ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

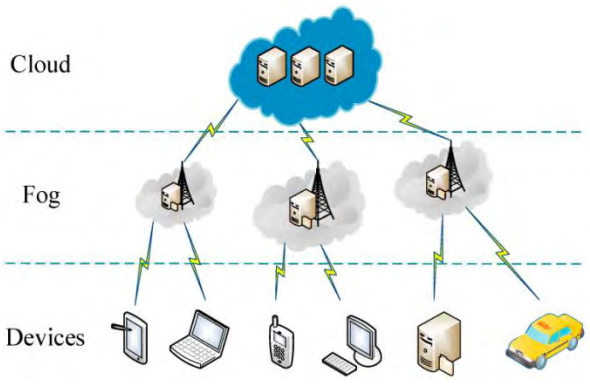
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

«Αποδοτικοί αλγόριθμοι ασφάλειας για εφαρμογές υπολογιστικού νέφους και ομίχλης-Efficient security algorithms for cloud and fog computing applications.»

Σπουδάστρια : Αντωνία Β. Σασιλιόγλου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Επιβλέπων : Σταμούλης Γεώργιος



Λαμία, 2019

Υπεύθυνη Δήλωση περί πνευματικών δικαιωμάτων και λογοκλοπής:

«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις [(1)], που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

*1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί **χωρίς να τα περικλείω σε εισαγωγικά** και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάσθηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.*

*2. Δέχομαι ότι η αυτολεξεί **παράθεση χωρίς εισαγωγικά**, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.*

3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια

4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

Ο/Η ΔΗΛΩΝ/-ΟΥΣΑ

Ημερομηνία

Υπογραφή

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε

να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»

Πίνακας Περιεχομένων

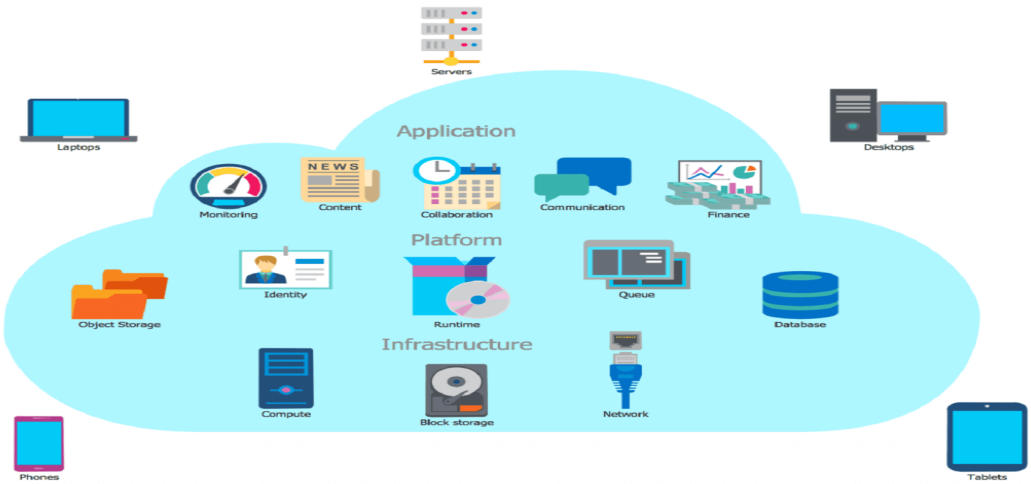
Εισαγωγική ενότητα

Στη παρούσα διπλωματική εργασία αναλύονται ορισμένοι αποδοτικοί αλγόριθμοι ασφάλειας για εφαρμογές υπολογιστικού νέφους και ομίχλης. Συγκεκριμένα αναλύεται αρχικά η έννοια του υπολογιστικού νέφους. Έπειτα παρουσιάζονται η αρχιτεκτονική των Cloud Computing συστημάτων, τα μοντέλα υπολογιστικού νέφους καθώς και η χρήση και η αναγκαιότητα του Fog Computing. Η ασφάλεια στο υπολογιστικό νέφος αποτελεί μια από τις σημαντικότερες ενότητες της εργασίας καθώς θίγονται εκεί οι έννοιες της εμπιστευτικότητας και της ιδιωτικότητας, τεχνολογίες με βαρύνουσα χρήση. Τέλος γίνεται συσχέτιση και εκτενής ανάλυση σε αλγόριθμους ασφάλειας.

Κεφάλαιο 1ο: Υπολογιστικό Νέφος, ορισμοί και χρήσεις (Cloud Computing)

1.1 Ορισμός Υπολογιστικού Νέφους

Η έννοια του υπολογιστικό νέφος δημιουργήθηκε ως απόρροια της εξέλιξης της τεχνολογίας με σκοπό να μειωθούν τα έξοδα των πελατών. Το όνομα του cloud computing προέκυψε από τον συσχετισμό του διαδικτύου ως ένα σύννεφο (cloud) σε διαγράμματα δικτύου. Πιο συγκεκριμένα η χρήση του υπολογιστικού νέφους είναι να δίνει τη δυνατότητα στους χρήστες να χρησιμοποιούν εφαρμογές και πόρους που δεν βρίσκονται εγκατεστημένες τοπικά σε κάποιον ηλεκτρονικό υπολογιστική ή σε κάποια άλλη ηλεκτρονική συσκευή με πρόσβαση στο διαδίκτυο αλλά απομακρυσμένα σε κάποιο κέντρο δεδομένων. Η δομή του υπολογιστικού νέφους παρουσιάζεται και στην εικόνα 1 περιλαμβάνει τρία συστατικά, τους υπολογιστές πελάτες , τους διακομιστές που βρίσκονται σε διαφορετικά σημεία και το απομακρυσμένο κέντρο δεδομένων που περιλαμβάνει όλους τους διακομιστές που εξυπηρετούν τις εφαρμογές. Αξίζει να τονιστεί ότι δεν υπάρχει συγκεκριμένη δομή υλοποίησης στο υπολογιστικό νέφος. [35],[36]



Εικόνα 1: Η δομή του Cloud Computing

1.2 Ιστορική Αναδρομή

Η έννοια του cloud computing εμφανίστηκε το 1996 όταν αναφέρθηκε σε έγγραφο της εταιρείας Compaq. Η ιστορία του υπολογιστικού νέφους όμως ξεκινάει πολύ πιο πριν στις αρχές του 1960 με μορφή κατανεμημένου συστήματος.

Το 1960 ο J.C.R Licklider που εργάστηκε για την εταιρεία DAPRA και μέλος του υπουργείου Άμυνας των Η.Π.Α οραματίστηκε ένα "παγκόσμιο δίκτυο υπολογιστών" . Αρχικά είχε ως στόχο οι επιστήμονες να ανταλλάσουν πληροφορίες μέσω τεσσάρων υπολογιστών. Έπειτα προσπάθησε να δημιουργήσει ένα σύστημα όπου οι χρήστες θα είχαν πρόσβαση σε πόρους από οποιοδήποτε γεωγραφικό μέρος επιθυμούσαν.

Επίσης το 1960 ο John Mc Carthy ανέφερε ότι "η αξιοποίηση του χρόνου χρήσης υπολογιστικών πόρων μπορεί κάποια μέρα να οργανωθεί ως κοινής ωφελείας." θεωρώντας ότι υπήρξε η αρχή για τη δημιουργία νέφους. Η έννοια cloud computing χρησιμοποιήθηκε πρώτη φορά το 1977 σε διάλεξη του επιστήμονα Ramnath K Chellappa.

Το 1980 το Εθνικό Επιστημονικό Ίδρυμα NFS επικεντρώθηκε στη δημιουργία ενός συστήματος όπου θα ήταν βασισμένα τα πρωτόκολλα TCIP/IP. Άλλος ένας στόχος τους ήταν να δημιουργηθούν κέντρα που θα περιλαμβάνουν υπέρυπολογιστές, στη δράση αυτή λάβανε μέρος πολλοί μεγάλοι οργανισμοί όπως η IBM.

Μετά το 1983 δημιουργήθηκε το τωρινό διαδίκτυο όπου τότε είχε ονομαστεί ως το δίκτυο των δικτύων. Ο Tim Bernels-Leeto 1990 δημιούργησε τον παγκόσμιο και έπειτα όλοι οι επιστημονικοί υπολογιστές απόκτησαν πρόσβαση στο διαδίκτυο.

Το 1996 ο Douglas Parkhill σημείωνε στο βιβλίο του με τίτλο "The challenge of the Computer utility" σχεδόν όλα τα χαρακτηριστικά του νέφους και παρέθετε διαφορές με άλλες μορφές κυρίως με την ηλεκτρική ενέργεια.

Στα τέλη του 1990 και πιο συγκεκριμένα το 1999 έκανε την εμφάνιση της η εταιρεία Salesforce.com η οποία ήταν η πρωτοπόρα με την δημιουργία του γνωστού μοντέλου πλέον Software as a Service (SaaS) δίνοντας το βήμα για τη δημιουργία

άλλων εταιρειών όπως η Microsoft Office 365.Πιο συγκεκριμένα η Salesforce δημιουργεί εφαρμογές οι οποίες είναι κατασκευασμένες για να λειτουργούν στο "σύννεφο" και να διατίθενται σε πολλές επιχειρήσεις σε χαμηλό κόστος.

Έπειτατο 2002 ακολούθησε η Amazonέχοντας ως όραμα να δημιουργήσουν υπηρεσίες τοποθετημένες σε "σύννεφο" με πληρωμή από τους χρήστες ανάλογη με την χρήση, έτσι δημιούργησαν την πλατφόρμα Amazon Web Services (AWS).

Έπειτα το 2006 η Amazon.comδημιούργησε άλλη μια πλατφόρμα την Elastic Compute Cloud (EC2) με σκοπό πελάτες είτε ιδιώτες είτε επιχειρήσεις να ενοικιάζουν υπολογιστές ώστε να υλοποιούν εκεί τις εφαρμογές τους. Αναδείχθηκε και πάλι πρωτοπόρα με τον πλήρη έλεγχο πόρων που πρόσφερε στο υπολογιστικό νέφος. Έπειτα ακολούθησαν και άλλες εταιρείες όπως η Netflix. Επομένως η Amazon.comκατάφερε να εξελιχθεί σε μια μεγάλη επιχειρηματική μονάδα με μεγάλη οικονομική αξία.

Το 2008 η εταιρεία Google δημιούργησε την πλατφόρμα Google App Engine (GAE) δίνοντας την δυνατότητα στους προγραμματιστές διαδικτυακών εφαρμογών να τις φιλοξενούν σε κέντρα δεδομένων , η λεγόμενη πλέον υπηρεσία Platform as a Service (PaaS). Η πλατφόρμα GAE βοήθησε στην δημιουργία εφαρμογών Google Apps.

Το 2008 επίσης η εταιρεία Gartner άρχισε να επικεντρώνεται στην άνοδο του υπολογιστικού νέφους διαχωρίζοντας τους καταναλωτές των υπηρεσιών cloud computing σε χρήστες και πωλητές.

Ηεταιρεία Googleμετά από την πλατφόρμα GAE, το 2010 δημιούργησε άλλη μια πλατφόρμα την Azureπου περιλαμβάνει λύσεις PaaS, SaaS αλλά και IaaS για την ανάπτυξη διαδικτυακών εφαρμογών αλλά και κινητών μέσω διαδικτύου.

Τέλος άλλη μια εταιρεία που βοήθησε στην ανάπτυξη του cloud computing ήταν η IBM το 2011, με την δημιουργία της πλατφόρμας Smarter Computing framework χρησιμοποιώντας τις υπηρεσίες υπολογιστικού νέφους IaaS, PaaSκαι SaaS. [20],[34]

1.3 Χρήσεις στο Υπολογιστικό Νέφος

Το υπολογιστικό νέφος πλέον αποτελεί μια διαδεδομένη υπηρεσία όπου πολλές επιχειρήσεις επιλέγουν ως λύση λόγω του χαμηλού κόστους που προσφέρει συγκριτικά με τις δυνατότητες που προσφέρει. Φιλοξενούνται ποικίλες εφαρμογές διαφορετικού τύπου. Ωστόσο η κάθε επιχείρηση έχει διαφορετικές απαιτήσεις και το υπολογιστικό νέφος δεν μπορεί να ανταποκριθεί σε όλες, αλλά προσφέρει τρείς διαφορετικές λύσεις.

- Αποθήκευση στο "Σύννεφο" .Αποτελεί μια από τις πρώτες λύσεις του υπολογιστικού νέφους ωστόσο εξακολουθεί να χρησιμοποιείται από πολλές επιχειρήσεις. Λόγω της μεγάλης ζήτησης πολλοί είναι οι προμηθευτές που παρέχουν αποθήκευση δεδομένων στο σύννεφο στις εταιρείες που δεν θέλουν να αποθηκεύσουν τα αρχεία αλλά και τα πολυμέσα τους στους προσωπικούς τους υπολογιστές. Η εταιρεία Amazon αποτελεί πλέον τον πιο δημοφιλή προμηθευτή με το Amazon S3.
- Υπολογιστικά "Σύννεφα". Δίνεται η δυνατότητα στους πελάτες να έχουν πρόσβαση σε εφαρμογές που είναι τοποθετημένες στους υπολογιστικούς πόρους του παρόχου.Το Google App Engine καλύπτει αυτές τις απαιτήσεις.
- Εφαρμογές "Σύννεφου". Δίνονται στους πελάτες εφαρμογές που λειτουργούν εξαρτώμενες από το διαδίκτυο και τις λειτουργούν συνήθως μέσω browser. Ο λόγος που οι πελάτες διαλέγουν αυτή την λύση είναι ότι δεν χρειάζεται να εγκαταστήσουν να εκτελέσουν αλλά και να συντηρήσουν τις εφαρμογές στον προσωπικό τους υπολογιστή. Το Skype είναι μια τέτοια εφαρμογή. [35],[36]

1.4 Πλεονεκτήματα Υπολογιστικού Νέφους

- Όσον αφορά την κλιμάκωση προσφέρεται εύκολη και οικονομική προσθαφαίρεση υπολογιστικών πόρων όπως χώρου αποθήκευσης.
- Βασικό μέλημα των αξιόπιστων παροχών "σύννεφου" είναι η ύπαρξη ασφάλειας.
- Το κόστος του υπολογιστικού νέφους είναι μικρότερο από την υλοποίηση λύσης τοπικά στις εγκαταστάσεις των πελατών διότι δεν απαιτείται αγορά εξοπλισμού.

- Υπάρχουν αξιόλογοι προμηθευτές όπως η Google και η Amazon που μπορούν να προσφέρουν έγκυρες λύσεις "σύννεφου".
- Οι επιχειρήσεις χρειάζονται λιγότερους υπαλλήλους με την ύπαρξη του υπολογιστικού νέφους όπου οι πάροχοι του προσφέρουν υποστήριξη.
- Δεν προκύπτουν πολλά σφάλματα στην λειτουργία του cloud computing αντίθεση με τις τοπικές λύσεις των επιχειρήσεων, λόγω του ειδικευμένου προσωπικού και των μεγάλων κέντρων δεδομένων που διαθέτουν. [35],[36]

1.5 Μειονεκτήματα Υπολογιστικού Νέφους

- Πολλές φορές προκύπτουν θέματα ασφάλειας όσον αφορά τα προσωπικά δεδομένα που αποθηκεύονται στο "σύννεφο". Η μυστικότητα χάνεται από την στιγμή που οι πληροφορίες δίνονται στους παρόχους. Ωστόσο το πρόβλημα μπορεί να επιλυθεί μέσω της κρυπτογραφίας.
- Εάν κάποιος πελάτης επιθυμεί την υλοποίηση μιας εξειδικευμένης εφαρμογής που δεν υπάρχει, θα χρειαστεί να πληρώσει ο ίδιος για τη δημιουργία της.
- Δυσκολία ένταξης της εφαρμογής "σύννεφου" με τις υπόλοιπες εφαρμογής της εταιρείας, έχοντας ως αποτέλεσμα μεγαλύτερες δαπάνες.
- Η υπηρεσία του υπολογιστικού νέφους θεωρείται πιο οικονομική διότι εξαλείφει την ανάγκη για αγορά εξοπλισμού. Πολλές φορές όμως μπορεί να αποδειχθεί πιο δαπανηρή λόγω διαμόρφωσης κατάλληλου περιβάλλοντος με την αγορά λογισμικού και εγκατάστασης web browser και εύρεσης λύσεων κρυπτογράφησης. [35],[36]

1.6 Εφαρμογές Υπολογιστικού Νέφους

Το cloud computing διαθέτει εφαρμογές που χρήστες έχουν πρόσβαση από ηλεκτρονικές συσκευές (επιτραπέζιους αλλά και φορητούς υπολογιστές, κινητά τηλέφωνα, tablet, διακομιστές..). Μέσω των εφαρμογών οι χρήστες έχουν πρόσβαση σε δεδομένα. Αυτό που ξεχωρίζει τις εφαρμογές στο υπολογιστικό νέφος είναι η αποθήκευση και η ύπαρξη βάσεων δεδομένων.

