληλο σύστημα, και για την επιστροφή των αποτελεσμάτων. Επίσης το CLC συναρμολογήσει τα αποτελέσματα και να τα επιστρέψει στον χρήστη.

3.3.1 Αυθεντικοποιημένη ανταλλαγή κλειδιού βασισμένη στη τυχαία επαναχρησιμοποίηση

Παρόμοια με το IKE, το σχήμα CCBKE περιέχει τρία ανεξάρτητα συστατικά, τη ρύθμιση του συστήματος, την αρχική ανταλλαγή και την επανέκδοση κλειδιού.

3.3.1.1 Συμβολισμοί

S:	Κατάσταση (μέρη) του διακομιστή
S_i:	υπ αριθμών ***i*** διακομιστή στη κατάσταση S
C:	ελεγκτής νέφους (Cloud Controller)
HDR_i:	κεφαλίδα που περιέχει παραμέτρους ασφαλείας
$CertReq_i$:	αίτηση πιστοποιητικού
$Cert_i$:	πιστοποιητικό
N_i:	χρήση αριθμού υπ αριθμόν ί
SA:	διαπραγμάτευση ασφάλειας, χρήση στην διαπραγμάτευση κρυπτογραφικών αλγορίθμων
ID_i:	πληροφορίες ταυτότητας
Sig_i:	υπογραφή, μπορεί να επαληθευτεί με τη χρήση προκαθορισμένων αλγορίθμων και με το δημόσιο κλειδί μέσα στο πιστοποιητικό
prf():	ψευδοτυχαία λειτουργία
$\{M\}_k$:	κρυπτογραφημένο μήνυμα ***M*** με κλειδί συνόδου το ***k***

3.3.1.2 Ρύθμισης του συστήματος **CCBKE**

Το σύστημα επιλέγει ένα μεγάλο περιττό ακέραιο αριθμό *p,* για να υλοποιήσει μια Diffie-Hellman ανταλλαγή, και μια γεννήτρια *g* από το σύνολο Z_p^*, όπου το *g* είναι μια πρωταρχική ρίζα του modulo *p*. Συνήθως το *p* είναι ένας Sophie Germain prime αριθμός, όπου (p-1)/2 είναι επίσης πρωταρχικός αριθμός, έτσι ώστε το σύνολο Z_p^* μεγιστοποιεί την ανθεκτικότητα του εναντίον της επίθεσης «Τετραγωνικής ρίζας» [63] σε πρόβλημα διακριτού λογαρίθμου. Μια αρχή έκδοσης πιστοποιητικών (CA) όπως και στο PKI εξακολουθεί να είναι αναγκαία στο πλαίσιο της ασφάλειας, ούτως ώστε τα επικοινωνούντα μέρη να μπορούν να ταυτοποιήσουν ο ένας τον άλλο μέσο ανταλλαγής επαληθεύσιμων πιστοποιητικών $Cert_c$ και $Certs_i$, αφού σε αυτά τα πιστοποιητικά εμπεριέχονται τα δημόσια κλειδιά τους, που μπορούν να χρησιμοποιηθούν για εξακρίβωση των εταίρων συνόδου. Τα

πιστοποιητικά, είναι σχετικά μακροπρόθεσμα δεδομένα που έχουν εκδοθεί σε όλους τους συμμετέχοντες της επικοινωνίας πριν από την έναρξη της επικοινωνίας, και η αρχή πιστοποίησης δεν θα συμμετάσχει εκτός αν επαληθευτούν εκ νέου οι ταυτότητες, ανακληθούν και εκδοθούν εκ νέου τα πιστοποιητικά των συμμετεχόντων που είναι απαραίτητα.

3.3.1.3 Αρχική ανταλλαγή CCBKE

Η αρχική ανταλλαγή υλοποιείτε όταν πρέπει να εκτελεστεί μια καινούρια εργασία, επειδή τότε είναι όταν ο CLC(cloud controller), θα πρέπει να αποφασίσει, πως θα διανέμει την καινούρια εργασία που θα εκτελεστεί και σε πια υπάρχουσα υπολογιστική υποδομή. Ο CLC επιλέγει μια μυστική τιμή $\chi < p$, υπολογίζει το δημόσιο του κρυπτογραφικό υλικό g^χ από το Z_p, και μεταδίδει το ακόλουθο μήνυμα για τον τομέα των καταστάσεων του διακομιστή S που περιέχει N περιπτώσεις $S_1,S_2,\ldots,S_n$:

1η επανάληψη, C→ S: HDR_{Si} , SA_{Si1} , g^x , N_c

Όπου το HDR και το SA χρησιμοποιούνται για την διαπραγμάτευση του αλγόριθμου, το g^x για την ανταλλαγή κλειδιού Diffie-Hellman, και το N για την επαλήθευση φρεσκάδας. Ο εκκινητής ενός συνηθισμένου σχήματος IKE θα παράγει *n* μυστικές τιμές $\chi_1,\chi_2,\ldots,\chi_n$, ακολούθως θα υπολογίσει και θα στείλει το g^{x_1}, $g^{x_2},\ldots$, g^{x_n}, είτε μέσο εκπομπής, ή αποστολής ένας προς έναν, για να δημιουργήσει κανάλια ασφαλούς επικοινωνίας με τον κάθε αποδέκτη ξεχωριστά. Στο CCBKE παρόλο που καθιερώνετε ένα SA για κάθε κατάσταση του διακομιστή S_i όπου το i=1,2,3,…,n, χρησιμοποιείτε μόνο μια μοναδική μυστική τιμή *χ* για το CLC σε όλα τα n μηνύματα για να μειωθεί το κόστος.

Μετά την παραλαβή του πρώτου μηνύματος, κάθε κατάσταση του διακομιστή δημιουργεί τι μυστική του τιμή $y_i < p$, υπολογίζει το κρυπτογραφικό υλικό g^{y_i} και ανταποκρίνεται με τη 2η επανάληψη ως εξής

2η επανάληψη, S → C: HDR_{Si} , SA_{Si1} , g^{y_i} , Ns_i , $CertReq_i$, for i=1,…,n

Η 2η επανάληψη περιλαμβάνει n διαφορετικά μηνύματα σταλμένα ξεχωριστά από $S_1,S_2,\ldots,S_n$. Μετά την ανταλλαγή των 2 πρώτον επαναλήψεων, τα κλειδιά συνόδου $g^{(x*y_1)},\ldots, g^{(x*y_n)}$, υπολογίζονται για όλα τα μέρη ως ακολούθως:

$$C : g^{(x*y_1)} = (g^{y_1})^x,\ldots, g^{(x*y_n)} = (g^{y_n})^x$$

$S_1 : g^{(x*y_1)} = (g^x)^{y_1}$

. . .

$S_1 : g^{(x*y_n)} = (g^x)^{y_n}$

Τα κλειδιά συνόδου έχουν τώρα ανταλλαχθεί μεταξύ του CLC και της κάθε κατάστασης του διακομιστή για τη χρήση αυτών στην κρυπτογραφία σε μεταγενέστερη επικοινωνία. Παρόλο που η ανταλλαγή κλειδιού Diffie-Hellman έχει ολοκληρωθεί, η CCBKE αρχική ανταλλαγή δεν έχει ολοκληρωθεί αφού οι εμπλεκόμενοι πρέπει να αυθεντικοποιηθούν ο ένας στον άλλον για να αποτρέψουν την επίθεση man-in-the-middle. Παρόμοια όπως και στο IKE, ο CLC παράγει υπογραφές Sig_{Ci} που είναι οι υπογραφές των *n* μηνυμάτων, κάνοντας χρήση του μυστικού κλειδιού από το ζεύγος κλειδιών που εκδόθηκε από το CA:

$$M_{Ci} = prf(prf(N_C||N_{Si}||g^{(x*y_i)} \,||\, g^x \,||\, g^{y_i} \,||\, SA_c \,||\, ID_c) \text{ for } i=1,\ldots,n$$

Και μεταδίδει το ακόλουθο μήνυμα στο S :

$3^{η}$ επανάληψη, C → S :

$HDR_{Si}, \{ID_c, SA_{c2}, Cert_c, CertReq_{s1}, Sig_c\}_{g^{(X*Y1)}}||\{ ID_c, SA_{c2}, CertReq_{s2}, Sig_c\}_{g^{(X*Y2)}}$
$||\ldots||\{ID_c, SA_{c2}, Cert_c, CertReq_{sn}, Sig_c\}_{g^{(X*Yn)}}$, for i=1,…,n

Τα μέρη του διακομιστή μπορούν τώρα να αυθεντικοποιήσουν τον εκκινητή του διαλόγου κάνοντας χρήση το κλειδί της συνόδου του g^(X*Yί) για να αποκρυπτογραφήσουν το δικό του μέρος από αυτό το μήνυμα. Οι υπογραφές μπορούν να ελεγχθούν από το δημόσιο κλειδί που εμπεριέχετε μέσα σε αυτό το πιστοποιητικό. Ομοίως τα μέρη του διακομιστή θα αποστείλουν την δική τους κρυπτογραφημένη ταυτότητα (ID), υπογραφή και πιστοποιητικό στον CLC για να ελεγχθούν:

$4^{η}$ επανάληψη, S → C: $HDR_{Si}, \{ID_C, SA_{Ci2}, Cert_{si}, Sig_{Si}\}_{g^{(X*Yi)}}$, for i=1,…, n

Και αυτή η επανάληψη περιλαμβάνει n μηνύματα. Αφού αυθεντικοποιηθούν οι ταυτότητες των CLC και των μερών του διακομιστή μέσα από την $3^{η}$ και $4^{η}$ επανάληψη, ο CLC θα στείλει από το *S1* μέχρι το *Sn* τα δεδομένα των μοιρασμένων διεργασιών τα οποία είναι κρυπτογραφημένα με τα κλειδιά συνόδου

g^(X*Y1) ,…, g^(X*Yn) κάνοντας χρήση συμμετρικής κρυπτογραφίας π.χ AES. Όταν τελειώσει η εκτέλεση των διεργασιών, τα *S1,...,Sn* μέρη του διακομιστή επιστρέφουν το αποτέλεσμα στον CLC τα οποία είναι επίσης κρυπτογραφημένα με τα g^(X*Y1),…, g^(X*Yn). Η λειτουργία prf() υλοποιείτε συχνά σαν μια λειτουργία HMAC[60], όπου τα εξερχόμενα δεδομένα είναι προκαθορισμένου μεγέθους (συνήθως 128bits) και έχουν υψηλή απόδοση, περίπου στα 200MB/s βάση των σημερινών αποδόσεων ενός υπολογιστή γραφείου.

3.3.1.4 Επαναδιαπραγμάτευση κλειδιού στο CCBKE

Η επαναδιαπραγμάτευση κλειδιού επιτυγχάνετε συνήθως επαναλαμβάνοντας την αρχική ανταλλαγή. Ωστόσο, στις ακόλουθες περιπτώσεις, χρειάζονται να εφαρμοστούν εναλλακτικές στρατηγικές.

3.3.1.4.1 Αποτυχία ανάκτησης κλειδιού

Αν κάποιο μήνυμα, που είναι απαραίτητο για την αρχική ανταλλαγή, αποτύχει να φτάσει στον προορισμό του, ο CLC θα ξεκινήσει μια έναν-προς-έναν ανταλλαγή κλειδιού συνόδου IKE με το συγκεκριμένο μέρος του διακομιστή. Δεδομένου ότι αυτό είναι μια περίπτωση μεμονωμένου λάθους μπορεί να αντιμετωπιστεί στην πορεία, καθιστώντας αυτή τη πρόσθετη κατανάλωση χρόνου σαν αμελητέα.

3.3.1.4.2 Διεργασίες πολλαπλών σταδίων

Σε μια διεργασία πολλαπλών σταδίων, τα δεδομένα μεταφέρονται μπροστά και πίσω. Σε αυτή την κατάσταση δεν είναι απαραίτητο για τους συμμετέχοντες να αυθεντικοποιούνται ξανά ο ένας στον άλλον, μετά από την επιτυχή αυθεντικοποίηση στη πρώτη επανάληψη, λόγω της υψηλής εξάρτισης των δεδομένων στην ίδια διεργασία. Επομένως μόνο η $1^{η}$ και $2^{η}$ επανάληψη χρειάζεται να εκτελεστούν, με νέα κλειδιά και μικρές αλλαγές στα πεδία SA και HDR. Έχοντας υπόψη τη περίπτωση των επιστημονικών εφαρμογών με τα μεγάλου όγκου δεδομένα, στη $3^{η}$ και $4^{η}$ επανάληψη περιέχονται μόνο γρήγορες λειτουργίες όπως η αυθεντικοποίηση της υπογραφής μέσο μηνύματος μικρού μήκους, και κρυπτογραφία συμμετρικού κλειδιού, η υπολογιστική επιβάρυνση της επαναδιαπραγμάτευσης των κλειδιών συνόδου, στον CLC είναι σχεδόν ταυτόσημη, από άποψη αποτελεσματικότητας, με την αρχική ανταλλαγή.

3.3.2 Ανάλυση επιπέδου ασφάλειας στο CCBKE

Θα αναλύσουμε την ασφάλεια του συστήματος με δύο τρόπους. Θα αποδειχτεί ότι το σχήμα *CCBKE* είναι ασφαλής ενάντια σε εσωτερικούς όσο και σε εξωτερικούς επιτιθέμενους διατηρώντας τέλεια την μυστικότητα.

3.3.2.1 Απόδειξη ασφάλειας

Οι ακόλουθες συζητήσεις στηρίζονται σε δύο πρότυπα κρυπτογραφικών παραδοχών, και έχουν ως εξής:

1η υπόθεση: Οποιοσδήποτε εμπλεκόμενος στο σύστημα δεν μπορεί να ανακτήσει δεδομένα τα οποία έτυχαν κρυπτογραφίας με κάποιο αλγόριθμο συμμετρικού κλειδιού εκτός αν έχει στη κατοχή του το κλειδί συνόδου το οποίο χρησιμοποιήθηκε για την κρυπτογράφηση των δεδομένων στο αρχικό στάδιο.

2η υπόθεση: Δεδομένης μιας κυκλικής ομάδας *G* της σειράς q, μια γεννήτρια *g* της *G* και δύο τυχαίοι ακέραιοι αριθμοί a,b E {0,…,(q-1)}, η ανάκτηση του g^(a*b) σε πολυωνυμικό χρόνο χρησιμοποιώντας μόνο το {g, g^a, g^b} είναι υπολογιστικά αδύνατη.

Παρόμοια με τις περισσότερες αναλύσεις ασφάλειας σε πρωτόκολλα επικοινωνίας δημοσίου κλειδιού, θα καθορίσουμε τις δυνατότητες ενός εσωτερικού και ενός εκτός δικτύου επιτιθεμένου.

Ορισμός 1 (επιτιθέμενος εκτός νέφους): ένας κακόβουλος επιτιθέμενος εκτός δικτύου M_0 είναι ένας αντίπαλος που είναι ικανός στην παρακολούθηση, υποκλοπή και αλλοίωση της επικοινωνιακής κυκλοφορίας των δεδομένων σε ολόκληρο το φόντο της δομής του νέφους, προκειμένου να αποκτήσει πρόσβαση σε προστατευμένα δεδομένα στη μετάδοση. Ωστόσο, ο M_0 δεν έχει πρόσβαση σε καμία από τις μηχανές του κόμβου και η ταυτότητα του δεν είναι νόμιμη, π.χ ο M_0 δεν έχει δικαίωμα να αποκτήσει ένα έγκυρο πιστοποιητικό για να μπορέσει να αυθεντικοποιηθεί από τον CLC ή οποιοδήποτε μέρος του διακομιστή.