Πιο αναλυτικά όσον αφορά την αποθήκευση ο πελάτης αγοράζει χώρο αποθήκευσης από τον πάροχο αντί να προμηθευτεί εξειδικευμένο μηχανισμό αποθήκευσης. Το κόστος προκύπτει από το μέγεθος του χώρου αποθήκευσης και από την τεχνική υποστήριξη που παρέχει ο προμηθευτής.

Σχετικά με τις βάσεις δεδομένων παρέχονται στον πελάτη είτε κατανεμημένες βάσεις δεδομένων δίνοντας στους πελάτες πληροφορίες που δεν βρίσκονται στο ίδιο μέρος είτε υπηρεσίες βάσεις δεδομένων όπως η Oracle Database ή Microsoft SQL Server . [35],[36]

1.7 Σύγκριση Υπολογιστικού Νέφους με άλλα συστήματα

Σε αυτό το κεφάλαιο θα αναλυθούν οι πιο σημαντικές διαφορές του cloud computing, με άλλα παρόμοια συστήματα όπως το gridκαι το fog computing.

Υπάρχουν αρκετά στοιχεία που διαφέρουν μεταξύ Cloud και Grid Computing και παρουσιάζονται παρακάτω:

- Τρόπος πληρωμής: Στο cloud computing η κοστολόγηση διαμορφώνεται ανάλογα με την χρήση της υπηρεσίας, αντίθετα στο grid computing υπάρχει σταθερή μηνιαία συνδρομή.
- Ευχρηστία: Το cloud computing είναι πιο προσιτό στους χρήστες λόγω των μειωμένων λεπτομερειών που παρουσιάζει.
- Δυναμική συστήματος: Οι πόροι του συστήματος μπορούν να διαχειριστούν ταυτόχρονα από πολλούς οργανισμούς δημιουργώντας καθυστερήσεις στο σύστημα, το πρόβλημα αυτό έλυσε το cloud computing με την δυναμική διαχείριση φόρτου εργασίας βασισμένη στον χρόνο απόκρισης των χρηστών.
- Προγραμματισμός: Ο προγραμματισμός σε περιβάλλον grid computingείναι πιο δύσκολος, λόγω του κατανεμημένου προγραμματισμού.
- Αρχιτεκτονική: Στο grid computing η αρχιτεκτονική είναι προκαθορισμένη και συγκεκριμένη ενώ στο cloud computing διαμορφώνεται ανάλογα με τις ανάγκες των πελατών.

- Ισχύς: Το cloud computing gαπαιτεί μεγάλη ισχύ λόγω των πολλών υπηρεσιών που διαθέτει σε αντίθεση με το grid computing.
- Διαλειτουργικότητα: Για να επιτευχθεί η διαλειτουργικότητα υπάρχουν συγκεκριμένα πρωτόκολλα σε συστήματα grid computing στον αντίποδα σε συστήματα cloud computing ο κάθε οργανισμός χρησιμοποιεί τα δικά του πρωτόκολλα . [19]

Το Fog Computing δημιουργήθηκε με βάση το Cloud Computing, ώστε να ξεπεράσει τις δυσκολίες που αντιμετώπισε το Cloud Computing. Ωστόσο υπάρχουν κάποιες διαφορές που παρουσιάζονται παρακάτω:

- Χρήση: Στο cloud computing οι πελάτες έχουν πρόσβαση στις εφαρμογές από οποιαδήποτε ηλεκτρονική συσκευή που διαθέτει δίκτυο σε αντίθεση με το fog computing που η πρόσβαση επιτυγχάνεται μόνο από κινητά τηλέφωνα.
- Μεγάλος όγκος δεδομένων :Το cloud computing απαιτεί πολύ μεγάλο χρόνο για να χειριστεί μεγάλο όγκο δεδομένων μπορεί η επεξεργασία τους να κρατήσεις μήνες ίσως και χρόνια, κάτι τέτοιο συνεπάγει και καλύτερο χρόνο απόκρισης. Στον αντίποδα το fog computing χρειάζεται πολύ μικρό χρόνο για την επεξεργασία των δεδομένων.
- Περιβάλλον εργασίας: Η πρόσβαση σε υπηρεσίες cloud computing μπορεί να πραγματοποιηθεί από χώρους με εξοπλισμό όπως γραφεία εταιριών, αντίθετα η πρόσβαση σε υπηρεσίες fog computingείναι πολύ πιο εύκολη αφού μπορεί να πραγματοποιηθεί και από εξωτερικούς χώρους όπως από ένα πάρκο.
- Είδος πληροφορίας: Στο cloud computing οι πληροφορίες είναι δημόσιες και έχουν συλλεχθεί από όλο τον κόσμο αντίθετα σε συστήματα “ομίχλης” η υπηρεσία πληροφοριών βρίσκεται σε προκαθορισμένη θέση. [18]

Κεφάλαιο 2 : Υπηρεσίες του Υπολογιστικού Νέφους

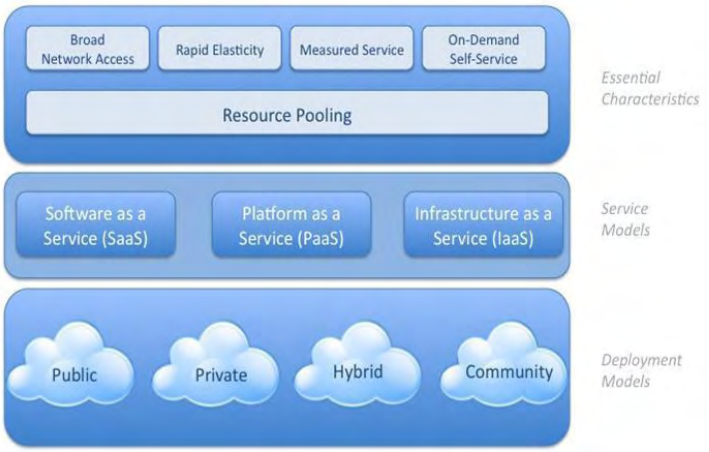
2.1 Εισαγωγή

Η υπηρεσία νέφους μπορεί να αλληλεπιδράσει με τους χρήστες με ποικίλους τρόπους, μέσω υπηρεσιών. Υπάρχουν τρεις διαφορετικοί τύποι μοντέλων που κατηγοριοποιούνται ως ένα σύνολο μοντέλων υπηρεσιών. Σε αυτό το κεφάλαιο θα μιλήσουμε για μερικές από τις υπηρεσίες που προσφέρονται από το υπολογιστικό νέφος. Πιο συγκεκριμένα θα ασχοληθούμε με το μοντέλο Software as a Service (SaaS)-Λογισμικό ως Υπηρεσία, με το μοντέλο Platform as a Service (PaaS)-Πλατφόρμα ως Υπηρεσία και με το μοντέλο Infrastructure as a Service (IaaS)-Υποδομή ως Υπηρεσία.

2.2 Αρχιτεκτονική των CloudComputing Συστημάτων

Το Εθνικό Ινστιτούτο Τυποποιήσεων και Τεχνολογίας (NIST – National Institute of Standards and Technology) είναι ένα ίδρυμα ευρέως γνωστό σε παγκόσμιο επίπεδο στον τομέα της τεχνολογίας πληροφοριών. Το NIST έχει ορίσει τις έννοιες που σχετίζονται με το cloud computing, έτσι ώστε να δημιουργήσει έναν πρότυπο, κοινό κώδικα επικοινωνίας και πιο συγκεκριμένα να υπάρξει μια βάση για συζήτηση όπως το τι είναι η υπολογιστική νέφους, τον τρόπο χρήσης του cloud στον υπολογιστή. Γι’ αυτό το λόγο θεωρούμε σημαντικό να αναφέρουμε και αυτόν τον ορισμό, καθώς η σημασία του είναι μεγάλη. Ο ορισμός που έχει δώσει το National Institute of Standards and Technology παρουσιάζονται παρακάτω. Το cloud computing είναι ένα μοντέλο που επιτρέπει ευέλικτη, on-demand δικτυακή πρόσβαση σε ένα κοινόχρηστο σύνολο παραμετροποιήσεων υπολογιστικών πόρων (π.χ. δίκτυα, servers, αποθηκευτικοί χώροι, εφαρμογές και υπηρεσίες), το οποίο μπορεί να τροφοδοτηθεί γρήγορα και να διατεθεί με ελάχιστη προσπάθεια διαχείρισης ή αλληλεπίδραση με τον πάροχο της υπηρεσίας. Αυτό το cloud μοντέλο προωθεί την διαθεσιμότητα και αποτελείται από πέντε βασικά χαρακτηριστικά, τρία μοντέλα παροχής υπηρεσιών που παρουσιάζονται στην εικόνα 2, και τέσσερα μοντέλα ανάπτυξης. Το NIST ορίζει την αρχιτεκτονική του Cloud Computing με πέντε

ουσιώδη χαρακτηριστικά, τρία μοντέλα υπηρεσίας νέφους και τέσσερα μοντέλα ανάπτυξης νέφους. [37],[40]



Εικόνα 2: Η Αρχιτεκτονική του μοντέλου Cloud Computing

Το σύστημα Cloud Computing πρέπει να αποτελείται από κάποια κύρια χαρακτηριστικά όπως την αυτό-εξυπηρέτηση κατά απαίτηση , όπου οι καταναλωτές μπορούν να παρέχουν τις υπολογιστικές ικανότητες (χρόνο διακομιστή, αποθήκευση δικτύου) χωρίς να χρειάζεται η αλληλεπίδραση των ανθρώπων με τους παρόχους των υπηρεσιών. Χρήσιμη είναι η ευρεία πρόσβαση στο δίκτυο, μέσω κάλυψης του δικτύου και πρόσβασης μέσω τυποποιημένων μηχανισμών όπως κινητά τηλέφωνα. Διατίθενται υπολογιστικοί όπως μνήμες και επεξεργαστές του παρόχου στους χρήστες. Οι πόροι τόσο φυσικοί όσο και εικονικοί πόροι σύμφωνα με τη ζήτηση των καταναλωτών. Επίσης αποτελείται από ελαστικότητα, όπου οι υπηρεσίες παρέχονται στους χρήστες γρήγορα και αποτελεσματικά κάνοντας τες συχνά να φαίνονται απεριόριστες και μπορούν να διατίθενται σε οποιαδήποτε ποσότητα ανά πάσα στιγμή. Τέλος οι υπηρεσίες του συστήματος είναι μετρίσιμες, αφού οι πόροι ελέγχονται και βελτιστοποιούνται αυτόματα, σχετικά με τους πόρους υπάρχει διαφάνεια τόσο για την χρήση τους από τους παρόχους όσο και από τους χρήστες. [38]

2.2.1 Μοντέλα του Υπολογιστικού Νέφους

Τα μοντέλα του υπολογιστικού νέφους μπορούν να διακριθούν σε μοντέλα υπηρεσίας και σε μοντέλα ανάπτυξης.

Μοντέλα υπηρεσίας του υπολογιστικού νέφους θεωρούνται:

- Λογισμικό Νέφους ως υπηρεσία (SaaS). Αποτελεί μια ολοκληρωμένη εφαρμογή λογισμικού, όπου δίνεται η δυνατότητα όλοι οι χρήστες να χρησιμοποιούν τις εφαρμογές που διατίθενται στο Υπολογιστικό Νέφος.
- Πλατφόρμα Νέφους ως υπηρεσία (PaaS). Αποτελεί μια πλατφόρμα όπου εντός οι προγραμματιστές μπορούν να αναπτύξουν τις εφαρμογές τους. Οι χρήστες είναι σε θέση να χρησιμοποιούν τις εφαρμογές που έχουν αναπτυχθεί από τον ίδιο η χρησιμοποιώντας κάποιο εργαλείο που παρέχεται από τον πάροχο.
- Υποδομή Νέφους ως υπηρεσία (IaaS). Πρόκειται για μια δυνατότητα η οποία διανέμει στους χρήστες μηχανές αποθήκευσης, πόρους δικτύου, τις οποίες ο χρήστης μπορεί να αναπτύξει και να τρέξει το λογισμικό. [37]

Μοντέλα ανάπτυξης του υπολογιστικού νέφους θεωρούνται:

- Δημόσιο Νέφος (Public Cloud). Το δημόσιο νέφος είναι διαθέσιμο σε όλο το κοινό, αυτή η ιδιότητα του το κάνει πολλές φορές λιγότερο ασφαλές από άλλα μοντέλα νέφους. Συνήθως ανήκει και επομένως διατίθεται από μεγάλους οργανισμούς ή από εταιρείες που προσφέρουν υπηρεσίες νέφους.
- Ιδιωτικό Νέφος (Private Cloud). Αυτού του τύπου νέφους είναι διαθέσιμο σε έναν οργανισμό ή χρησιμοποιείται από έναν από τους πελάτες του οργανισμού. Προσφέρει ασφάλεια με μεγάλο κόστος.
- Νέφος Κοινότητας (Community Cloud). Αυτού του τύπου νέφους είναι διαθέσιμο σε δύο η περισσότερους οργανισμούς. Υποστηρίζει μια συγκεκριμένη ομάδα με κοινά ενδιαφέροντα και ανάγκες (σχολεία, πανεπιστήμια),

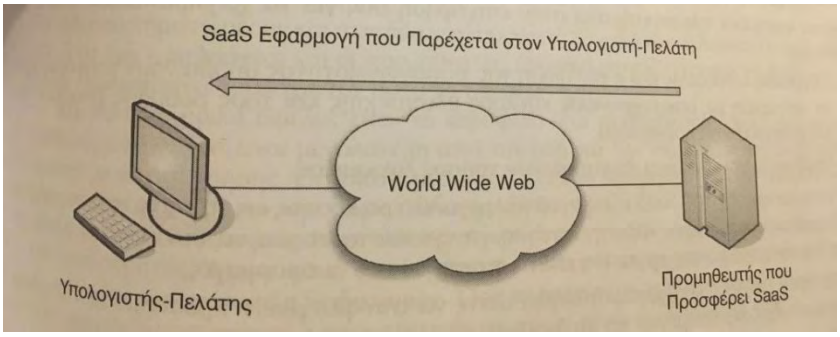
- Υβριδικό Νέφος (Hybrid Cloud). Αποτελείται από δύο ή περισσότερες διαφορετικές μορφές νέφους όπως ο συνδυασμός ιδιωτικού νέφους, δημόσιου νέφους ή νέφους κοινότητας. [37],[38]

2.3 Μοντέλα υπηρεσίας των Cloud Computing συστημάτων

Σε αυτή την ενότητα θα αναλυθούν εκτενώς τα Μοντέλα Υπηρεσίας Νέφους SaaS, IaaS, PaaS, IDaaS που παρουσιάσαμε παραπάνω στο δεύτερο επίπεδο αρχιτεκτονικής.

2.3.1 Λογισμικό Νέφους ως υπηρεσία (SaaS)

Το SaaS (Software as a Service) είναι μια εφαρμογή που φιλοξενείται σε έναν απομακρυσμένο πάροχο και προσπελάζεται μέσω Διαδικτύου από έναν υπολογιστή πελάτη, οι χρήστες δεν αγοράζουν ένα πακέτο λογισμικού και μια άδεια χρήσης πληρώνοντας μια φορά αλλά οι εφαρμογές παρέχονται μέσω συνδρομής ανάλογα με τους πόρους που καταναλώνονται ανά χρήση. Το μοντέλο SaaS που παρουσιάζεται και στην εικόνα 3, είναι ένα όλο και πιο διαδεδομένο μοντέλο παράδοσης καθώς αποτελεί μια αρχιτεκτονική η οποία είναι προσανατολισμένη προς τις υπηρεσίες SOA (Service-Oriented Architecture), στην οποία υποστηρίζονται μέθοδοι για ανάπτυξη εφαρμογών με λύσεις που διαθέτουν υπηρεσίες ιστού. Επιπρόσθετα το μοντέλο SaaS είναι παρόμοιο με το μοντέλο ASP (Application Service Provider), όπου ο πάροχος παραδίδει το λογισμικό στους χρήστες μέσω του διαδικτύου. [35], [36]



Εικόνα 3: Μοντέλο SaaS

Το μοντέλο SaaSμπορεί να κατηγοριοποιηθεί σε δύο υπηρεσίες. Η πρώτη κατηγορία αφορά τις επιχειρησιακές υπηρεσίες που αποτελούν επιχειρηματικές λύσεις που προσφέρονται σε επιχειρήσεις αλλά και σε εταιρείας και πωλούνται μέσω μια υπηρεσίας συνδρομής. Η δεύτερη κατηγορία αφορία τις υπηρεσίες που είναι προσανατολισμένες στους πελάτες και προσφέρονται στο ευρύ κοινό μέσω συνδρομής ή δωρεάν εάν έχουν κέρδη από διαφημίσεις.[35],[36]

Το δοθέν μοντέλο παρουσιάζει ποικίλα πλεονεκτήματα:

- Παρέχεται ανάκαμψη μετά από καταστροφές.
- Μειώνεται η ανάγκη για διαχείριση από το site του κάθε πελάτη αλλά από κεντρικά σημεία επιτρέποντας στους πελάτες να έχουν πρόσβαση σε εφαρμογές εξ αποστάσεως μέσω του διαδικτύου.
- Μειώνεται η ανάγκη για διαχειριστές στις εφαρμογές, συνήθως την ευθύνη την έχει ο πάροχος του μοντέλου SaaS.
- Μείωση του χρόνου για ενεργοποίηση ενός νέου συστήματος η εφαρμογής σε σχέση με τα παραδοσιακά λογισμικά. Το μοντέλο SaaS απαιτεί μόνο έναν web server.
- Προσφέρεται συνήθως δοκιμαστική περίοδος του παρόχουSaaS ώστε ο πελάτης αν είναι ικανοποιημένος να ξεκινήσει την μετακίνηση δεδομένων.
- Υπάρχει μικρό κόστος εκκίνησης. Το μοντέλο SaaS έχει χαμηλό κόστος χρήσης. Επομένως η χρησιμοποίηση μιας λύσης SaaS είναι πολύ πιο οικονομική από το να εγκατασταθεί σύνθετο λογισμικό.
- Υπάρχει υποστήριξη από τον προμηθευτή του μοντέλου SaaS.
- Αποτελεί μια λιγότερο επικίνδυνη επιλογή από το παραδοσιακό λογισμικό που εγκαθίσταται τοπικά. Ο χρηματικός κίνδυνος μειώνεται.
- Η ασφάλεια στις υπηρεσίες υπολογιστικού νέφους είναι αμφιλεγόμενη. Θεωρείται ότι στις υπηρεσίες «σύννεφου» υπάρχει μειωμένη ασφάλεια αν μετακινηθεί το «σύννεφο». Ωστόσο οι περισσότεροι προμηθευτές SaaS καταλαβαίνουν την ανάγκη για προστασία των δεδομένων και για την τακτική αντιγραφή τους και παρέχουν αρκετό προσωπικό ώστε να εξασφαλιστεί ότι τα δεδομένα είναι ασφαλή.
- Απαιτείται μειωμένο κεφάλαιο για έξοδα καθώς δεν υπάρχει ανάγκη για αγορά υλικού και λογισμικού.

- Ανταποκρίνεται σε βραχυπρόθεσμες ανάγκες, αντί να αγοραστεί νέο υλικό για να αντιμετωπιστούν επιπλέον απαιτήσεις, είναι δυνατό ένας πάροχος να προσφέρει άμεσα περισσότερους πόρους με το αντίστοιχο κόστος.
- Οι προμηθευτές λαμβάνουν υπ' όψιν τους τις απόψεις και τις ανάγκες των πελατών, ώστε να είναι ευχαριστημένοι και να αποκριθούν εάν είναι απαραίτητο και εφικτό.

Ωστόσο τα οφέλη του SaaS μπορούν να επικεντρωθούν στον καταναλωτή αφού δεν υπάρχει από εγκατάσταση και συντήρηση λογισμικού από πλευράς του. Η εγκατάσταση γίνεται από την υπηρεσία σε συντομότερο χρόνο και παρέχονται αναβαθμίσεις. Επιπλέον υπάρχει παγκόσμια διαθεσιμότητα της υπηρεσίας και τα δεδομένα είναι συμβατά σε ολόκληρη την επιχείρηση. Τέλος υπάρχει συμφωνία επιπέδου υπηρεσιών (SLA) ώστε εάν υπάρξουν προγραμματιστικά λάθη ο προμηθευτής θα τα διορθώσει με έναν τρόπο διαφανή. [39]

Ωστόσο υπάρχουν οφέλη και για τους προμηθευτές του υπολογιστικού νέφους εκτός από τα χρήματα. Πιο συγκεκριμένα ένα όφελος είναι το λειτουργικό περιβάλλουν όπου ο πάροχος έχει το δικό του τομέα. Η προβλέψιμη ροή των εσόδων, διότι οι συνδρομές των καταναλωτών είναι συγκεκριμένες. Το SaaS μπορεί να μελετηθεί ώστε να χρησιμοποιηθεί. Επίσης αναπτύσσονται ισχυρές σχέσεις με τους πελάτες διότι οι υπηρεσίες είναι βασισμένες στη συνδρομή. Τέλος τα λάθη που δημιουργούν μπορούν να βελτιωθούν με σταδιακές βελτιώσεις.

Σίγουρα ποτέ δεν υπάρχουν μόνο πλεονεκτήματα αλλά και μειονεκτήματα. Πιο συγκεκριμένα τα μειονεκτήματα που εντοπίζονται στο συγκεκριμένα μοντέλο αφορούν τη δυσκολία που εντοπίζεται στο σχεδιασμό της εφαρμογής, εάν υπάρχουν πολλαπλές μισθώσεις. Επίσης στην εξάρτηση των επιχειρήσεων με εξωτερικούς προμηθευτές για τη παροχή λογισμικού. Τέλος εμφανίζεται δυσκολία στην αλλαγή της επιχείρησης από εταιρεία λογισμικού σε εταιρεία υπηρεσίας.

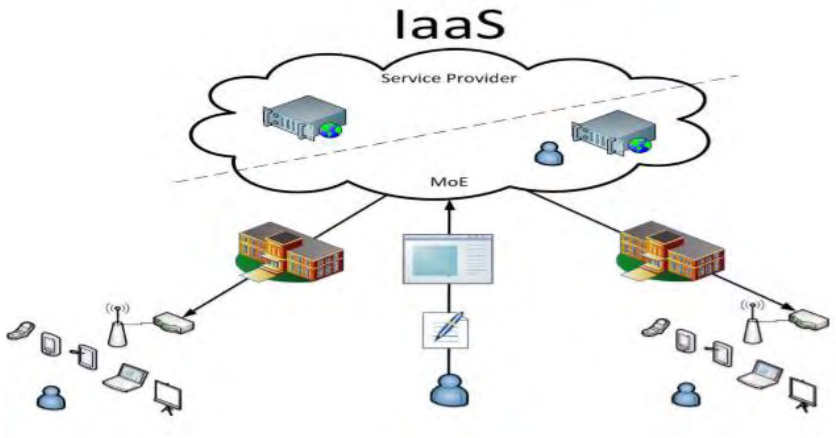
Έπειτα από τη δημιουργία του μοντέλου SaaS η εταιρεία Microsoft δημιούργησε το λογισμικό Software plus Services-S+S αξιοποιώντας τις δυνατότητες της υπηρεσίας SaaS αλλά με εναλλακτικές μεθόδους. Το δοθέν λογισμικό τρέχει τοπικά και η ύπαρξη το «σύννεφου» υπάρχει για επιπρόσθετες δυνατότητες. Το λογισμικό αυτό γνώρισε μεγάλη επιτυχία λόγω το ότι υπάρχει το «σύννεφο» και τα

οφέλη που προσφέρει αλλά συγχρόνως παρέχεται η ασφάλεια αφού τα δεδομένα υπάρχουν εντός της επιχείρησης του πελάτη.

2.3.2 Υποδομή Νέφους ως υπηρεσία (IaaS)

Το IaaS (Infrastructure as a Service) αποτελεί έναν εικονικό διακομιστή (πλατφόρμα εικονικοποίησης περιβάλλοντος) και τρέχετε εξειδικευμένο λογισμικό σ' αυτόν. Στο μοντέλο IaaS εκμεταλλεύονται οι υπηρεσίες, η τεχνολογία και τα δεδομένα ώστε να δοθούν ως πόρους υλικού υπολογιστών σε ένα πακέτο υπηρεσιών στους πελάτες . Σε αντίθεση με τα παραδοσιακά μοντέλα διανομής λογισμικού, το IaaS δεν διαθέτει χρονοβόρες διαδικασίες αλλά είναι προσανατολισμένο και τυποποιημένο γύρω από τις ανάγκες των πελατών.

Οι πάροχοιIaaS είναι υπεύθυνοι για την φιλοξενία εφαρμογών ενώ οι πελάτες χρειάζεται οι ίδιοι να διαχειριστούν το λογισμικό και το λειτουργικό σύστημα και αναλάβουν την ενημέρωση τους. Πιο συγκεκριμένα οι πελάτες της υπηρεσίας IaaS χρεώνονται μόνο για τους πόρους που έχουν καταναλώσει και δεν προμηθεύονται οι ίδιοι το λογισμικό, το εξυπηρετητές, τα δεδομένα , τον εξοπλισμό (κλιματισμός, πυροσβεστήρας). Επομένως η κάθε εταιρεία δεν χρειάζεται να αγοράσουν και να διατηρήσουν δικιά της εγκατάσταση (κέντρο δεδομένων). Η αρχιτεκτονική του μοντέλου IaaS παρουσιάζεται στην εικόνα 4. [35]



Εικόνα 4: Μοντέλο IaaS

Προκειμένου να εφαρμοστεί το μοντέλο IaaS, απαιτείται όσο αφορά τον εξοπλισμό η ύπαρξη κλιματισμού για την εξάλειψη θερμότητας των περιφερειακών συσκευών (διακομιστές, δίσκοι), γεννήτριες σε περίπτωση εφεδρική λύσης, σύστημα

για τη καταστολή πυρκαγιάς και κέντρο δεδομένων. Επίσης θα πρέπει να υπάρχει πρόσβαση σε υψηλές ταχύτητες εντός το διαδικτύου. Τέλος όσο αφορά το προσωπικό θα πρέπει να είναι κατάλληλο για την υποστήριξη θεμάτων υλικού, δικτύου και λειτουργικού συστήματος. [36]

Το δοθέν μοντέλο παρουσιάζει ποικίλα πλεονεκτήματα:

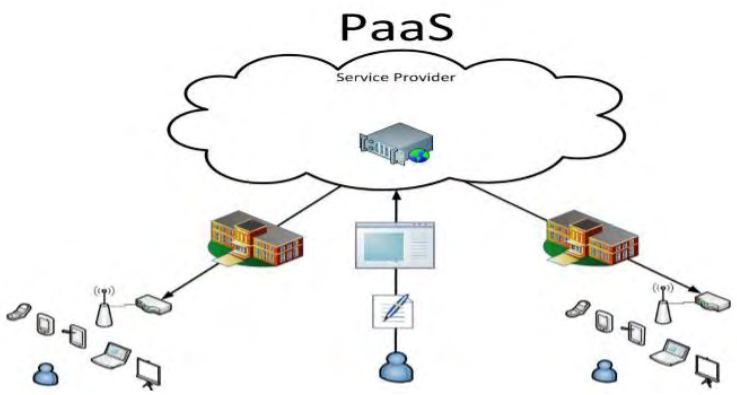
- Εύκολη επέκταση υλικού λόγω του μειωμένου χρόνου που προσφέρουν τα νέα χαρακτηριστικά της υπηρεσίας.
- Μειωμένο κόστος υλικού διότι δεν απαιτούνται υψηλά κεφάλαια για την απόκτηση εξοπλισμού.
- Οι πελάτες πληρώνουν ανάλογα με την χρήση της υπηρεσίας.
- Χρησιμοποιείται τελευταία τεχνολογία.
- Παρέχετε ασφάλεια στις πλατφόρμες.
- Γίνονται δοκιμές υπηρεσίας σε κατάλληλο περιβάλλον.
- Ύπαρξη ολοκληρωμένου περιβάλλοντος υποστήριξης και διαχείρισης.
- Δίνεται η δυνατότητα διαχείρισης της υπηρεσίας.

Ωστόσο εντοπίζονται μειονεκτήματα που αφορούν τον κίνδυνο στο να διαρρεύσουν τα δεδομένα στους πελάτες. Υπάρχει μειωμένη ασφάλεια εάν χρησιμοποιείται παλιά έκδοση λογισμικού. Τέλος δεν μπορεί να υπάρξει πρόβλεψη συμπεριφοράς του εικονικού διακομιστή από τους πελάτες διότι είναι ήδη προ σχεδιασμένος. [39]

Με βάση το μοντέλο IaaS, χρησιμοποιήθηκε η υπηρεσία On-Demand και γίνεται όλο και πιο διαδεδομένο διότι οι πόροι δίνονται από τους παρόχους στους πελάτες σύμφωνα με τις ανάγκες τους. Οι πιο διαδεδομένες υπηρεσίες είναι η Amazon Elastic Compute Cloud (EC2) και η Go Grid. Η EC2 αποτελεί μια υπηρεσία διαδικτύου που προσφέρει υπολογιστική χωρητικότητα προσαρμόζοντας την, ανάλογα με τις ανάγκες των πελατών στο «σύννεφο». Επιπρόσθετα οι η Amazon δίνει την δυνατότητα στους πελάτες να τρέχουν το λογισμικό τους στην πλατφόρμα της. Η Go Grid αποτελεί μια υπηρεσία όπου φιλοξενούνται διακομιστές Linuxαλλά και Windows. Οι εκδόσεις που προσφέρονται σε Windows Server 2008 είναι σε 32-bit και 64-bit αποτελώντας αρχικό πάροχο σε υπηρεσία IaaS. Είναι ευρέως διαδεδομένο λόγο της ασφάλειας, σταθερότητας και γρήγορης εγκατάστασης του Windows Server 2008. [39]

2.3.3 Πλατφόρμα Νέφους ως υπηρεσία (PaaS)

Το μοντέλο PaaS (Platform as a Service) δίνει τη δυνατότητα στους πελάτες του να δημιουργήσουν τις δικές τους εφαρμογές και να τοποθετηθούν στο «σύννεφο», δίνοντας τους πόρους υλικού και λογισμικού (εικόνα 5). Επομένως εξαλείφεται η ανάγκη αγοράς διακομιστών από τους πελάτες, ώστε να φιλοξενηθούν οι εφαρμογές τους. Οι εταιρείες πληρώνουν μόνο για τους πόρους που έχουν καταναλώσει. Πολλοί συγχέουν την υπηρεσία PaaS με την υπηρεσία SaaS που παρέχει μόνο τον εξοπλισμό για να φιλοξενηθούν οι εφαρμογές που δημιουργούνται μέσω της υπηρεσίας PaaS. [35], [36], [39]



Εικόνα 5: Μοντέλο PaaS

Τα πλεονεκτήματα PaaS εντοπίζονται παρακάτω:

- Χαμηλό κόστος διοικητικών εξόδων, η υπηρεσία PaaS διαθέτει υπαλλήλους διοίκησης.
- Χαμηλό κόστος για την αγορά υλικού (διακομιστές, βάσεις δεδομένων), αφού δεν απαιτείται η απόκτηση εξοπλισμού από τις εταιρείες.