Ορισμός 2 (επιτιθέμενος εντός νέφους): ένας κακόβουλος επιτιθέμενος εντός δικτύου M_i είναι ένας αντίπαλος που είναι ικανός να αυθεντικοποιείται από τον CLC και να ενεργεί σαν ένα νόμιμο μέρος του διακομιστή της επικοινωνίας. Ωστόσο, ο M_i δεν έχει νόμιμη πρόσβαση σε κάποιο άλλο νόμιμο συμμετέχοντα, συμπεριλαμβανομένου και τα μέρη του διακομιστή, και τις ιδιωτικές πληροφορίες του CLC.

Ακολουθεί απόδειξη ότι το σχήμα *CCBKE* είναι ασφαλής ενάντια σε τέτοιου είδους επιτιθέμενους.

Θεώρημα 1: ένας επιτιθέμενος εκτός νέφους M_0 δεν μπορεί να ανακτήσει σε πολυωνυμικό χρόνο οποιαδήποτε ανταλλαγή κλειδιού συνόδου $g^{(x*y_i)}$ στο CCBKE.

Απόδειξη: ακολουθώντας τον ορισμό 1, ξέρουμε ότι ένας επιτιθέμενος εκτός νέφους M_0 μπορεί να έχει πρόσβαση σε όλα τα δημόσια κρυπτογραφικά κλειδιά $g, p, g^x, g^{y_1},\ldots, g^{y_n}$ παρακολουθώντας ολόκληρο το δίκτυο, αλλά δεν μπορεί να έχει πρόσβαση στις μυστικές πληροφορίες όπως είναι τα $x,y_1,\ldots,y_n$. Λαμβάνοντας υπόψη το επίπεδο δυσκολίας του υπολογισμού του προβλήματος Diffie-Hellman, γνωρίζουμε ότι ο M_0 δεν μπορεί να υπολογίσει κανένα από τα $g^{(x*y_i)}$ όπου το $i=1,\ldots,n$, σε πολυωνυμικό χρόνο, έχοντας στη κατοχή του τα $g^x, g^{y_1},\ldots, g^{y_n}$.

Θεώρημα 2: Υποθέτουμε ότι $\kappa=g^{(x*y_m)}$ είναι το κλειδί συνόδου που διαπραγματεύονται ένας επιτιθέμενος εντός δικτύου M_i με τον CLC. Ο M_i αδυνατεί να ανακτήσει σε πολυωνυμικό χρόνο οποιαδήποτε κλειδιού συνόδου $g^{(x*y_i)}$ εκτός από το *κ*.

Απόδειξη: Σύμφωνα με το θεώρημα 2, ο M_i έχει τη δική του μυστική τιμή y_t, προσθέτουμε και το *κ* που προήλθε από τις πληροφορίες που ελέγχονται από τον M_0. Εφόσον όλες οι μυστικές τιμές $y_1,\ldots,y_n$ παράγονται από μεμονωμένα μέρη του διακομιστή και όχι κατανεμημένα, υπάρχει μια πιθανότητα να παραχθεί ένα ίδιο μυστικό κλειδί και να χρησιμοποιηθεί για ανταλλαγή κλειδιού από διαφορετικά μέρη του διακομιστή, αυτό το συμβάν ονομάζετε '*σύγκρουση'*. Όταν υπάρξει '*σύγκρουση' μεταξύ του* M_i και ενός νόμιμου μέρους του διακομιστή S_i(με την ιδιωτική του τιμή Y_i), έχουμε $Y_i=Y_m$. Αυτό σημαίνει ότι ο M_i μπορεί να ανακτήσει τα δεδομένα που στέλνει ο S_i χρησιμοποιώντας το δικό του κλειδί $g^{(x*y_m)}$. Επειδή τα $y_1,\ldots,y_n$ επιλέγονται τυχαία από το Z^*_p, η πιθανότητα E να υπάρξει σύγκρουση είναι:

$$\epsilon = \prod_{i=1}^{n-1}\left(1-\frac{i}{p}\right) \approx \prod_{i=1}^{n-1} e^{-\frac{i}{p}} = e^{-\frac{n(n-1)}{2p}}$$

Στο *CCBKE* το *p* είναι 1024-bit και το *n* που αντιπροσωπεύει τα διάφορα μέρει του διακομιστή είναι συνήθως μερικές χιλιάδες, καθιστώντας τη πιθανότητα τις σύγκρουσης αμελητέα. Αυτό σημαίνει ότι ένας επιτιθέμενος από το εσωτερικό του δικτύου δεν μπορεί να ανακτήσει κανενός άλλου το κλειδί εισόδου εκτός από

μια αμελητέα πιθανότητα. Συνδυάζοντας αυτό το συμπέρασμα με το **θεώρημα 1**, έχουμε αποδείξει το **Θεώρημα 2**.

Δεδομένου ότι όλες οι πληροφορίες ταυτοποίησης κρυπτογραφούνται με τα κλειδιά συνόδου στις επαναλήψεις ταυτοποίησης 3 και 4 και οι επιτιθέμενοι δεν μπορούν να ανακτήσουν κανενός τα κλειδιά συνόδου, καταλήγουμε στο εξής συμπέρασμα:

Κάθε δήθεν συμμετέχων θα αποτύχει στις επαναλήψεις ταυτοποίησης 3 και 4.

Μετά την αποτυχία τις αυθεντικοποίησης στις επαναλήψεις 3 και 4, η επικοινωνία μεταξύ των συμμετεχόντων θα τερματιστεί. Μια Man-in-the-middle επίθεση, ή οποιαδήποτε άλλη επίθεση πλαστοπροσωπίας στο σχήμα *CCBKE* δεν μπορεί να επιτευχθεί.

3.3.2.2 Επιπρόσθετη ασφάλεια

Όπως και στο IKE, τα κλειδιά συνόδου που χρησιμοποιήθηκαν στη κρυπτογράφηση της επικοινωνίας, χρησιμοποιούνται μόνο μια φορά μέχρι τη λήξη και καταστροφή τους. Έτσι, εάν ένα κλειδί συνόδου που έχει ήδη χρησιμοποιηθεί και έχει εκτεθεί, με κάποιο τρόπο, σε ένα κακόβουλο αντίπαλο, είναι παντελώς άχρηστο. Αυτό είναι ένα από τα μεγαλύτερα πλεονεκτήματα της χρήσης ανταλλαγής κλειδιού σε ένα υβριδικό σύστημα κρυπτογραφίας, για αυτό το λόγω δεν επιλέχτηκε η απλή κρυπτογράφηση ενός κλειδιού συνόδου με ένα αλγόριθμο κρυπτογραφίας ασύμμετρου κλειδιού, παρόλο που αυτό θα είχε εύκολη υλοποίηση μέσο του PKI λαμβάνοντας υπόψη το γεγονός ότι υιοθετήθηκε ένα CA στην αρχιτεκτονική του *CCBKE.*

3.4 Αξιολόγηση τεχνικών διανομής κλειδιών στο cloud

Στις πιο πάνω ενότητες του κεφαλαίου 3 τρία, αναφέρθηκαν τρις τεχνικές διανομής κλειδιών, και αναλύθηκαν επεξηγηματικά οι λειτουργίες και τα βήματα που διενεργούνται για τη υλοποίηση τους. Η τεχνική του *PKI* παρέχει ένα καλό επίπεδο ασφάλειας, αλλά επειδή για να λειτουργήσει είναι απαραίτητη μια αρχή πιστοποίησης, δηλαδή ένα τρίτο έμπιστο μέλος, που πρέπει να ελέγχει αυτοπροσώπως τους συμμετέχοντες πριν τη διανομή των κλειδιών, καθίσταται χρονοβόρα και μη λειτουργική για κάποιες περιπτώσεις. Η τεχνική διανομής του *IKE* παρέχει ένα αποδεκτό επίπεδο ασφάλειας και ευχρηστίας στη διανομή κλειδιών για αυτό και καθιερώθηκε στο διαδίκτυο, αλλά όταν έρχεται αντιμέτωπη με

μεγάλου όγκου δεδομένα, είναι αργή και δεν μπορεί να εξυπηρετήσει κάποιες περιπτώσεις. Το σχήμα *CCBKE* παρέχει εξίσου υψηλό επίπεδο ασφάλειας με το IKE αλλά σε περιπτώσεις με μεγάλου όγκου δεδομένων τείνει να έχει πολύ καλύτερη απόδοση από το IKE.

Αλγόριθμοι	**Πλεονεκτήματα**	**Μειονεκτήματα**	**Κατάλληλο για cloud**
PKI	Αποδεκτό επίπεδο ασφάλεια	Δύσκολη υλοποίηση λόγω των πιστοποιητικών που πρέπει να ενημερώνονται και να ανταλλάζονται συχνά, μη πρακτικό για κάποιες περιπτώσεις.	**ΜΗ ΚΑΤΑΛΛΗΛΟ**
IKE	Πάρα πολύ καλό επίπεδο ασφάλειας.	Υπάρχει πρόβλημα όταν έρχεται αντιμέτωπη με τεραστίων όγκου δεδομένα	**ΚΑΤΑΛΛΗΛΟ**
CCBKE	Πάρα πολύ καλό επίπεδο ασφάλειας. Προορίζετε για χρήση σε επιστημονικές εφαρμογές λόγω της δυνατότητας γρήγορης κρυπτογράφησης σε μεγάλου όγκου δεδομένων.		**ΚΑΤΑΛΛΗΛΟ**

ΚΕΦΑΛΑΙΟ 4. ΑΞΙΟΛΟΓΗΣΗ ΠΕΙΡΑΜΑΤΩΝ ΣΕ ΔΙΑΝΟΜΗ ΚΛΕΙΔΙΩΝ ΣΤΟ CLOUD

Εισαγωγή

Για τη τεκμηρίωση των υπολογισμών που έγιναν για τη διανομή κλειδιού, υλοποιήθηκαν κάποια πειράματα για να αποδειχτούν οι υποδείξεις των υποθέσεων. Σε αυτό το κεφάλαιο θα περιγραφεί ένα πείραμα το οποίο καταλήγει σε βάσιμα, τεκμηριωμένα συμπεράσματα ότι το *CCBKE* υπερέχει του *IKE*.

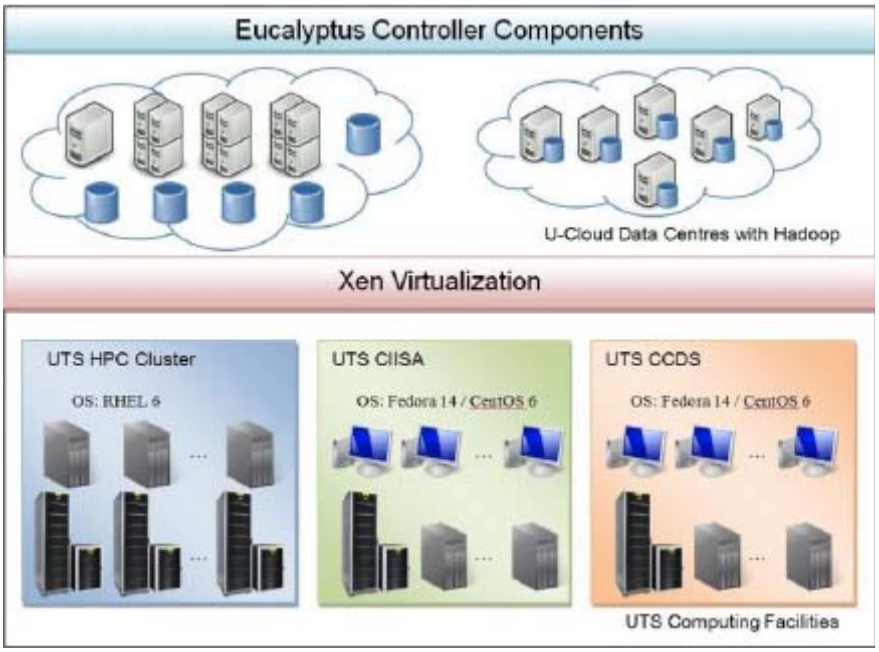
Το σχήμα του *CCBKE* που προτείνετε σε αυτή την έρευνα είναι γενικό. Μπορεί να αναπτυχθεί με ποικίλες επιστημονικές εφαρμογές και σε διάφορες πλατφόρμες υπολογιστικής νέφους για την βελτίωση της αποδοτικότητας και τη μείωση του κόστους της εγγύησης, για την ασφάλεια της επικοινωνίας.

Σύμφωνα με την ανάλυση του προβλήματος που αναφέραμε στο υποκεφάλαιο 3.3, η κατανάλωση χρόνου για της διαδικασίες που υλοποιούνται εκτός CLC μπορούν να θεωρηθούν αμελητέες. Στο *CCBKE* οι επαναλήψεις ***1*** και ***3*** διεκπεραιώνονται εντός του CLC, περιέχοντας και οι δύο ***n*** σπονδυλωτές εκθετικές λειτουργίες. Αφότου ο CLC χρησιμοποίησε μόνο μια τυχαία μεταβλητή ***χ*** αντί για $\{\chi_1,\ldots,\chi_n\}$ για να διεξάγει την ανταλλαγή *Diffie-Hellma,* με τα $\{S_1,\ldots,S_n\}$, υπάρχει μόνο μία λειτουργία αντί για ***n*** σπονδυλωτές εκθετικές λειτουργίες. Ως εκ τούτου, περίπου οι μισές λειτουργίες αποκόπτονται. Από αυτή την άποψη το σχήμα CCBKE μπορεί να έχει τεράστια πλεονεκτήματα απόδοσης έναντι του IKE. Στο επόμενο υποκεφάλαιο θα περιγραφεί ένα πείραμα το οποίο έχει σκοπό να αποδείξει ότι, όταν το CCBKE εφαρμοστεί στο cloud, μπορεί πράγματι να βελτιώσει σημαντικά την αποτελεσματικότητα του IKE.

4.1 Το περιβάλλον του πειράματος

Το U-Cloud είναι ένα σύστημα cloud computing που βρίσκετε στο University of technology, στο Sydney της Αυστραλίας πάνω στο οποίο χτίστηκε και διεξάχθηκε το πειραματικό περιβάλλον. Οι υπολογιστικές εγκαταστάσεις αυτού του συστήματος βρίσκονται σε διάφορα εργαστήρια στο Τμήμα Μηχανικών UTS και της πληροφορικής. Πάνω από το υλικό του υπολογιστή, και του λειτουργικού συστήματος Linux, εγκαταστάθηκε το Xen Hypervisor [64] το οποίο εικονοποιεί την υποδομή και της επιτρέπει να παρέχει ενοποιημένους υπολογιστικούς και

αποθηκευτικούς πόρους. Πάνω στα εικονικά κέντρα δεδομένων, εγκαταστάθηκε το Hadoop[65] για τη διευκόλυνση του προγραμματισμού MapReduce[66] και της κατανεμημένης διαχείρισης των δεδομένων. Επιπλέον εγκαταστάθηκε η πλατφόρμα νέφους ανοικτού κώδικα Eucalyptus [67] η οποία είναι υπεύθυνη για τη γενική διαχείριση, το προγραμματισμό των πόρων, τη διανομή των διεργασιών, και τη αλληλεπίδραση με τους χρήστες. Η δομή του συστήματος U-Cloud απεικονίζεται στην εικόνα 17.



Εικόνα 17 Δομή συστήματος U-Cloud

4.2 Η διαδικασία του πειράματος

Είναι σαφές ότι η πραγματική βελτίωση της απόδοσης που προήλθε από το σχήμα CCBKE εξαρτάται σε μεγάλο βαθμό από το μέγεθος του συνόλου δεδομένων και τον αριθμό των server instances. Τα πειράματα διεξήχθησαν σε πολλαπλά σύνολα δεδομένων εμπλέκοντας κάθε φορά διαφορετικό αριθμό περιπτώσεων του διακομιστή. Τα πειράματα αυτά, θα επιδειχθούν στη επόμενη ενότητα.