- Ύπαρξη σύγχρονου λειτουργικού συστήματος, η υπηρεσία PaaS αναλαμβάνει για την σωστή και έγκυρη ενημέρωση του.
- Οι εταιρείες μπορούν να αφοσιωθούν σε θέματα εσωτερικών πολιτικών και όχι σε θέματα υλικού και εξοπλισμού.
- Η πληρωμή από τους πελάτες γίνεται ανάλογα με τις ανάγκες τους, σύμφωνα με όσους πόρους έχουν καταναλώσει.

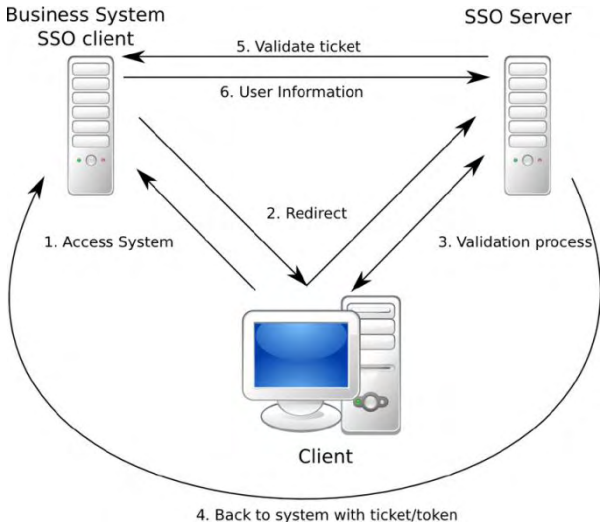
Ωστόσο παρά τα πλεονεκτήματα παρουσιάζεται μειωμένη ασφάλεια δεδομένων που υπάρχουν αποθηκευμένα στο «σύννεφο», αλλά και προβλήματα στη συμβατότητα, εάν η εφαρμογή είναι παλιού τύπου και δεν έχει ενημερωθεί. Τέλος η υπηρεσία PaaS δεν μπορεί να υποστηρίξει τους πελάτες της σωστά, οι εφαρμογές θα υποστούν βλάβες ειδικά κατά την μετακίνηση τους. Συμπερασματικά τα κύρια προβλήματα που μπορούν να υπάρξουν είναι ασφάλειας, διαθεσιμότητας και απόδοσης. [35],[36], [39]

2.3.4 Ταυτότητα Νέφους ως υπηρεσία (IDaaS)

Με την εξέλιξη της τεχνολογίας οι ποικίλες εργασίες μπορούν να διεκπεραιωθούν με την πρόσβαση σε πολλαπλά συστήματα. Τα συστήματα μπορεί να αφορούν την ύπαρξη τους σε «σύννεφο», ή να βρίσκονται τοπικά. Κάτι τέτοιο απαιτεί από τους χρήστες να απομνημονεύουν πολλά στοιχεία λογαριασμών. Στις εταιρείες η ύπαρξη πολλών λογαριασμών δεν είναι πάντα εύκολη, διότι όταν κάποιο άτομο από το προσωπικό τους παύει να εργάζεται στην εταιρεία ταυτόχρονα πρέπει να διαγραφούν και όλοι οι λογαριασμοί του για θέματα ασφάλειας. Αξίζει να τονιστεί ότι η διαχείριση ταυτότητας αποτελεί μια ακριβή, απαιτητική και μακροπρόθεσμη διαδικασία.

Η λύση στους πολλούς λογαριασμούς τω εφαρμογών που βρίσκονται σε ποικίλους διακομιστές είτε «σύννεφου» είτε τοπικούς στην υπηρεσία IDaaS είναι το λογισμικό SingleSign-On (SSO) η δομή του εμφανίζεται στην εικόνα 6 . Το δοθέν λογισμικό διευκολύνει τους χρήστες με το να συνδέονται μόνο μια φορά για όλες τις εφαρμογές που χρησιμοποιούν.

Το εξελιγμένο αυτό μοντέλο παρουσιάζει πλεονεκτήματα που εντοπίζονται στους λιγότερους λογαριασμούς και συνεπώς στα λιγότερα στοιχεία σύνδεσης που απαιτούνται να απομνημονεύσουν οι χρήστες όπως λιγότεροι κωδικοί πρόσβασης. Επίσης ο χρόνος χρήστη στις διάφορες εφαρμογές είναι μικρότερος. Τέλος τα τμήματα IT, έχουν μικρότερο φόρτο εργασίας στην εξυπηρέτηση πελατών που έχουν ξεχάσει τα στοιχεία σύνδεσης τους (username, password). Ωστόσο αν το λογισμικό SSO υποστεί βλάβη και δεν λειτουργεί, τότε οι χρήστες δεν θα έχουν πρόσβαση σε κανέναν διακομιστή και επομένως σε καμία εφαρμογή. [36]



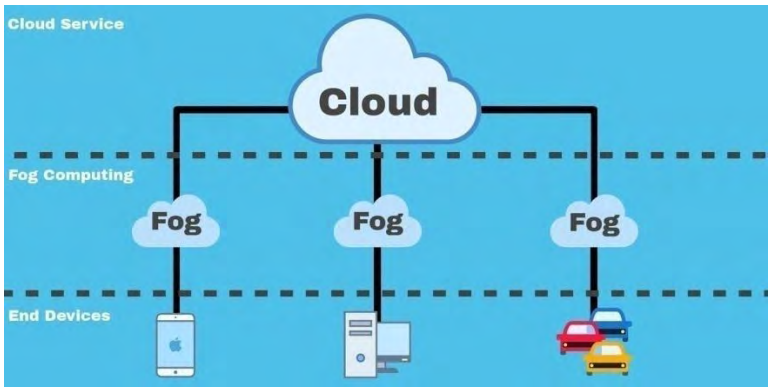
Εικόνα 6: Μοντέλο SSO

Κεφάλαιο 3: Μοντέλο Υπολογιστικής Ομίχλης

3.1 Εισαγωγή

Το μοντέλο υπολογιστικής ομίχλης αποτελεί επέκταση του μοντέλου υπολογιστικού νέφους λόγω της ανάπτυξης της κινητής τηλεφωνίας και του cloud computing. Δημιουργήθηκε για να αντιμετωπίσει τα προβλήματα που εμφάνισε το υπολογιστικό νέφος, με την δυσκολία διεκπεραίωσης αιτήσεων λόγω γεωγραφικών περιορισμών, ασφάλειας, προστασίας προσωπικών δεδομένων και λανθασμένης πρόβλεψης της υπάρχουσας κατάστασης . Όμοια με το μοντέλο υπολογιστικού νέφους παρέχει υπολογιστικούς πόρους υπηρεσίες δεδομένων, αποθήκευσης και εφαρμογές σε πελάτες μέσω δικτύου. Η κύρια διαφορά τους είναι ότι επεξεργάζονται μεγάλο όγκο δεδομένων τοπικά, λειτουργώντας σε φορητές συσκευές από ετερογενή υλικά σε μεγάλες γεωγραφικές αποστάσεις. Το μοντέλο υπολογιστικής ομίχλης είναι προσιτό στους τελικούς χρήστες, αποτελώντας μια κατανεμημένη και εικονικοποιημένη πλατφόρμα. Πολλές φορές χρησιμοποιείται ο όρος edge computing αντί για fog computing.

Πιο συγκεκριμένα, η εταιρεία Cisco υλοποίησε το μοντέλο υπολογιστικής ομίχλης ως ιδανική πλατφόρμα για την υλοποίηση των στόχων της τεχνολογίας Internet of Thing (IOT) . Έδωσε τη δυνατότητα σε ένα τεράστιο ποσοστό συνδεδεμένων συσκευών στο διαδίκτυο να διαχειριστούν και να εκτελέσουν εφαρμογές λογισμικού εντός του δικτύου Cisco Iox. Το Cisco Iox αποτελείτε από λειτουργικό σύστημα Cisco IOS και από λογισμικό ανοικτού κώδικα Linux. [42]



Εικόνα 7: Η Αρχιτεκτονική του μοντέλου Fog Computing

3.2 Χρήση του Fog Computing

Η χρήση του μοντέλου υπολογιστικής ομίχλης εντοπίζετε σε ποικίλες εφαρμογές.

- Ροη βίντεο- Video Streaming: Οι εφαρμογές που διαθέτουν υπηρεσίες βίντεο μπορούν να λειτουργήσουν καλύτερα μέσω της υπολογιστικής ομίχλης λόγω των πλεονεκτημάτων που παρουσιάζει στην αλλαγή τοποθεσίας, μειωμένη λανθάνουσα κατάσταση και στην επεξεργασία σε πραγματικό χρόνο. Πολλές έξυπνες συσκευές δίνουν την δυνατότητα στους χρήστες να παρακολουθήσουν ένα βίντεο σε ζωντανή ροή-live streaming. Η υπηρεσία υπολογιστικής ομίχλης μπορεί να βελτιώσει την αλληλεπίδραση των χρηστών με την εικονική υποδοχή όπου παρουσιάζετε το βίντεο σε πραγματικό χρόνο από μια κάμερα που καταγράφει το γεγονός. [18]
- Ηλεκτρονικά παιχνίδια-Gaming: Το cloud computing δημιούργησε μια καινοτόμα πλατφόρμα όπου οι παίκτες ηλεκτρονικών παιχνιδιών δεν χρειάζεται να διαθέτουν εξειδικευμένο υλικό στην ηλεκτρονική συσκευή τους. Πιο συγκεκριμένα επιτρέπετε στους πελάτες να κατεβάσουν τα πλήρη παιχνίδια με αντάλλαγμα ένα προκαθορισμένο ποσό μέσω διαδικτύου, λεγόμενη υποδομή game on demand. Το παιχνίδι μπορεί να αποκτηθεί σε ηλεκτρονικούς υπολογιστές αλλά και σε κινητές συσκευές όπως κινητά τηλέφωνα και tablet. Σε κινητές συσκευές ο μεγαλύτερος φόρτος εργασίας στο παιχνίδι ανατίθεται σε διακομιστές ομίχλης, όπου από τους διακομιστές αυτούς λαμβάνει η κινητή συσκευή τα δεδομένα του παιχνιδιού. Το cloud computing υστερεί στο να ανταποκριθεί σε αυτό το είδος παιχνιδιών λόγο του μειονεκτήματος που παρουσιάζει στη καθυστερημένη απόκριση. Μπορούμε να καταλάβουμε τη τεχνολογία των ηλεκτρονικών παιχνιδιών στην εικόνα 9. [18]
- Τομέας υγείας-Healthcare: Οι υπηρεσίες υγείας κατάφεραν να αναπτυχθούν μέσω των εφαρμογών διαδικτύου σε συσκευές ομίχλης που επεξεργάζονται τα δεδομένα σε πραγματικό χρόνο πριν σταλούν στους διακομιστές σύννεφου . Καινοτόμες ανακαλύψεις που πραγματοποιήθηκαν μέσω του μοντέλου υπολογιστικής ομίχλης αφορούν το καρδιογράφημα, τη πρόβλεψη

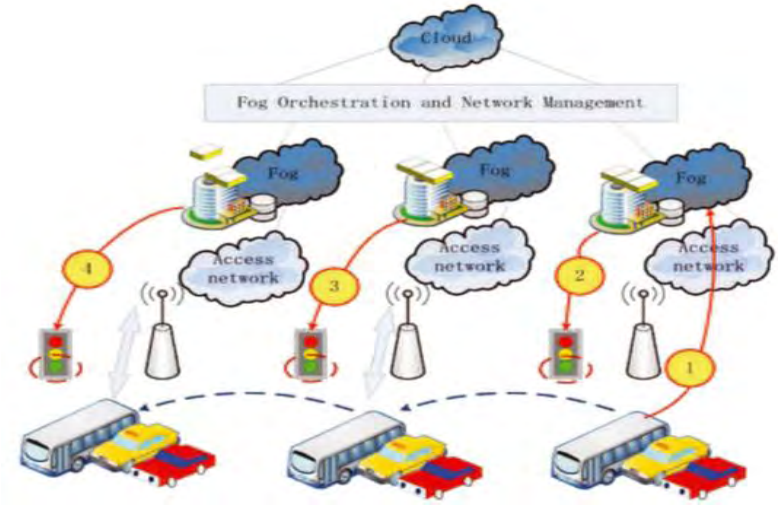
εγκεφαλικού επεισοδίου αλλά και την ενημέρωση του τμήματος υγείας με αποστολή της θέσης του ασθενούς. [18]

- Συστήματα έξυπνου φωτισμού-Smart traffic light system:Όπως βλέπουμε και στην εικόνα 8, τα συστήματα έξυπνου φωτισμού λειτουργούν με τη χρήση αισθητήρων που ανιχνεύουν την ύπαρξη πεζών και ποδηλατών, εκτιμάνε τη ταχύτητα και τις αποστάσεις των οχημάτων και μπορούν να προβλέψουν και να αποτρέψουν ατυχήματα στέλνοντας κατάλληλα σήματα. Ο φωτισμός ανάβει όταν αναγνωριστεί κίνηση και απενεργοποιείται όταν δεν υπάρχει κίνηση. Τα έξυπνα φώτα λειτουργούν ως συσκευές ομίχλης συνδεδεμένος στο δίκτυο μέσω WIFI ή 3G.

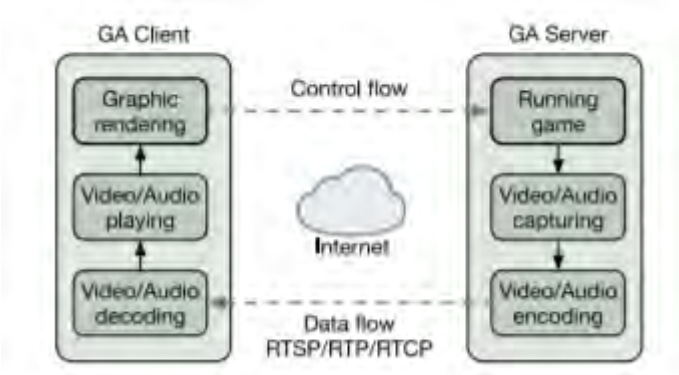
  Μια σημαντική χρήση του φωτισμού αυτού είναι όταν εντοπιστεί ένα όχημα έκτατης ανάγκης όπως ασθενοφόρο να αλλάξει ο φωτισμός ώστε να απελευθερωθεί μια λωρίδα για να περάσει το όχημα αυτό. [18], [31]
- Έξυπνο πλέγμα-Smart Grid: Οι ανάγκες για ενέργεια με την πάροδο του χρόνου έχουν αυξηθεί για το λόγο αυτό δημιουργήθηκαν τα δίκτυα έξυπνων πλεγμάτων. Οι συσκευές αυτές μετατρέπονται σε εναλλακτικές μορφές ενέργειας όπως ηλιακή ή αιολική ενέργεια, λειτουργούν με βάση το μοντέλο υπολογιστικής ομίχλης συλλέγοντας δεδομένα από αισθητήρες και στέλνονται για ανάλυση σε ανώτερη βαθμίδα όπου είναι το σύννεφο σε πραγματικό χρόνο Ο στόχος της εφεύρεσης αυτής είναι να περιοριστούν οι εκπομπές αερίου και να μην αλλάξει περαιτέρω το κλίμα του περιβάλλοντος. Το έξυπνο πλέγμα μπορεί να κατανοηθεί καλύτερα αν ανατρέξουμε στη εικόνα 10. [18],[31]
- Έξυπνες πόλεις-Smart Cities: Οι έξυπνες πόλεις βασίζονται στη διαχείριση της ενέργειας, των κτηρίων, των οδικών μεταφορών μέσω διαδικτύου βελτιώνοντας τις συνθήκες ζωής των κατοίκων. Οι έξυπνες πόλεις αποτελούν σημείο ενδιαφέροντος λόγω της ταχείας αστικής ανάπτυξης. Ωστόσο μπορούν να υπάρξουν ποικίλα προβλήματα αν δεν υπάρχει κατάλληλος εξοπλισμός και ατόμων καταρτισμένων με ικανότητες διαχείρισης και υπολογισμών. Η υλοποίηση μιας έξυπνης πόλης βασίζετε σε υπολογιστές ομίχλης με ποικίλες υποδομές και υπηρεσίες. [18]
- Έξυπνα οχήματα: Το νέφος χρησιμοποιείται στο όχημα μέσω εφαρμογής για να υπάρξει πιο ασφαλής οδήγηση προβλέποντας τη κυκλοφοριακή συμφόρηση αλλά και τα ατυχήματα μέσω αισθητήρων. [18]

- Δίκτυα βιομηχανικών ασύρματων αισθητήρων: Σε βιομηχανίες μεγάλης κλίμακας η τεχνολογία είναι άρρηκτα συνδεδεμένη. Μπορούν να υπάρξουν έξυπνα εργοστάσια με δίκτυα αισθητήρων που βοηθούν στην καλύτερη αυτοματοποιημένη παραγωγή, παράγοντας μια πολύ μεγάλη ποσότητα δεδομένων σε πραγματικό χρόνο αλλά και στην ασφάλεια της βιομηχανίας. Το μοντέλο υπολογιστικής ομίχλης εξαλείφει τις καθυστερήσεις και τη συμφόρηση στο δίκτυο και μειώνει τα συνολικά έξοδα της. [41]
- Έξυπνα κτήρια: Οι εφαρμογές επικεντρώνεται σε μετρήσεις που γίνονται στο κτήριο μέσω ασύρματων αισθητήρων όπως η μέτρηση της θερμοκρασίας, των αερίων, της υγρασίας. Όλες οι πληροφορίες μπορούν να ενοποιηθούν και να προκύψουν χρήσιμα στοιχεία για τη πολυκατοικία από τις μετρήσεις αυτές. Τα συστήματα νέφους είναι ικανά να αντιμετωπίσουν πιθανά προβλήματα που μπορούν να προκύψουν χαμηλώνοντας τη θερμοκρασία, μείωση της υγρασίας και των αερίων. Επιπρόσθετα οι αισθητήρες είναι ικανοί να ανιχνεύσουν κινήσεις. Συμπερασματικά τα έξυπνα κτήρια μέσω της υπολογιστικής ομίχλης μπορούν να εξοικονομήσουν πόρους και προσφέρουν να ένα καλύτερο και πιο ασφαλές περιβάλλον στους κατοίκους. [31]
- Συστήματα IoTκαι CPSs: Με την εξέλιξη του διαδικτύου και των τηλεπικοινωνιών τα συστήματα IoTείναι χρήσιμα αφού μπορούν να συνδέσουν φυσικές συσκευές με ηλεκτρονικές διευθύνσεις. Τα συστήματα CPS- ενσωματώνουν υπολογιστικά στοιχεία του συστήματος με φυσικά. Στόχος των συστημάτων αυτών είναι μέσω της υπολογιστικής ομίχλης να τροποποιηθεί η λειτουργία διάφορων συσκευών με την ένταξη κατάλληλου λογισμικού και προγραμμάτων. Παράδειγμα τέτοιων συσκευών είναι παιχνίδια, μεταφορικά μέσα, ιατρικά μηχανήματα. [31]
- Δίκτυα SDN: Το δοθέν μοντέλο δικτύωσης είναι ευρέος πλέον διαδεδομένα. Η χρήση του SDN- Software Defined Network εντοπίζεται στο έξυπνα οχήματα που έχουν αναφερθεί παραπάνω προκειμένου να αποτρέψουν συγκρούσεις να δημιουργήσουν μια δομή επικοινωνίας για τα οχήματα και για να αποτελέσουν το δίκτυο τους. Εφαρμόστηκαν πάνω σε δίκτυα WLAN και σε ασύρματα δίκτυα αισθητήρων. [31]
- Έξυπνη διαχείριση ενέργειας: Το ηλεκτρικό δίκτυο είναι αναγκαίο με την παροχή ρεύματος επομένως πρέπει να υπάρχει μια σωστή διαχείριση της

ηλεκτρικής ενέργειας ώστε να εξισορροπηθεί ο φόρτος παραγωγής αλλά και κατανάλωσης τόσο στους κατοίκους των πόλεων όσο και στις επιχειρήσεις. Το Fog Computing σε συστήματα διαχείρισης ενέργειας επεξεργάζονται τα δεδομένα εξαρχής και ικανοποιούν τις απαιτήσεις ενέργειας. Αποτελείται από αισθητήρες οι οποίοι ελέγχουν την κατανάλωση ενέργειας και την υπολογιστική συσκευή που κάνει τις επεξεργασίες των δεδομένων. Οι λειτουργίες της επικεντρώνονται στον έλεγχο της κατανάλωσης ισχύος παραδείγματος χάριν στην κατανάλωση ρεύματος που έχει μια επιχείρηση και στη διαχείριση της ενέργειας ώστε να μειωθεί η κατανάλωση της παραδείγματος χάριν έξυπνος φωτισμός. [41]



Εικόνα 8: Χρήση μοντέλου υπολογιστικής ομίχλης σε έξυπνα φώτα



Εικόνα 9: Χρήση μοντέλου υπολογιστικής ομίχλης σε ηλεκτρονικά παιχνίδια

Εικόνα 10: Χρήση μοντέλου υπολογιστικής ομίχλης σε έξυπνο πλέγμα

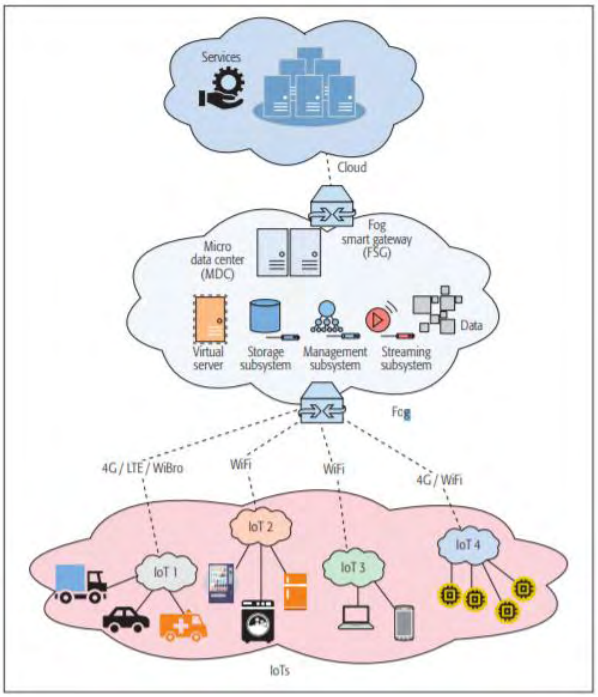
3.3 Τεχνολογία Fog Computing

Το μοντέλο υπολογιστικής ομίχλης αποτελεί μια κατανεμημένη αρχιτεκτονική. Αποτελείται από πολλούς κόμβους που παράγουν δεδομένα και τα μεταδίδουν από κατώτερες σε ανώτερες βαθμίδες. Οι κόμβοι ομίχλης-fognodes προκειμένου να παρέχουν πόρους, δικτύωση και να αποθηκεύσουν δεδομένα επικοινωνούν με τελικές συσκευές και με στοιχεία σύννεφου. Πιο συγκεκριμένα η αρχιτεκτονική είναι ιεραρχική αφού αποτελείται από τρείς βαθμίδες. Η πρώτη χαμηλότερη βαθμίδα ονομάζεται επίπεδο εικονικοποίησης και αποτελείται από κόμβους IoT δηλαδή από οποιαδήποτε συσκευή μπορεί να συνδεθεί στο διαδίκτυο οι λεγόμενες έξυπνες συσκευές. Η δεύτερη και μεσαία βαθμίδα από κόμβους ομίχλης όπως δρομολογητές πύλες, αισθητήρες, οχήματα και τέλος η τρίτη και ανώτερη βαθμίδα ονομάζεται cloud computing και περιλαμβάνει κέντρα δεδομένων και διακομιστές υψηλού επιπέδου.

Ωστόσο στην αρχιτεκτονική της υπολογιστικής ομίχλης υπάρχουν έξι βασικά στρώματα. Το χαμηλότερο στρώμα είναι το φυσικό επίπεδο- επίπεδο εικονικοποίησης

όπου αναλύθηκε παραπάνω. Έπειτα ακολουθεί το επίπεδο παρακολούθησης όπου ελέγχονται οι ενέργειες των κόμβων και καθορίζονται οι επόμενες ενέργειες τους . Πιο πάνω βρίσκετε το επίπεδο επεξεργασίας όπου αναλύει τα δεδομένα. Το στρώμα αποθήκευσης αναλαμβάνει την αποθήκευση των δεδομένων σε μοντέλα υπολογιστικής ομίχλης συνήθως βραχυπρόθεσμα, αφού για μακροχρόνια αποθήκευση ενδείκνυται η αποθήκευση σε σύννεφο. Την ασφάλεια και τη προστασία των προσωπικών δικαιωμάτων αναλαμβάνει το επίπεδο ασφάλειας διαμέσου του καναλιού όπου μεταδίδονται οι πληροφορίες , την μεταφορά την αναλαμβάνει το στρώμα μεταφοράς.

Η αρχιτεκτονική αυτή έχει ως αποτέλεσμα την εξάλειψη καθυστερήσεων και την υψηλή ποιότητα σε παρακολούθηση γεγονότων σε πραγματικό χρόνο ακόμη και σε κινητές συσκευές. [43],[18]



Εικόνα 11: Αρχιτεκτονική του μοντέλου υπολογιστικής ομίχλης

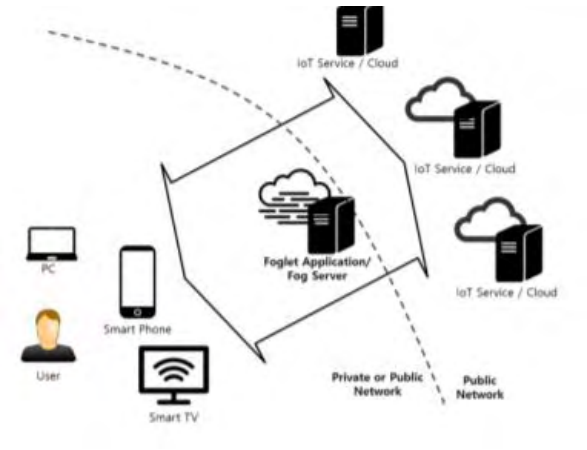
3.4 Αναγκαιότητα Fog Computing

Το cloud computing προσφέρει υπηρεσίες στους πελάτες πληρώνοντας μια σταθερή συνδρομή πριν την χρήση της υπηρεσίας γνωστό ως μοντέλο “pay-as-you-go”. Εξαλείφετε η ανάγκη των πελατών για γνώση πολλών τεχνικών λεπτομερειών. Ωστόσο μπορεί να υπάρξουν κίνδυνοι σε εφαρμογές που είναι επιρρεπής σε καθυστερήσεις με αυτή τη διαχείριση. Το Fog Computing λύνει τα προβλήματα που δημιουργούνται από λανθάνουσα κατάσταση και της αλλαγής της γεωγραφικής θέσης, βελτιώνοντας την ποιότητα στη χρήση υπηρεσιών σε εφαρμογές. Τα πλεονεκτήματα που παρουσιάζει είναι ποικίλα και παρουσιάζονται παρακάτω.

- Επεξεργασία μεγάλου όγκου δεδομένων σε πραγματικό χρόνο. Τα δεδομένα στέλνονται σε κόμβους ομίχλης που βρίσκονται κοντά στους τελικούς χρήστες.
- Υπάρχει μικρότερη καθυστέρηση από το υπολογιστικό νέφος σε υπηρεσίες που είναι επιρρεπής στις καθυστερήσεις.
- Γρήγορη και μεγάλη κλιμάκωση προσθέτοντας κόμβους ομίχλης σε υπολογιστές ομίχλης.
- Οι πόροι βρίσκονται τοπικά.
- Χαμηλή λανθάνουσα κατάσταση και καλύτερη εμπειρία χρήσης.
- Υπάρχει ετερογένεια στις τελικές συσκευές, switches, access points και routers. Η ετερογένεια είναι χρήσιμη διότι υπάρχουν διάφορες μορφές που μπορούν να υλοποιηθούν σε πολλά περιβάλλοντα.
- Προσφέρετε γεωγραφική κατανομή αφού το fog computing αποτελεί μια κατανεμημένη ανάπτυξη που έχει ως σκοπό την παροχή πόρων σε κινητές αλλά και σε ακίνητες συσκευές.
- Δίνει τη δυνατότητα στους χρήστες να παρακολουθήσουν στατιστικά εντός του δικτύου και να αλληλεπιδράσουν με το υπολογιστικό νέφος.
- Ύπαρξη διαλειτουργικότητας, οι διεπαφές του συστήματος συνεργάζονται για να υποστηρίξουν τις αλλαγές που παρουσιάζονται στις υπηρεσίες παραδείγματος χάριν η ροή δεδομένων. [31]

3.5 Συνεργασία και διεπαφές μεταξύ Fog Computing Cloud Computing και IoT.

Η συνεργασία του Fog Computing με συσκευές IoT έχει ως στόχο να αύξηση την απόδοση των συσκευών αυτών με την προσθήκη διακομιστών. Πιο συγκεκριμένα τοποθετούνται διακομιστές ομίχλης ανάμεσα στις εφαρμογές IoT και των υπηρεσιών IoT σύννεφου. Η μέθοδος αυτή μπορεί να γίνει πιο κατανοητή από την εικόνα 12.



Εικόνα 12: Συνεργασία Fog Computing ,Cloud Computing και IoT

Η συνεργασία που έχει το Fog Computing με τις υπηρεσίες IoT αφορούν την αποθήκευση δεδομένων. Ο διακομιστής ομίχλης αποκτά τους πόρους από κάθε υπηρεσία νέφους IoTκαι έπειτα τους αποθηκεύει. Οι διακομιστές ομίχλης έχουν πρόσβαση σε όλα τα δεδομένα του υπολογιστικού νέφους με εφαρμογές IoT. Επομένως όταν μια υπηρεσία IoT θέλει να αποκτήσει δεδομένα χρησιμοποιεί τους διακομιστές ομίχλης και μέσω αυτών έχει πιο γρήγορη πρόσβαση στα δεδομένα αυτά.

Επιπρόσθετα προκειμένου να υπάρξει πρόσβαση από μια μόνο εφαρμογή σε πολλές υπηρεσίες IoT η υλοποίηση της κάθε υπηρεσίας πρέπει να γίνει ξεχωριστά. Η υλοποίηση βασίζεται σε ένα ενιαίο σύνολο API για διακομιστές ομίχλης με σκοπό να μειωθούν τα έξοδα συντήρησης. Η υλοποίηση αυτή είναι επικεντρωμένοι σε

υπηρεσίες υπολογιστικού νέφους που βασίζονται στον έλεγχο δραστηριότητας και παρακολούθησης καιρού.