Θα συγκριθεί το σχήμα CCBKE με ένα άλλο σχήμα, όπου οι ανταλλαγές κλειδιού του CLC με κάθε μέρος του διακομιστή υλοποιούνται με τη χρήση του πρωτοκόλλου IKE βάση ενός διαχωρισμένου τρόπου. Τα κρυπτοσυστήματα που κάνουν χρήση ασύμμετρου κλειδιού, είναι περίπου χίλιες φορές πιο αργά από τα συστήματα που χρησιμοποιούν συμμετρικά κλειδιά. Για αυτό τα μεγάλα σε όγκο δεδομένα δεν κρυπτογραφούνται ποτέ με ασύμμετρα κλειδιά στην πραγματικότητα.

Εφόσον το πείραμα αυτό έχει να κάνει με μεγάλου όγκου δεδομένα, που στην μικρότερη τους μορφή θα κυμαίνονται γύρο στα μερικά GB, δεν πρόκειται να γίνει σύγκριση ενός υβριδικού κρυπτογραφικού συστήματος με ένα σχήμα που χρησιμοποιεί τεχνικές ασύμμετρης κρυπτογραφίας, διότι τα αποτελέσματα θα είναι φανερά προβλέψιμα.

Για την αξιολόγηση του CCBKE, υλοποιήθηκαν δύο σχήματα ανταλλαγής κλειδιών, το πρώτο έκανε χρήση του CCBKE και το δεύτερο ένα Java: basic multi-user IKE σχήμα. Για τις παραμέτρους της ανταλλαγής κλειδιού Diffie-Hellman χρησιμοποιήθηκε 1024-bit MODP group με ένα 1024-bit πρώτο αριθμό ρ και γεννήτρια $g = 2$. Για τη λειτουργία του *prf* έγινε χρήση του ψευδοτυχαίου κώδικα κατακερματισμού MD5, και για την υπογραφή ο αλγόριθμος RSA. Εφαρμόστηκαν τα δύο αυτά σχήματα ανταλλαγής κλειδιών πάνω στο περιβάλλον του νέφους με μερικά σύνολα δεδομένων αστροφυσικής, και καταγράφτηκε η συνολική κατανάλωση χρόνου στον CLC.

Στο επόμενο υποκεφάλαιο θα γίνει αξιολόγηση της απόδοσης του CCBKE αξιολογώντας τη σημαντική βελτίωση της αποδοτικότητας στον CLC.

4.3 Ανάλυση και αποτελέσματα πειράματος

Στην αρχή αποδεικνύετε ότι τα σχήματα ανταλλαγής κλειδιού χρειάζονται ένα μεγάλο ποσοστό στο χρόνου που καταναλώνετε όταν τρέχουν σε περιβάλλοντα κατανεμημένης υπολογιστικής όπως είναι το νέφος, γεγονός που δείχνει τη σημασία της έρευνας στις τεχνικές για την αποτελεσματική αυθεντικοποιημένη ανταλλαγή κλειδιού. Κατά την εφαρμογή υβριδικής κρυπτογραφίας σε εφαρμογές με μεγάλου όγκου δεδομένα, τα σχήματα ανταλλαγής κλειδιού χρησιμοποιούνται σε συνδυασμό με κρυπτογραφία συμμετρικού κλειδιού για να διασφαλιστεί η ασφάλεια των δεδομένων. Σε αυτά τα σενάρια η κατανάλωση χρόνου της ανταλλαγής κλειδιού μπορεί να αγνοηθεί σε σύγκριση με την χρονοβόρα κατανάλωση για τη κρυπτογράφηση. Ωστόσο η κατάσταση είναι διαφορετική στο cloud computing ως εξής. Μια υποδομή cloud συχνά απασχολεί χιλιάδες μέρη του διακομιστή. Για μια εφαρμογή που έχει υψηλή ένταση δεδομένων και ο χρόνος καθίσταται κρίσιμος, όπως σε μια επιστημονική εφαρμογή, τα μεγάλα σύνολα δεδομένων από GB διαιρούνται σε κομμάτια μερικών MB και διανέμονται στα μέρη του διακομιστή μέσο του MapReduce για να τύχουν επεξεργασίας. Στο πείραμα χρησιμοποιείτε ο χρόνος κατανάλωσης δεδομένων του

IKE για να αντιπροσωπεύσει την αποτελεσματικότητα του σχήματος ανταλλαγής κλειδιού CCBKE. Για τους αλγόριθμους κρυπτογραφίας συμμετρικού κλειδιού, συμπεριλήφθηκαν δύο αλγόριθμοι που χρησιμοποιήθηκαν για την κρυπτογράφηση των συνόλων των δεδομένων, ο πιο διαδεδομένος block cipher, *AES*, που τρέχει σε Galois Counter Mode (GCM) των 64K tables, και ο salsa 20/12, ένας stream cipher ο οποίος είναι διάσημος λόγω της υψηλής του απόδοσης στη κρυπτογράφηση μεγάλων συνόλων από δεδομένα. Έχει αποδειχτεί ότι και οι δύο αλγόριθμοι είναι ασφαλής έναντι διαφόρων τύπου επιθέσεων κρυπτανάλυσης. Για την αποτελεσματικότητα των αλγορίθμων κρυπτογράφησης, αναφερόμαστε στα αποτελέσματα των επιδόσεων που βρίσκονται στην ιστοσελίδας της Crypto++ [68] η οποία υποδεικνύει τη ταχύτητα του *AES/GCM 64 tables* σε *108MB/s,* και τη ταχύτητα του *Salsa20/12* σε *643MB/s* για την κρυπτογραφία δεδομένων. Τα πειράματα διεξήχθηκαν σε διάφορα σύνολα δεδομένων που παραλήφθηκαν από την αστροφυσική έρευνα, τα αποτελέσματα παρατίθενται στους πιο κάτω πίνακες.

Σύγκριση πρώτης επανάληψης, κατανάλωσης χρόνου ανταλλαγής κλειδιού και κρυπτογραφίας AES στο CLC