Σχετικά με τις διεπαφές-interfaces του fog computing και των υπηρεσιών IoT γνωρίζοντας ότι ένας διακομιστής ομίχλης βρίσκεται στις άκρες του ιδιωτικού και του δημόσιου δικτύου, αντιμετωπίζεται πρόβλημα στη μεταφορά δεδομένων εντός του ιδιωτικού δικτύου. Ο λόγος του προβλήματος είναι η χρήση IPV4. Η λύση στο πρόβλημα είναι ένας κόμβος με δύο interfaces ένα για το ιδιωτικό και ένα για το δημόσιο δίκτυο επιπρόσθετα και τα δύο δίκτυα έχουν interfaces για εφαρμογές και υπηρεσίες IoTβασισμένες στο σύννεφο. [45]

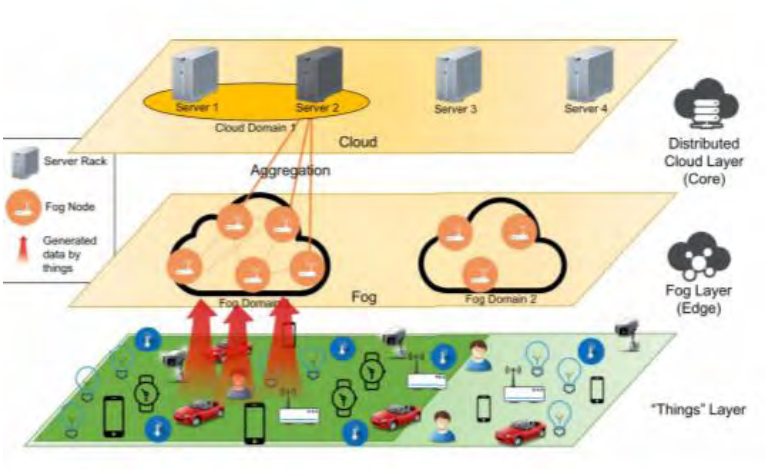
Με βάση την αρχιτεκτονική Fog, IoT και Cloud όπως έχει προαναφερθεί υπάρχουν τρία στρώματα ένα για τις συσκευές IoT, ένα για το fog computing για τους κόμβους του και ένα για το cloud computing όπου είναι τοποθετημένοι οι διακομιστές σύννεφου. Οι διακομιστές σύννεφου μπορεί να αποτελούνται από ένα σύνολο φυσικών διακομιστών ή από έναν διακομιστή με πολλαπλές μονάδες επεξεργασίας .Σε κάθε επίπεδο οι κόμβοι αποκτάνε domain ώστε να επικοινωνούν με άλλους σχετικούς κόμβους σε άλλο επίπεδο. Οι κόμβοι ομίχλης-fog nodestοποθετούνται σε μικρή απόσταση μεταξύ τους. Επομένως κάθε σύνολο κόμβων ομίχλης επικοινωνεί με ένα άλλο σύνολο διακομιστών ομίχλης και δημιουργείτε μια ενιαία εφαρμογή. Η επικοινωνία των κόμβων επιτυγχάνεται ως εξής οι κόμβοι IoT επιθυμούν την τοπική επεξεργασία. τα δεδομένα στέλνονται στους κόμβους ομίχλης με σκοπό την μείωση των καθυστερήσεων, γι’ αυτό το στρώμα ομίχλης βρίσκεται ανάμεσα στα άλλα δύο στρώματα αφού καταφέρνει να χειριστεί περισσότερες αιτήσεις (εικόνα 13). Οι κόμβοι ομίχλης διαβάζουν και να επεξεργάζονται το αίτημα εάν δεν υπάρχει φόρτος στον κόμβο, εάν υπάρχει φόρτος μπορεί να ανατεθεί σε κάποιο άλλο κόμβο ομίχλης, διαφορετικά θα σταλεί στους κόμβους σύννεφου ώστε να επεξεργαστούν και αυτοί τα δεδομένα και να στείλουν την τελική απάντηση πίσω στις συσκευές IoT.

Η επικοινωνία των κόμβων ομίχλης πραγματοποιείται με δύο τρόπους. Ο πρώτος αφορά ένα κεντρικό τρόπο αλληλεπίδρασης όπου ένα κεντρικό σύστημα αλληλεπιδρά με τους κόμβους ομίχλης, πιο συγκεκριμένα σε κάθε σύνολο κόμβων ομίχλης υπάρχει και ένας κεντρικός κόμβος που είναι υπεύθυνος για την αλληλεπίδραση, αφού γνωρίζει τη τοπολογία των κόμβων ομίχλης αλλά και την κατάσταση τους. Ο

δεύτερος τρόπος αποτελεί ένα κατανεμημένο σύστημα αλληλεπίδρασης όπου δεν υπάρχει κάποιος κεντρικός κόμβος αλλά η αλληλεπίδραση σε ένα τομέα κόμβων γίνεται μέσω ενός καθολικού πρωτοκόλλου. Τόσο ο κεντρικός όσο και ο κατανεμημένος τρόπος αλληλεπίδρασης βασίζονται στην αλληλεπίδραση ενός κόμβου ομίχλης με τον κόμβο που διαθέτει μικρότερο χρόνο αναμονής. [46]

Συμπερασματικά, η συνεργασία αυτή μπορεί να επιφέρει πολλά πλεονεκτήματα που παρουσιάζονται παρακάτω:

- Μείωση καθυστέρησης λόγω ότι το υπηρεσίες νέφους βρίσκονται τοπικά ή κοντά στο κέντρο δεδομένων.
- Αξιοπιστία. Οι υπηρεσίες IoT βασίζονται στην ύπαρξη αισθητήρων που διασφαλίζουν τη δημόσια ασφάλεια κάτι που δεν επιτυγχάνεται πάντα σε υπηρεσίες υπολογιστικού νέφους.
- Ιδιωτικότητα: Οι υπηρεσίες IoT δημιουργούν θέματα σχετικά με το εάν οι ευαίσθητες προσωπικές πληροφορίες που διαθέτουν είναι ασφαλές. Το πρόβλημα αυτό εξαλείφετε στα τοπικά συστήματα του fog computing αρκεί να μην υπάρχουν κενά ασφάλειας.[44]



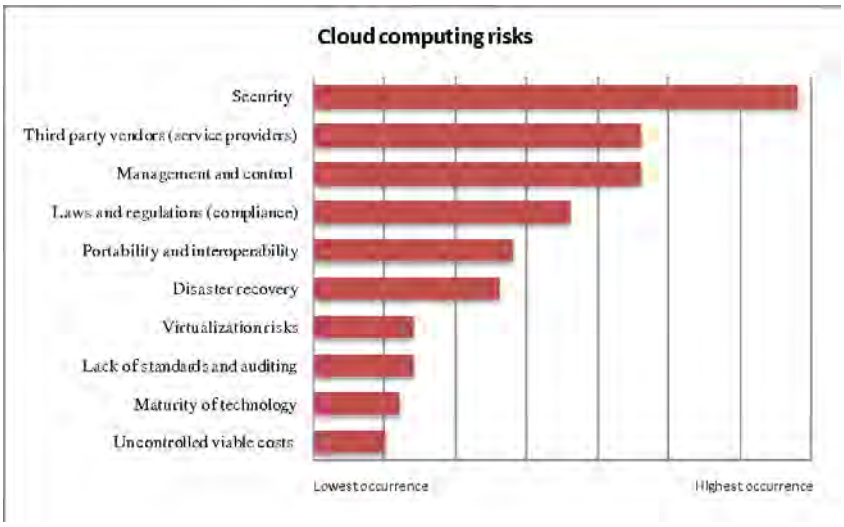
Εικόνα 13: Στρώματα Cloud Computing, Fog Computing, συσκευών IoT.

Κεφάλαιο 4: Ασφάλεια στο Υπολογιστικό Νέφος (Security and Privacy in Cloud Computing)

4.1 Εισαγωγή

Τα οφέλη του υπολογιστικού νέφους είναι ποικίλα και έχουν αναφερθεί λεπτομερώς, ωστόσο κατά τη χρήση του μπορούν να υπάρξουν πολλοί κίνδυνοι όπως η απώλεια δεδομένων και η μειωμένη ασφάλεια τους αλλά και η καταπάτηση των προσωπικών δεδομένων. Διότι το μοντέλο βασίζεται στην επικοινωνία πελατών και μηχανημάτων εξ αποστάσεως δηλαδή σε απομακρυσμένους εξυπηρετητές, ο υπολογιστής του χρήστη χρησιμοποιείται μόνο για την αποστολή και λήψη δεδομένων στο σύννεφο.

Οι κίνδυνοι αυτοί πρέπει να ληφθούν σοβαρά κατά το σχεδιασμό ενός συστήματος υπολογιστικού νέφους. Προκύπτουν τρείς σημαντικές έννοιες η ασφάλεια-security η εμπιστευτικότητα-trust και η ιδιωτικότητα-privacy. [1]



Εικόνα 14: Στατιστικά μετρήσεων σχετικά με την ασφάλεια στο cloud computing

4.2 Ασφάλεια στο Υπολογιστικό Νέφος

Με τον όρο ασφάλεια υπολογιστικού νέφους αναφερόμαστε στις πολιτικές που αφορούν τους ελέγχους και τις τεχνολογίες που χρησιμοποιούνται για την προστασία των δεδομένων και της υποδομής του σύννεφου. Πιο συγκριμένα η ασφάλεια υπολογιστικού νέφους ασχολείται και διασφαλίζει την λύσει προβλημάτων που σχετίζονται με την προστασία προσωπικών δεδομένων, με τη κρυπτογράφηση δεδομένων και δεδομένων που η ασφάλεια τους κινδυνεύει. Πρέπει να σημειωθεί ότι με τον έννοια της ασφάλειας στο υπολογιστικό νέφος δεν εννοείται κάποιο αντιικό πρόγραμμα-antivirus. [2]

4.3Εμπιστευτικότητα-Trust στο Υπολογιστικό Νέφους

Με την έννοια της εμπιστευτικότητα αναφερόμαστε στη βεβαιότητα ότι δεν θα υπάρξει αποκάλυψη του περιεχομένου και όλες οι οντότητες θα συμπεριφερθούν όπως προβλέπουν οι νόμοι και η ηθική δεοντολογία. Μπορούμε να διακρίνουμε τρία είδη εμπιστευτικότητα την εμπιστευτικότητα από άνθρωπο σε άνθρωπο, την εμπιστευτικότητα από μηχάνημα σε μηχάνημα και την εμπιστευτικότητα από μηχάνημα σε άνθρωπο. Τέλος, η τήρηση της εμπιστευτικότητας μπορεί να θεωρηθεί ως επίτευξη του στόχους της ασφάλειας και της ιδιωτικότητας. [2]

4.3.1 Χρήσεις της εμπιστευτικότητας

- Διεπαφές :χρησιμοποιούνται για να συνδέσουν δυο οντότητες μεταξύ τους και να ανταλλάσουν πληροφορίες. Συνήθως χρησιμοποιούνται για τη διασφάλιση ασφάλειας σε βάσεις δεδομένων.
- Κανάλια επικοινωνίας: προκειμένου να διασφαλιστεί η ασφαλής μεταφορά μηνυμάτων μέσω κατάλληλων μηχανισμών.
- Διασφάλιση πνευματικών δικαιωμάτων: που προστατεύονται από νόμους.
- Πρωτόκολλα και υπηρεσίες δικτύων.

- Εντοπισμός κίνησης μηνυμάτων: μέσω της εύρεσης της πηγής και του προορισμού του μηνύματος ακόμη και αν το μήνυμα είναι κωδικοποιημένο ώστε να υπάρξει εμπιστευτικότητα.
- Κρυπτογράφηση και υπηρεσίες κρυπτογράφησης δεδομένων: Η κρυπτογράφηση χρησιμοποιείται ώστε να μην μπορούν να διαβαστούν και αποκτηθούν από μη εξουσιοδοτημένα άτομα.

4.3.2 Τεχνολογία εμπιστευτικότητας

Η ομάδα Trusted Computing Group-TCG παρέχει την τεχνολογία εμπιστοσύνης σε συστήματα υπολογιστικού νέφους. Η εμπιστευτικότητα είναι αναγκαία αφού όλο και περισσότερα προσωπικά δεδομένα διαχειρίζονται από ανθρώπους, αποτελώντας στόχο έλξης υποκλοπής. Πολλές τεχνολογίες στοχεύουν στην ενσωμάτωση ενός μηχανισμού ασφαλείας δεδομένων στα υπολογιστικά συστήματα. Τα συστήματα αυτά αποκαλούνται συστήματα TC και η λειτουργία τους είναι να κρυπτογραφούν τα τμήματα του υπολογιστή που διαθέτουν προσωπικά δεδομένα και προκειμένου να αποκρυπτογραφηθούν δημιουργούνται κλειδιά αποκρυπτογράφησης που αποκαλύπτονται μόνο σε προγράμματα που εμπιστεύεται η τεχνολογία. Στην δοθέν τεχνολογία, προκειμένου να υπάρχουν οι νέες τεχνολογίες εμπιστευτικότητας οι κατασκευαστές χρησιμοποιούν το TCP πρωτόκολλο, λογισμικό αλλά και υλικό σε κάθε υπολογιστή επίσης οι συσκευές έχουν λειτουργικό σύστημα TC. Η τεχνολογία TCμπορεί να προσφέρει ασφάλεια στα συστήματα υπολογιστικού νέφους. Το δοθέν μοντέλο εμπιστοσύνης αρχικά σχεδιάστηκε για την παροχή εμπιστοσύνης και ιδιωτικότητας σε πλατφόρμες χρηστών και σε πλατφόρμες εμπιστοσύνης. Λόγω της ανάπτυξης του διαδικτύου και των δικτύων η τεχνολογία αυτή εφαρμόστηκε σε κατανεμημένα συστήματα σε δίκτυο υπολογιστών. Συμπερασματικά αφού το cloud computing αποτελεί ένα τέτοιο σύστημα εντοπίζουμε και την εφαρμογή της τεχνολογίας. [3]

4.4 Ιδιωτικότητα-Privacy στο Υπολογιστικό Νέφος

Η διασφάλιση των προσωπικών δεδομένων και της ιδιωτικής ζωής των πελατών είναι ένα βασικό μέλημα που πρέπει να ληφθεί σοβαρά, με τη σωστή χρήση των προσωπικών πληροφοριών όσο αφορά το τομέα των καταναλωτών. Σχετικά με το τομέα των επιχειρήσεων και των οργανισμών η ιδιωτικότητα συμπεριλαμβάνει την επιβολή των κατάλληλων νόμων σχετικά με τα προσωπικά δεδομένα των ανθρώπων που περιβάλλεται. Συμπερασματικά και στους δυο τομείς η ιδιωτικότητα αφορά τις ανάγκες των ανθρώπων να κρατηθούν μυστικά και ασφαλείς τα προσωπικά τους δεδομένα. [2]

Σημαντικό κατά την ιδιωτικότητα είναι να τηρείται το μοντέλο data life cycle για το πώς πρέπει ένας οργανισμός να χρησιμοποιεί τις πληροφορίες από τη στιγμή που τις χρησιμοποιεί έως τις στιγμή που τις διαθέτει σημαντικό είναι τα προσωπικά δεδομένα των πελατών να αντιμετωπίζονται ως δεδομένα της εταιρείας. Οι φάσεις που διαθέτει σχετικά με τις πληροφορίες είναι:

1. Δημιουργία
2. Χρήση
3. Μεταφορά
4. Μετασχηματισμός
5. Αποθήκευση
6. Αρχειοθέτηση
7. Καταστροφή [6]

Εικόνα 15: Φάσεις στο data life cycle

4.4.1 Χρήσεις της ιδιωτικότητας

- Κρυπτογράφηση: Αποτελεί την επιστήμη όπου δημιουργούνται chippers και συναρτήσεις κατακερματισμού ώστε να επιτευχθεί μια ασφαλής επικοινωνία μεταξύ δυο οντοτήτων και ας διαμεσολαβεί και μια τρίτη οντότητα. Υπάρχουν ποικίλοι αλγόριθμοι κρυπτογράφησης όπως ο αλγόριθμος DES και RSA, που χρησιμοποιούνται για την ασφάλεια της εμπιστευτικότητας αλλά και της ακεραιότητας των δεδομένων. Πιο συγκεκριμένα ένας αλγόριθμος κρυπτογράφησης μετατρέπει ένα απλό κείμενο εισόδου σε ένα κείμενο που μόνο ο παραλήπτης θα μπορεί να διαβάσει με τη χρήση μυστικού κλειδιού. Οι αλγόριθμοι κρυπτογράφησης διαχωρίζονται σε δυο κατηγορίες ανάλογα με το πλήθος των κλειδιών σε αλγόριθμους συμμετρικού κλειδιού εάν υπάρχει ένα κοινό κλειδί μεταξύ παραλήπτη και αποστολέα και σε αλγόριθμους ασύμμετρου κλειδιού εάν υπάρχουν δυο διαφορετικά κλειδιά ένα για τον αποστολέα και ένα για τον παραλήπτη.
- Έλεγχος ταυτότητας χρήστη για πρόσβαση: Ελέγχεται εάν ο χρήστης έχει εξουσιοδοτημένη πρόσβαση για να έχει πρόσβαση σε κάποια υπηρεσία. Για τον έλεγχο αυτό έχει δημιουργηθεί η υπηρεσία CSA- Cloud Security Alliance για να έχει έλεγχο πρόσβαση και διαχείρισης των ταυτοτήτων πρόσβασης.

- Έλεγχος αυθεντικοποίησης και εξουσιοδότησης ταυτότητας: Ελέγχεται η ταυτότητα των χρηστών μέσω στοιχειών επαλήθευσης δηλαδή μέσω ενός ονόματος χρήστη και ενός κωδικού πρόσβασης . Μετά την επαλήθευση της ταυτότητας ο πάροχος του cloud computing πρέπει να δώσει εξουσιοδότηση μέσω παροχής δικαιωμάτων που καθορίζουν τις αρμοδιότητες των χρηστών.
- Διαχείριση εισβολών: Αποτελεί μια ανεπιθύμητη δραστηριότητα που αποκτά πόρους άλλων χρηστών. Χρησιμοποιείται το επιστημονικό πεδίο αναγνώρισης προτύπων που έχει ως στόχο την ανάπτυξη αλγορίθμων επικεντρωμένων στον εντοπισμό ασυνήθιστων συμβάντων και την αντιμετώπιση τους.
- Ψηφιακές υπογραφές: Αποτελεί ένα είδος της ασσύμετρης κρυπτογραφίας. Χρησιμοποιείται από τον παραλήπτη του μηνύματος ώστε να επαληθεύσει την ταυτότητα του αποστολέα προκειμένου ώστε μέσω της ψηφιακής υπογραφής να εξασφαλίσει την αυθεντικοποίηση και την ακεραιότητα ενός ψηφιακού μηνύματος. Η ψηφιακή υπογραφή δημιουργείται μέσω του αλγορίθμου Digital Signature Standared. [4]

4.4.2 Τεχνολογία της ιδιωτικότητας

Τα δεδομένα στο διαδίκτυο μπορούν πολλές φορές να κλαπούν από εγκληματίες. Στα συστήματα υπολογιστικού νέφους τα προσωπικά δεδομένα εκτίθενται σε ένα τρίτο μέρος , καθιστώντας την διασφάλιση της ασφάλειας σημαντική. Η ασφάλεια μπορεί να επιτευχθεί με την μείωση κενών ασφαλείας μέσω της κρυπτογράφησης και της αποκρυπτογράφησης των αρχείων κειμένων με βάση τους αλγορίθμους AES,RSA.

Η κρυπτογράφηση αρχείων κειμένου, όπου αρχικά φορτώνεται το αρχείο κειμένου ώστε να κρυπτογραφηθεί. Έπειτα εφαρμόζεται το πρώτο επίπεδο κρυπτογράφηση μέσω του αλγορίθμου RSA. Ακολουθεί το δεύτερο επίπεδο κρυπτογράφησης μέσω του αλγορίθμου AES. Τέλος το κρυπτογραφημένο κείμενο που δημιουργήθηκε, αποθηκεύεται σε μια βάση δεδομένων.

Στην αποκρυπτογράφηση αρχείων υπάρχουν τα ίδια βήματα με την κρυπτογράφηση αρχείων κειμένου αλλά σε αντίστροφη σειρά. Αρχικά φορτώνεται το κρυπτογραφημένο κείμενο από τη βάση δεδομένων που είχε αποθηκευτεί.

Εφαρμόζεται το πρώτο επίπεδο αποκρυπτογράφησης μέσω το αλγόριθμου AES. Ακολουθεί το δεύτερο επίπεδο αποκρυπτογράφησης μέσω του αλγορίθμου RSA. [5]

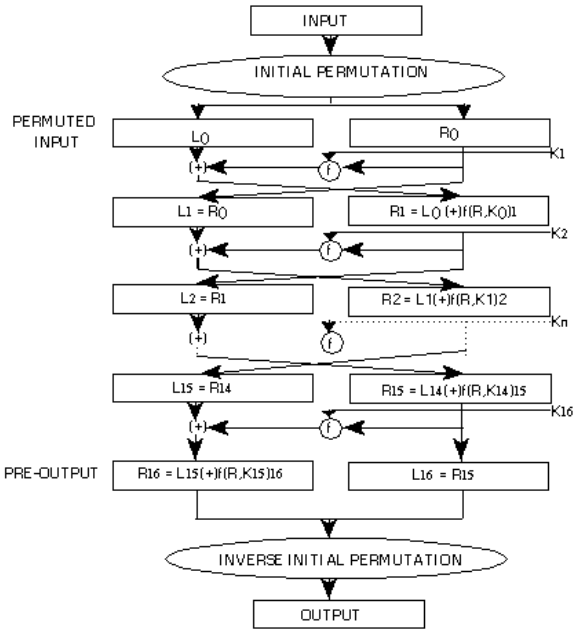
4.5Αλγόριθμοι ασφαλείας στο Cloud Computing

Μπορούμε να χωρίσουμε τους αλγορίθμους ασφαλείας σε δύο κατηγορίες. Στους συμμετρικούς αλγορίθμους κρυπτογράφησης και στους ασύμμετρους αλγορίθμους.

Οι συμμετρικοί αλγόριθμοι κρυπτογράφησης λειτουργούν με την ύπαρξη ενός κοινού μυστικού κλειδιού μεταξύ αποστολές και παραλήπτη. Το κλειδί αυτό χρησιμοποιείται για την κρυπτογράφηση αλλά και για την αποκρυπτογράφηση του μηνύματος. Η κρυπτογράφηση είναι απαραίτητη στην ασφαλή ανταλλαγή δεδομένων. Πρόκειται για μια μέθοδο όπου το αρχικό κείμενο μετατρέπεται σε μια μορφή μη κατανοητή από εισβολείς. [17]

Τέτοιοι αλγόριθμοι είναι:

Data Encryption Standard-DES: Δημιουργήθηκε το 1970 από την εταιρεία IBM για να κρυπτογραφηθούν δεδομένα. Ο αλγόριθμος DES θεωρείται ισχυρότερος από άλλους αλγορίθμους ωστόσο επειδή πλέον απαιτείται μικρός χρόνος για κρυπτογραφία το DES μπορεί να θεωρηθεί ευάλωτο σε επιθέσεις ένα δεν ενισχυθεί. Ο αλγόριθμος παρουσιάζεται στην εικόνα 16. [7],[16]

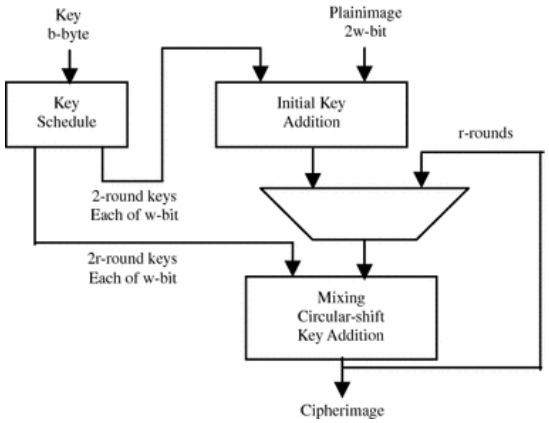


Εικόνα 16: ΑλγόριθμοςDES

Advanced Encryption Standard-AES: Συχνά αναφέρεται ως αλγόριθμος Rijnadel, αποτελεί έναν αλγόριθμο για τη κρυπτογράφηση δεδομένων που δημιουργήθηκε από τον οργανισμό NIST το 2001 και προτάθηκε ως ένα μοντέλου αλγορίθμου που προσφέρει ασφάλεια. Ο αλγόριθμος αυτός χρησιμοποιεί τρία block ciphers 128,192,256 bits, εκτελώντας 4 μετασχηματισμούς SubBytes, ShiftRows, MixColumn και AddRoundKey. Ο αλγόριθμος αυτός παρουσιάζει μειονέκτημα στο ότι είναι δύσκολο να εφαρμοστεί. Στο cloud computing ο αλγόριθμος AES, εφαρμόζεται πριν ανέβουν τα δεδομένα στο σύννεφο όπως φαίνεται στο σχήμα 17. [8],[15],[17].

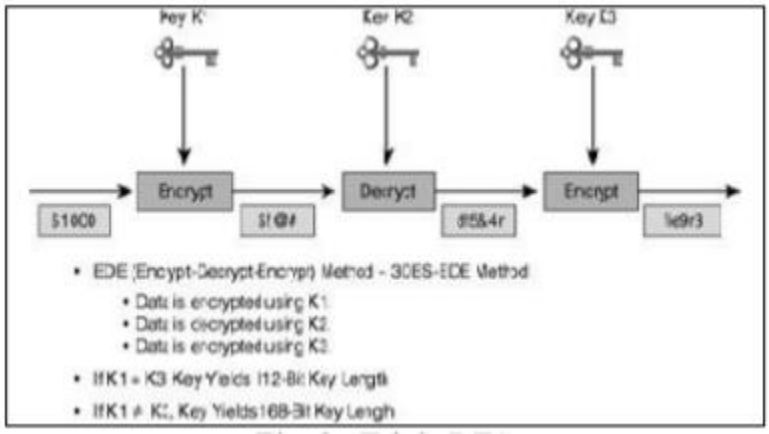
Εικόνα 17: ΑλγόριθμοςAESστοcloud computing.

River Cipher-RC5: Δημιουργήθηκε από τον Ronald Rivest το 1994.Διαθέτει ένα μεταβλητό μέγεθος μπλοκ, κλειδιών και γύρων, με τομείς επέκτασης κλειδιού, κρυπτογράφησης και αποκρυπτογράφησης. Η υλοποίηση του αλγορίθμου που παρουσιάζεται και στην εικόνα 18, είναι εύκολη αλλά παρουσίασε μειωμένη απόδοση με σχέση άλλους αλγορίθμους κρυπτογράφησης. Αποτελείται από πέντε στάδια την αρχικοποίηση δύο μεταβλητών, τη δημιουργία κλειδιού, την εξαγωγή υποκλειδιών από το κύριο κλειδί, την μίξη των υποκλειδιών και την κρυπτογράφηση. Ο αλγόριθμος RC5 στο σύννεφο μπορεί να εφαρμοστεί κατά τη μετάδοση στην κρυπτογράφηση δεδομένων. Κατά τη μεταφορά των δεδομένων θα υπάρχει κρυπτογράφηση και εάν κλαπούν οι εισβολείς δεν θα έχουν το κλειδί για να τα αποκρυπτογραφήσουν. Το κλειδί το ξέρουν μόνο οι χρήστες του συστήματος. Επομένως επιτυγχάνεται και η προστασία της μυστικότητας του χρήστη με την προστασία στα αρχεία. [9],[17]



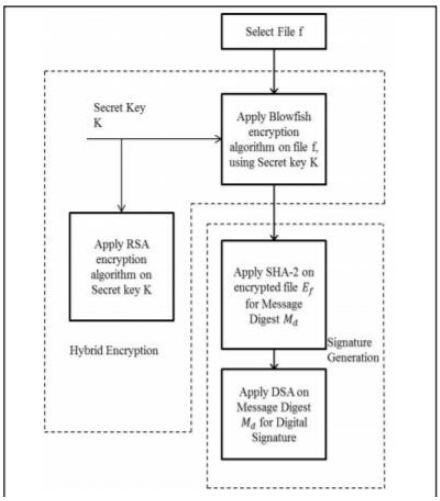
Εικόνα 18: Αλγόριθμος RC5

Triple Data Encryption-3DES: Δημιουργήθηκε το 1998 και χρησιμοποιείται κατά την εγγραφή ενός χρήστη σε κάποιο σύστημα. Η μεθοδολογία του είναι η εξής χρησιμοποιούνται 8 bytesγια τα 3 μπλοκ, άρα 24 διαφορετικά bytes για το κλειδί. Όσο αφορά το cloudκατά την εγγραφή των χρηστών στην υπηρεσία του σύννεφου εισάγεται ένα κλειδί, κατά την πρόσβαση στο σύννεφο ο χρήστης έχει επιλογές για τα κλειδιά ώστε να μπορέσει να θυμηθεί το κλειδί που είχε εισάγει και να πραγματοποιηθεί η κρυπτογράφηση και η αποκρυπτογράφηση δεδομένων. Πιο αναλυτικά με το 3^{ο} κλειδί επιτυγχάνεται η αποκρυπτογράφηση, με το 2^{ο} η κρυπτογράφηση και με το 1^{ο} η αποκρυπτογράφηση όπως φαίνεται στην εικόνα19. Κατά την εγγραφή των χρηστών τα δεδομένα διατηρούνται στο σύννεφο. Μετά την ολοκλήρωση της εγγραφής, ο χρήστης ενεργοποιεί την σύνδεση του με τα στοιχεία που διαθέτει όπως το όνομα χρήστη. Έπειτα ο χρήστης θα έχει πρόσβαση στα δεδομένα του υπολογιστικού νέφους. Αξίζει να τονιστεί ότι ο δοθέν αλγόριθμος είναι αργός και αδύναμος σε σχέση με τους υπόλοιπους αλγορίθμους. [10],[17],[30]



Εικόνα 19: Αλγόριθμος 3DES

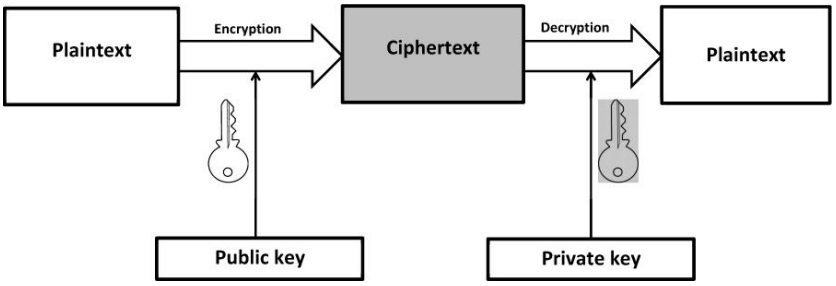
Blowfish: Ο Bruce Schneier δημιούργησε τον αλγόριθμο το 1993. Κατατάσσετε ως ένας αλγόριθμος DES, ικανός να ξεπεράσει πολλά μειονεκτήματα που εμφάνισαν άλλοι αντίστοιχοι αλγόριθμοι κρυπτογράφησης. Αποτελείται από δύο τομείς τον τομέα επέκτασης κλειδιού και τομέα της κρυπτογράφησης. Το κύριο πλεονέκτημα του αλγορίθμου είναι ότι διατέθηκε δημόσια με δωρεάν πρόσβαση. Η χρήση του αλγορίθμου στο cloud computingεντοπίστηκε στην υβριδική κρυπτογραφία ώστε να λυθούν θέματα ασφάλειας και ιδιωτικότητας στα δεδομένα. Η υβριδική κρυπτογραφία αφορά τη χρήση της αλγορίθμου Blowfish αλλά και του RSA και της ψηφιακής υπογραφής κατά τη μετάδοση δεδομένων. Στην κρυπτογράφηση ο αλγόριθμος Blowfishυπάρχει ένα μυστικό κλειδί που στέλνετε με κρυπτογραφημένο τρόπο στον παραλήπτη. Ο αλγόριθμος RSA, χρησιμοποιείται για να κρυπτογραφήσει το μυστικό κλειδί. Τέλος η ψηφιακή υπογραφή υπάρχει για να πιστοποιήσει τα δεδομένα. Η δομή του υβριδικού αλγορίθμου Blowfish παρουσιάζεται στην εικόνα 20. [11],[17],[29]



Εικόνα 20: Αλγόριθμος Blowfish

Οι ασύμμετροι αλγόριθμοι χρησιμοποιούν διαφορετικό κλειδί για κρυπτογράφηση και για αποκρυπτογράφηση. Αυτό το είδος αλγορίθμων είναι πιο σημαντικό διότι επιτυγχάνουν την ασφάλεια δεδομένων και την μεταφορά κλειδιών κρυπτογραφίας. Οι αλγόριθμοι παρουσιάζονται παρακάτω. [16]

River Shamir Adleman-RSA: Αποτελεί τον πιο απλό και ευρέως γνωστό ασύμμετρο αλγόριθμο. Υλοποιείται τόσο για κρυπτογράφηση όσο και για αποκρυπτογράφηση δεδομένων ώστε να μην μπορούν να τα αποκτήσουν οι εισβολείς και εφαρμόζεται συνήθως σε ψηφιακές υπογραφές. Έπειτα από τη κρυπτογράφηση τα δεδομένα αποθηκεύονται στο σύννεφο. Όταν κάποιος χρήστης επιθυμεί να αποκτήσει τα δεδομένα ενημερώνει τον πάροχο σύννεφου, ελέγχεται η ταυτότητα του και αποκτά τα δεδομένα. Είναι ο μοναδικός αλγόριθμος που συμβάλλει στην παράγωγη κλειδιών, του ιδιωτικού κλειδιού που παρέχετε μόνο στους χρήστες που κατέχουν αρχικά τα δεδομένα και του δημόσιου κλειδιού που παρέχετε σε όλους τους χρήστες (Εικόνα 21). Ο δοθέν αλγόριθμος περιλαμβάνει τρία στάδια την δημιουργία ιδιωτικών και δημόσιων κλειδιών, την κρυπτογράφηση και την αποκρυπτογράφηση. [17]



Εικόνα 21: Αλγόριθμος RSA

Ο κώδικας του αλγορίθμου για τα τρία στάδια:

Δημιουργία κλειδιών: Δημιουργούνται από τον πάροχο σύννεφου και του χρήστη.