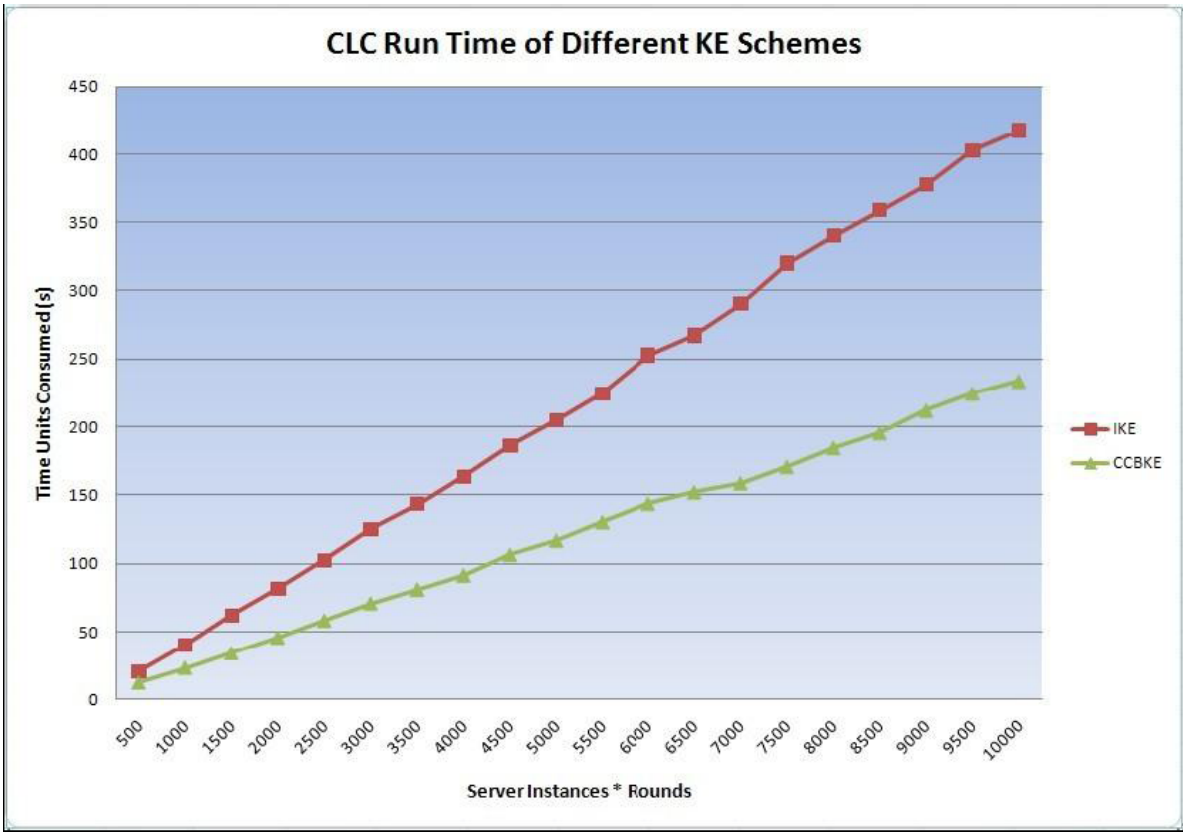
Dataset Size (GB)	**2**	**8**	**12**	**15**	**32**
Server Instances Involved	100	500	1000	1500	4000
Data Block Size (MB)	20	16	12	10	8
AES/GCM Encryption Time (s)	18.52	74.07	111.11	138.89	296.31
IKE Key Exchange Time (s)	4.04	20.48	41.77	61.92	163.10
Key Exchange Take Percentage of (%)	**17.91**	**21.66**	**27.32**	**30.84**	**35.50**

Σύγκριση πρώτης επανάληψης, κατανάλωσης χρόνου ανταλλαγής κλειδιού και κρυπτογραφίας Salsa στο CLC

Dataset Size (GB)	2	8	12	15	32
Server Instances Involved	100	500	1000	1500	4000
Data Block Size (MB)	20	16	12	10	8
Salsa20/12 Encryption Time (s)	3.11	12.44	18.66	23.33	49.77
IKE Key Exchange Time (s)	4.04	20.48	41.77	61.92	163.10
Key Exchange Take Percentage of (%)	56.50	62.21	69.12	72.63	76.62

Αυτό αποδεικνύει ότι σε περιβάλλοντα υβριδικού cloud computing, οι διεργασίες ανταλλαγής κλειδιού καταναλώνουν ένα μεγάλο ποσοστό του χρόνου που καταναλώθηκε σε ένα σχήμα ασφάλειας (αυτό το ποσοστό θα είναι ακόμη μεγαλύτερο σε ένα πραγματικό σενάριο, αφού η αποδοτικότητα τις ανταλλαγής κλειδιού εξαρτάτε από τον προγραμματισμένο αλγόριθμο και τη κατάσταση του δικτύου, ενώ ο χρόνος της κρυπτογράφησης παραμένει σχετικά σταθερός). Αυτό σημαίνει ότι η συνολική απόδοση, θα βελτιωθεί σημαντικά αν χρησιμοποιηθεί ένα σχήμα που έχει καλύτερη απόδοση στην ανταλλαγή κλειδιού.

Σύμφωνα με την ανάλυση του προβλήματος στα προηγούμενα υποκεφάλαια, ο χρόνος εκτέλεσης στον CLC είναι ο επικρατέστερος παράγοντας για τη σύγκριση της αποδοτικότητας δύο ή περισσότερων σχημάτων για την αυθεντικοποιημένη ανταλλαγή κλειδιού σε ένα δίκτυο τύπου νέφους. Έχοντας συγκριθεί βάση της κατανάλωσης χρόνου στον CLC η αποδοτικότητα των δύο τεχνικών ανταλλαγής κλειδιού επιδεικνύεται στο πιο κάτω διάγραμμα:



Διάγραμμα αποδοτικότητας IKE με CCBKE

Μπορούμε να δούμε ξεκάθαρα ότι το σχήμα CCBKE καταναλώνει περίπου το μισό χρόνο στο χρόνο εκτέλεσης του CLC σε σύγκριση με το IKE, βελτίωση η

οποία θεωρείτε εξαιρετικά σημαντική. Σε ένα σύστημα όπου ο χρόνος είναι περιορισμένος και υπάρχουν πολλές επαναλήψεις στις λειτουργίες των εφαρμογών, όπου εμπλέκονται χιλιάδες μέρη του διακομιστή, μπορούν να σωθούν πολλές ώρες στο χρόνο κατανάλωσης. Αυτόματα, επιτυγχάνονται γρηγορότερα αποτελέσματα και λιγότερες πιθανότητες να χαθεί μια επιστημονική ανακάλυψη.

ΚΕΦΑΛΑΙΟ 5. ΣΥΜΠΕΡΑΣΜΑΤΑ ΚΑΙ ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

5.1 Συμπεράσματα

Το cloud computing θα αρχίσει να κυριαρχεί στην αγορά, διότι είναι πλέον μια ανάγκη, μια υπηρεσία που θα προσφέρεται για το κοινό και τις επιχειρήσεις-οργανισμούς. Οι χρήστες μπορούν να έχουν πρόσβαση στο νέφος από οποιανδήποτε συσκευή η οποία υποστηρίζει τη τεχνολογία αυτή και είναι συνδεδεμένη σε κάποιο δίκτυο. Υιοθετώντας τη τεχνολογία του νέφους προσκομίζονται πολλά πλεονεκτήματα, κυρίως στο κόστος, αλλά κι κάποια θέματα-προβλήματα από τις απαιτήσεις της ασφάλειας που δημιουργούνται. Για αυτά τα θέματα ασφαλείας επιστρατεύτηκαν διάφορες τεχνικές και αλγόριθμοι που ενδεχομένως να δώσουν λύσεις αφού πολλές τεχνικές είναι αποτελεσματικές στην υπάρχουσα τεχνολογία δικτύου και μπορούν να υλοποιηθούν και στο cloud. Ο συνδυασμός των τεχνικών αυτών σε συνεργασία, μέσα σε ένα υβριδικό περιβάλλον, έχουν πολύ μεγάλες δυνατότητες και μπορούν να χρησιμοποιηθούν για την επίλυση διαφόρου τύπου προβλημάτων.

Λύσεις στα προβλήματα της ασφαλούς επικοινωνίας, αλλά και της διασφάλισης των αποθηκών δεδομένων θα βρεθούν μέσα από την επιστήμη της κρυπτογραφίας. Η κρυπτογραφία διαφυλάσσει την εμπιστευτικότητα στη πρόσβαση-επικοινωνία, την ακεραιότητα του αρχείου(μη αλλοίωση δεδομένων), τη μη απάρνηση του αποστολέα και του παραλήπτη, και τη πιστοποίηση της ταυτότητας των μελών καθώς και της πηγής-προορισμού της πληροφορίας. Στο δεύτερο κεφάλαιο έγινε εκτεταμένη αναφορά για τη λειτουργία διάφορων αλγόριθμων και τεχνικών κρυπτογραφίας όπως των Des, Triple-Des, RSA, MD5, SHA, PEKS, IBE, FPE, AES, SSL/TLS, SSH, με τους τελευταίους(PEKS, IBE, FPE, AES, SSL/TLS, SSH) να υπερτερούν σε καταλληλότητα και λειτουργικότητα για τη χρήση τους σε ένα δίκτυο νέφους.

Η διανομή κλειδιών στα δίκτυα επικοινωνίας αποτελεί ένα μεγάλο κομμάτι στην ασφαλή κρυπτογραφία. Στο τρίτο κεφάλαιο της έρευνας αυτής περιγράφηκαν τρεις τεχνικές διανομής κλειδιών. Η πρώτη τεχνική που αναφέρθηκε είναι η υποδομή δημοσίου κλειδιού (PKI) η οποία έχει υλοποιηθεί, και χρησιμοποιήθηκε για αρκετό καιρό, λόγω του ότι παρέχει ένα αποδεκτό επίπεδο ασφάλειας, αλλά για το περιβάλλον του cloud τίθεται ακατάλληλη για χρήση, διότι η υλοποίηση της είναι αρκετά δύσκολη και για πολλές περιπτώσεις είναι αδύνατη. Χωρίς να έχει την δυνατότητα να εξυπηρετήσει ένα μεγάλο φάσμα κοινού, αποκλίσαμε τη χρήση της τεχνικής αυτής για τη διανομή κλειδιών στο cloud. Η δεύτερη τεχνική η οποία αναλύθηκε ονομάζεται IKE(Internet Key Exchange Protocol). Το IKE είναι ένα υβριδικό πρωτόκολλο διαχείρισης προτύπων το οποίο παρέχει ένα πολύ υψηλό επίπεδο ασφάλειας και λόγω της δομής του, που υποστηρίζει πολλά πρότυπα, κυριάρχησε στο χώρο του διαδικτύου και χρησιμοποιείτε μέχρι και σήμερα. Το IKE μπορεί να υιοθετηθεί και στο cloud, παρέχοντας ένα αποδεκτό επίπεδο ασφάλειας, και να επιλύσει πολλά προβλήματα στην ασφαλή επικοινωνία των χρηστών. Όταν όμως έρχεται αντιμέτωπο με μεγάλου όγκου δεδομένα παρατηρείτε μεγάλη καθυστέρηση τόσο στην ανταλλαγή κλειδιών όσο και στην κρυπτογράφηση τους. Η τρίτη τεχνική ανταλλαγής κλειδιών που μελετήσαμε είναι το σχήμα CCBKE (Cloud Computing Background Key Exchange) και είναι βασισμένη στη στρατηγική της τυχαίας επαναχρησιμοποίησης και στο σχήμα του πρωτοκόλλου IKE. Παρέχει ένα επίπεδο ασφάλειας ίδιο με το IKE. Στην έρευνα περιγράφηκε αναλυτικά ο τρόπος με τον οποίο λειτουργεί το σχήμα αυτό επανάληψη προς επανάληψη, και αποδείχθηκε με επιτυχία, με τους ορισμούς και της υποθέσεις που διενεργήθηκαν, ότι στο σχήμα CCBKE δεν μπορεί να διενεργηθεί οποιαδήποτε πράξη πλαστοπροσωπίας ή υποκλοπής των κλειδιών που ανταλλάσσονται για την ασφαλή επικοινωνία από επιτιθέμενους εντός αλλά και εκτός του νέφους.

Μια ομάδα επιστημόνων στη προσπάθειά τους να βελτιώσουν τα μειονεκτήματα του IKE διενήργησαν κάποια πειράματα και συγκρίνανε της δύο τεχνικές, το IKE και το CCBKE. Το πείραμα αυτό, που έλαβε χώρα σε ένα πανεπιστήμιο στην Αυστραλία, τεκμηριώνει τους υπολογισμούς που έγιναν για τη διανομή κλειδιού και καταλήγει σε βάσιμα συμπεράσματα ότι το CCBKE υπερέχει του IKE. Το σχήμα CCBKE καταναλώνει περίπου το μισό χρόνο από τον οποίο χρειάζεται το IKE για να φτάσει στη κρυπτογραφία ενός μεγάλου, σε όγκο, πακέτου δεδομένων. Όταν το CCBKE εφαρμοστεί στο cloud, μπορεί πράγματι να βελτιώσει σημαντικά την αποδοτικότητα του IKE, το οποίο δείχνει να καθυστερεί στις

κρυπτογραφήσεις μεγάλου όγκου δεδομένων και στις πράξεις υψηλής πυκνότητας. Χαρακτηριστικά τα οποία διακατέχουν πολλές επιστημονικές εφαρμογές. Για αυτό προτάθηκε το σχήμα CCBKE για να τις επιστημονικές εφαρμογές, το οποίο δεν έχει ακόμη υιοθετηθεί, αλλά γίνονται συνεχώς έρευνες για την περαιτέρω βελτίωση του.

5.2 Μελλοντικές επεκτάσεις

Σε αυτό το έγγραφο μελετήθηκαν διάφορες τεχνικές κρυπτογραφίας και καναλιών ασφαλούς επικοινωνίας. Επίσης έγινε εκτενής αναφορά στα μοντέλα ανταλλαγής κλειδιών και αναλύθηκαν για κάποια μοντέλα, βήμα προς βήμα, οι διαδικαστικές επαναλήψεις που διενεργούνται, οι επιδράσεις τους, και τα αποτελέσματα που επιφέρουν. Προτάθηκε ένα καινοφανή σχήμα που τα παραγόμενα των πειραμάτων στα οποία υποβλήθηκε, κατέληξαν, με τεκμηριωμένα αποτελέσματα, στο συμπέρασμα ότι το μοντέλο διανομής κλειδιών CCBKE τείνει να βελτιώνει κατά πολύ μεγάλο ρυθμό τη κατανάλωση χρόνου στη διανομή κλειδιού για ασφαλή επικοινωνία στις επιστημονικές εφαρμογές.

Εφόσον έχει ήδη προταθεί ένα αποτελεσματικό σύστημα ανταλλαγής κλειδιού που μπορεί να μειώσει σημαντικά την κατανάλωση του χρόνου, η αποδοτικότητα του ασφαλούς προγραμματισμού για επιστημονικές εφαρμογές σε υβριδικά περιβάλλοντα μπορεί να αναπτυχθεί περαιτέρω. Η ανάπτυξη αυτή μπορεί να υλοποιηθεί με την ενσωμάτωση αποδοτικότερης, από πλευράς κόστους σε χρόνο, κρυπτογράφησης, η οποία εξακολουθεί να καταναλώνει ένα μεγάλο μέρος στη κατανάλωσης χρόνου ακόμα και στο cloud computing για επιστημονικές εφαρμογές. Ως εκ τούτου, στο μέλλον, θα διερευνούσα περαιτέρω νέες στρατηγικές για τη βελτίωση της αποδοτικότητας και αποτελεσματικότητας της κρυπτογραφίας συμμετρικού κλειδιού για πιο ασφαλή και αμυντικογενή προγραμματισμό επιστημονικών εφαρμογών.

ΚΕΦΑΛΑΙΟ 6. ΑΝΑΦΟΡΕΣ

1. http://www.kiosterakis.gr/new/ti-einai-to-cloud-computing , (προσπελάστηκε στις 3/9/2012)
2. "The NIST Definition of Cloud Computing". National Institute of Science and Technology. Retrieved 24 July 2011.
3. 'The Institute IEEE', "A View inside the Cloud", by Ania Monaco, June 2012, page 8
4. M. A. Rahaman, A. Schaad, and M. Rits, “Towards secure SOAP message exchange in a SOA,” in SWS ’06: Proceedings of the 3rd ACM workshop on Secure Web Services. ACM Press, 2006, pp. 77–84.
5. S. Gajek, L. Liao, and J. Schwenk, “Breaking and fixing the inline approach,” in SWS ’07: Proceedings of the 2007 ACM workshop on Secure web services. New York, NY, USA: ACM, 2007, pp. 37–43.
6. N. Gruschka and L. Lo Iacono, “Vulnerable Cloud: SOAP Message Security Validation Revisited,” in ICWS ’09: Proceedings of the IEEE International Conference on Web Services. Los Angeles, USA: IEEE, 2009.
7. Google, “Browser security handbook,” 2009.
8. D. Kaminski, “Dns server+client cache poisoning, issues with ssl, breaking *forgot my password* systems, attacking autoupdaters and unhardened parsers, rerouting internal traffic; ” -, 2008.
9. S. Stamm, Z. Ramzan, and M. Jakobsson, “Drive-by pharming,” Indiana University Computer Science, Tech. Rep. 641, 2006.
10. D. Kormann and A. Rubin, “Risks of the passport single signon protocol,” Computer Networks, vol. 33, no. 1–6, pp. 51–58, 2000.
11. *Laudenslager, P., six cloud computing benefits,* http://www.informationweek.com/six-cloud-computing-benefits-for-smbs/225200751 (προσπελάστηκε στις 15/9/2012).
12. http://en.wikipedia.org/wiki/Microsoft_account (προσπελάστηκε στις 25/9/2012).
13. M. Slemko, “Microsoft passport to trouble,” 2001, http://alive.znep.com/□marcs/passport/.
14. T. Scavo, “SAML V2.0 Holder-of-Key Assertion Profile,” Working Draft 09, 20.01.2009, 2009, http://www. Oasis

open.org/apps/org/workgroup/security/download. php/30782/sstc-saml2-holder-of-key-draft-09.pdf.

15. S. Gajek, T. Jager, M. Manulis, and J. Schwenk, “A Browser-based Kerberos Authentication Scheme,” in Computer Security - ESORICS 2008, 13th European Symposium on Research in Computer Security, M´alaga, Spain, LNCS 5283. Springer, 2008, pp. 115–129.
16. http://en.wikipedia.org/wiki/Denial-of-service_attack
17. http://en.wikipedia.org/wiki/SYN_flood
18. Andrew S. Tanenbaum, “*Δίκτυα Υπολογιστών”,* Παπασωτηρίου 2000
19. Federal Information Processing Standards Publication 46-3, “Data Encryption Standard”, U.S Department of Commerce/National Institute of Standards and Technology, 1999. http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf
20. X. Lai, “On the design and security of block ciphers”, *ETH Series in Information Processing* (J.L. Massey, ed.), Vol. 1, Hartung-Gorre Verlag Konstanz, Technische Hochschule (Zurich), 1992.
21. R. Baldwin, R. Rivest, “The RC5, RC5-CBC, RC5-CBC-Pad, and RC5-CTS Algorithms” RFC2040, 1996 http://www.ietf.org/rfc/rfc2040.txt
22. Adams C, “The CAST-128 Encryption Algorithm” RFC2144, May 1997.
23. Federal Information Processing Standards Publication 197, “Announcing the Advanced Encryption Standard (AES)”, NIST, 2001 http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf. (προσπελάστηκε στις 2/10/2012)
24. https://www.eff.org/ (προσπελάστηκε στις 2/10/2012).
25. http://en.wikipedia.org/wiki/Lucifer_(cipher) (προσπελάστηκε στις 2/10/2012).
26. www.**nist**.gov/ (προσπελάστηκε στις 2/10/2012).
27. http://en.wikipedia.org/wiki/MARS_(cryptography) (προσπελάστηκε στις 8/10/2012).
28. http://www.logic.at/wiki/index.php/Rijndael (προσπελάστηκε στις 8/10/2012).
29. http://en.wikipedia.org/wiki/Serpent_(cipher) (προσπελάστηκε στις 8/10/2012).
30. http://en.wikipedia.org/wiki/RC6 (προσπελάστηκε στις 8/10/2012).
31. http://en.wikipedia.org/wiki/Twofish(προσπελάστηκε στις 8/10/2012).
32. www.itl.nist.gov/fipspubs/(προσπελάστηκε στις 8/10/2012).

33. http://en.wikipedia.org/wiki/Block_cipher_modes_of_operation (προσπελάστηκε στις 9/10/2012).
34. http://en.wikipedia.org/wiki/RSA_(algorithm) (προσπελάστηκε στις 10/10/2012).
35. Descriptions of SHA-256, SHA-384 and SHA-512, http://csrc.nist.gov/CryptoToolkit/shs/sha256-384-512.pdf (προσπελάστηκε στις 12/10/2012).
36. Rivest R., "The MD5 Message-Digest Algorithm", RFC1321, IETF, 1992 http://www.ietf.org/rfc/rfc1321.txt (προσπελάστηκε στις 12/10/2012).
37. Dobbertin H., Bosselaers A., Preneel B., "RIPEMD-160, A Strengthened Version of RIPEMD", 1996
38. Descriptions of SHA-256, SHA-384 and SHA-512, http://csrc.nist.gov/CryptoToolkit/shs/sha256-384-512.pdf (προσπελάστηκε στις 12/10/2012).
39. http://www.webopedia.com/TERM/D/digital_envelope.html (προσπελάστηκε στις 13/10/2012).
40. http://en.wikipedia.org/wiki/SSL (προσπελάστηκε στις 13/10/2012).
41. http://en.wikipedia.org/wiki/Transport_Layer_Security (προσπελάστηκε στις 13/10/2012)
42. http://en.wikipedia.org/wiki/Dictionary_attack(προσπελάστηκε στις 13/10/2012)
43. http://en.wikipedia.org/wiki/Replay_attack(προσπελάστηκε στις 14/10/2012)
44. http://en.wikipedia.org/wiki/Man-in-the-middle_attack (προσπελάστηκε στις 14/10/2012)
45. http://en.wikipedia.org/wiki/Transport_Layer_Security (προσπελάστηκε στις 18/10/2012)
46. http://en.wikipedia.org/wiki/Secure_Shell (προσπελάστηκε στις 26/10/2012)
47. http://en.wikipedia.org/wiki/Generic_Security_Services_Application_Program_Interface (προσπελάστηκε στις 26/10/2012)
48. http://eprint.iacr.org/2005/191.pdf (προσπελάστηκε στις 26/10/2012)
49. Khader, D., στο http://eprint.iacr.org/2006/358.pdf (προσπελάστηκε στις 5/11/2012)
50. http://en.wikipedia.org/wiki/ID-based_encryption (προσπελάστηκε στις 13/11/2012)
51. http://en.wikipedia.org/wiki/Adi_Shamir (προσπελάστηκε στις 13/11/2012)

52. http://en.wikipedia.org/wiki/Format-preserving_encryption#The_motivation_for_FPE (προσπελάστηκε στις 20/10/2012)
53. http://en.wikipedia.org/wiki/Public_key_infrastructure(προσπελάστηκε στις 2/11/2012)
54. http://en.wikipedia.org/wiki/X.509 (προσπελάστηκε στις 2/12/2012)
55. Cisco systems 2001, στο http://www.net130.com/tutorial/cisco-pdf/4T_IKE_Bcamp.pdf
56. Krawczyk, H., στο http://www.dia.unisa.it/~ads/corso-security/www/CORSO-9900/oracle/skeme.pdf (προσπελάστηκε στις 22/10/2012)
57. http://en.wikipedia.org/wiki/IPsec (προσπελάστηκε στις 24/10/2012)
58. http://en.wikipedia.org/wiki/IPsec#Security_architecture(προσπελάστηκε στις 26/10/2012)
59. http://en.wikipedia.org/wiki/Diffie-Hellman (προσπελάστηκε στις 26/10/2012)
60. http://en.wikipedia.org/wiki/Hash-based_message_authentication_code
61. Australia Telescope, Parkes Observatory. Available: http://www.parkes.atnf.csiro.au/ (προσπελάστηκε στις 7/11/2012)
62. http://www.ibm.com/developerworks/cloud/library/cl-mapreduce/ (προσπελάστηκε στις 7/11/2012)
63. http://www.di.ens.fr/~pnguyen/MPRI/MPRI2010_Squareroot.pdf
64. Xen Hypervisor. http://xen.org/ (προσπελάστηκε στις 7/11/2012)
65. http://hadoop.apache.org/ (προσπελάστηκε στις 10/11/2012)
66. http://en.wikipedia.org/wiki/MapReduce (προσπελάστηκε στις 12/11/2012)
67. http://www.eucalyptus.com/ (προσπελάστηκε στις 20/11/2012)
68. http://www.cryptopp.com/benchmarks.html(προσπελάστηκε στις 20/11/2012)