1. Διαλέγονται δύο πρώτοι αριθμοί x,y.
2. n=x*y
3. Υπολογίζεται η συνάρτηση του Euler Ø(n) = (x-1) * (y-1).
4. Διαλέγετε ένας ακέραιος PU, 1 <PU< Ø(n) και μέγιστος κοινός διαιρέτης του PU, Ø(n) να είναι 1. Το PU είναι το δημόσιο κλειδί.
5. Υπολογίζεται το PR ως ιδιωτικό κλειδί ώστε να είναι αντίστροφο της PUmodØ(n), με βάση την PR = PU-1(mod Ø(n)).
6. Πρέπει να ισχύει για τα κλειδιά PR*PU= 1 mod Ø(n).
7. Το δημόσιο κλειδί συσχετίζεται με το PUκαι τον συντελεστή n (PU,n)
8. Το ιδιωτικό κλειδί συσχετίζεται με το PRκαι τον συντελεστή n (PR,n)

Κρυπτογράφηση:

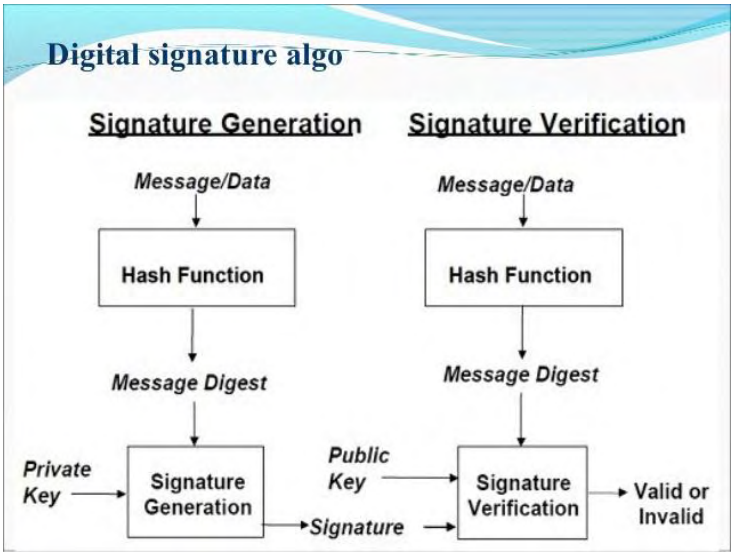
1. Ο πάροχος σύννεφου στέλνει το δημόσιο κλειδί (PU,n) στο χρήστη που επιθυμεί να κρυπτογραφήσει τα δεδομένα.
2. Τα δεδομένα χαρτογραφούνται σε έναν ακέραιο αριθμό με βάση κάποιο πρωτόκολλο.
3. Τα δεδομένα κρυπτογραφούνται με βάση την εξίσωση CT=PTPU(modn).
4. Το κρυπτογραφημένο κείμενο αποθηκεύεται στο σύννεφο.

Αποκρυπτογράφηση:

1. Ο πάροχος σύννεφου ζητάει από το CSP για τα δεδομένα
2. Το CSP επαληθεύει την ταυτότητα του χρήστη και το παρέχει τα κρυπτογραφημένα δεδομένα-CT.

3. Ο χρήστης αποκρυπτογραφεί τα δεδομένα με βάση τη συνάρτηση PT= CTPR (modn)
4. Μόλις ο χρήστης λάβει το PT, μπορεί να αποκτήσει τα αρχικά δεδομένα αντιστρέφοντας τη συνάρτηση κωδικοποίησης. [27]

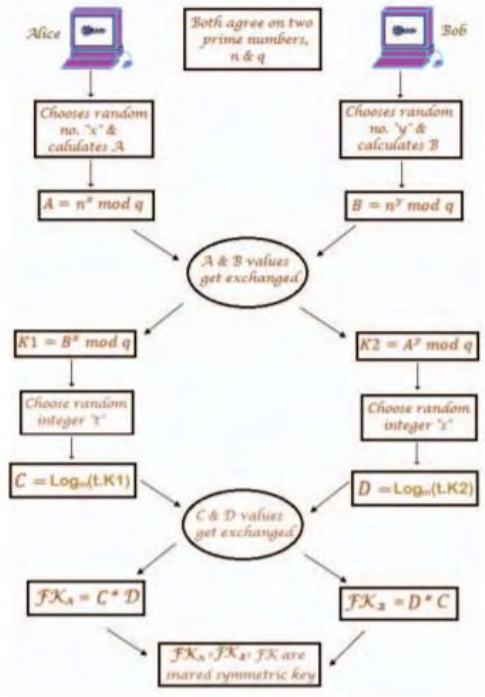
Digital Signature Algorithm-DSA: Υλοποιήθηκε από το ινστιτούτο NIST το 1991. Έχει στοιχεία του αλγορίθμου ElGamal. Είναι σημαντικός αλγόριθμος για την επεξεργασία ηλεκτρονικών δεδομένων δημιουργία κλειδιών επιτυγχάνεται αμέσως, για αυτό το λόγο η διαδικασία της κρυπτογράφησης και της αποκρυπτογράφησης είναι πολύ γρήγορη. Αποτελείται από δύο μέρη, ένα για τη δημιουργία ψηφιακής υπογραφής περιλαμβάνοντας και το κλειδί της και από το μέρος της επαλήθευσης υπογραφής. Κάθε υπογραφή έχει δημόσιο και ιδιωτικό κλειδί που χρησιμεύουν και στα δύο μέρη του αλγορίθμου. Η λειτουργία του μπορεί να κατανοηθεί καλύτερα από την εικόνα 22 [12],[17]



Εικόνα 22: Αλγόριθμος DSA

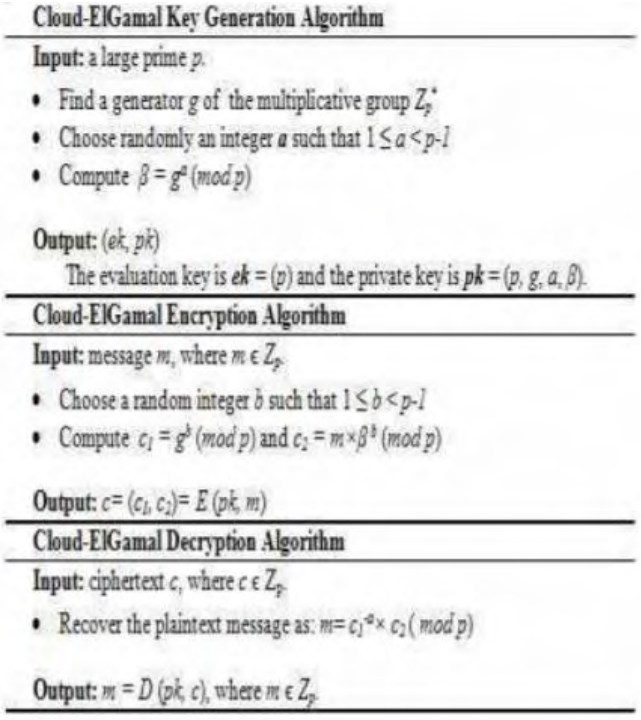
Diffie Helman: Αποτελεί τον πρώτο ασύμμετρο αλγόριθμο, δημιουργήθηκε το 1976 από τον Diffie Helman. Ουσιαστικά πρόκειται για ένα πρωτόκολλο που δημιουργεί ένα μυστικό κλειδί σε ένα μη ασφαλές κανάλι, μεταξύ δύο μερών που δεν έχουν ξανά επικοινωνήσει μεταξύ τους. Έπειτα το κλειδί χρησιμοποιείται για άλλες κρυπτογραφήσεις. Το δοθέν πρωτόκολλο διασφαλίζει την εμπιστευτικότητα και τη

μυστικότητα. Ο τρόπος που επιτυγχάνεται η ανταλλαγή κλειδιών εντός του cloud computing παρουσιάζεται στην εικόνα 23. [13],[17]



Εικόνα 23: Ανταλλαγή κλειδιών μέσω του αλγορίθμου Diffie Hellman.

ElGamal: Βασίζεται στην κρυπτογράφηση μέσω δημόσιου κλειδιού για δεδομένα αλλά και για ηλεκτρονικές υπογραφές πριν ακόμη εμφανιστεί ο αλγόριθμος DSA. Κατά την υλοποίηση του βασίστηκε στον αλγόριθμο Diffie Helman. Το κύριο χαρακτηριστικό του είναι ότι το κείμενο που προκύπτει από τη κρυπτογράφηση είναι δυο φορές μεγαλύτερο από το αρχικό κείμενο, επομένως πρόκειται για ένα χρονοβόρο υπολογισμό που μπορεί να δημιουργήσει καθυστερήσεις. [14],[17]. Στα συστήματα cloud computingo αλγόριθμος χρησιμοποιεί ένα ζευγάρια κλειδιών όπου το ένα είναι το ιδιωτικό κλειδί που παρέχεται μόνο στον κάτοχο των δεδομένων και ένα ακόμη κλειδί που χρησιμοποιείται από ένα τρίτο μέρος για να εκτελέσει εργασίες σε κρυπτογραφημένα δεδομένα. Ο κώδικας του αλγορίθμου παρουσιάζεται στην εικόνα 24. [28]



Εικόνα 24: Αλγόριθμος ElGamal στο cloud computing

4.6 Επιθέσεις στο Cloud Computing

Λόγω των πολλαπλών επιθέσεων που εντοπίζονται στο cloud computingκατηγοριοποιούνται σύμφωνα με τα επίπεδα ασφάλειας, VM, ασφάλειας εφαρμογών και ασφάλειας δικτύων. [16]

- Επίθεση SQL Injection: Η επίθεση γίνεται μέσω του αρχικού κώδικα SQL, εισάγοντας κώδικα που είναι κακόβολους. Το αποτέλεσμα είναι η μη εξουσιοδοτημένη πρόσβαση σε βάσεις δεδομένων που περιέχουν προσωπικά δεδομένα χρηστών μέσω ενός SQL server.Συνήθως αυτό το είδος επιθέσεων προκύπτει από το διαδίκτυο.

  Επίσης όταν ο πηγαίος κώδικας σε μια εφαρμογή είναι προσβάσιμος οι εισβολείς μπορούν να τον αξιοποιήσουν τα δεδομένα ώστε να μάθουν το σχήμα της βάσης δεδομένων για την επίθεση. Μέσω δυναμικών διακομιστών ελέγχεται η είσοδος των χρηστών και αποτρέπεται η πρόσβαση σε βάσεις δεδομένων και στον πηγαίο κώδικα. (Εικόνα 25)

- Επίθεση Cross Attack Scripting-XSS: Ο εισβολέας ανακατευθύνει τις ιστοσελίδες χρήστη εισάγοντας σε αυτές κακόβουλο κώδικα, στον δικό του ιστιότοπο αποκτώντας έτσι τα δεδομένα που επιθυμεί. Υπάρχουν δύο είδη επιθέσεων XSS. Το πρώτο είδος αποκαλείται Stored XSS και αφορά την μόνιμη αποθήκευση του κακόβουλου λογισμικού στον ιστό του χρήστη. Το δεύτερο είδος ονομάζεται Reflected XSS και αντίθετα με το είδος Stored στέλνει τον κακόβουλο κώδικα στον χρήστη και μετά τον αφαιρεί. Η λύση στην επίθεση αυτή είναι το φιλτράρισμα όπου καταργεί το περιεχόμενο που δεν είναι αξιόπιστο μετά τον έλεγχο της εισόδου χρήστη στην εφαρμογή ιστιότοπου. Ωστόσο η μέθοδος αυτή μπορεί να μην είναι έγκυρη σε εφαρμογές που επιτρέπουν HTML αρχεία.

- Επίθεση Domain Name Serves-DNS: Σε πολλές περιπτώσεις ο χρήστης αποκτά πρόσβαση σε ένα διακομιστή καλώντας το όνομα του τομέα-domain namekαι αντί του τομέα που ζητάει δρομολογείται σε κάποιον άλλον τομέα που ουσιαστικά είναι ένας κακόβουλος κώδικας. Αυτό συμβαίνει σε περιπτώσεις επιθέσεωνDNS όπου ο χάκερ χρησιμοποιεί το DNS για να μεταφράσει το όνομα τομέα σε διεύθυνση IP με σκοπό την πρόσβαση στα προσωπικά δεδομένα των χρηστών. Οι επιθέσεις αυτές μπορούν να εξαλειφθούν μέσω μέτρων ασφαλείας. Ένα τέτοιο μέτρο είναι Domain Name System Security Extensins- DNSSEC το οποίο παρέχει αυθεντικοποίηση, ακεραιότητα δεδομένων και επικυρωμένη διάψευση για το αν υπάρχει επίθεση μέσω ενός διακομιστή. Το μέτρο DNSSEC δημιουργεί ψηφιακά αποτελέσματα, που βοηθάνε στην επαλήθευση της προέλευσης των δεδομένων δηλαδή είναι τα δεδομένα είναι παρόμοια με τα δεδομένα στον έγκυρο διακομιστή DNS.

- Επίθεση Phishing: Οι υπηρεσίες υπολογιστικού νέφους χρησιμοποιούνται για να γίνει επίθεση phishing. Ο επιτιθέμενος ανακατευθύνει τον χρήστη σε έναν ψεύτικο ιστότοπο, ζητώντας τον να εισάγει τα στοιχεία χρήστη με αυτόν τρόπο αποκτά πρόσβαση στα δεδομένα του χρήστη. Παράδειγμα τέτοιων επιθέσεων είναι ανεπιθύμητα μηνύματα στο ηλεκτρονικό ταχυδρομείο ή αναδυόμενα παράθυρα με διαφημίσεις.

- Επίθεση Man in the Middle Attacks-MITM: Αφορά την επίθεση εντός δικτύου, όπου ο εισβολέας προσπαθεί να ενταχθεί σε μια συνεχόμενη υπάρχον συζήτηση, δημιουργώντας ανεξάρτητες συνδέσεις με τις οντότητες που επικοινωνούν, μεταδίδοντας ψευδείς πληροφορίες , δημιουργώντας έτσι την ψευδαίσθηση ότι επικοινωνούν κατευθείαν μεταξύ τους, χωρίς να γνωρίζουν ν την ύπαρξη του εισβολέα. Ο εισβολέας όμως έχει καταγράψει όλη την επικοινωνία και έχει αποκτήσει πρόσβαση σε ευαίσθητες πληροφορίες. Προκειμένου να υπάρξει ασφάλεια σε εφαρμογές διαδικτύου το επίπεδο ασφαλείας Secure Socket Layer-SST χρησιμοποιεί το πρωτόκολλο TCP ώστε να υπάρξει ακεραιότητα και εμπιστευτικότητα στα μηνύματα αλλά και για να ελέγχεται η ταυτότητα του χρήστη. (Εικόνα 23)

- Επίθεση Denial of Service-DOS: Σε αυτήν την επίθεση ο εισβολέας έχει ως στόχο ιστοσελίδες ή υπηρεσίες που φιλοξενούνται στο διαδίκτυο, προσπαθεί να κάνεις τις υπηρεσίες μη διαθέσιμες ώστε να επιτραπεί στους χρήστες να προωθούν στους διακομιστές επιθέσεις SYN flooding, UDP flooding και ICMP flooding. Στόχος της επίθεσης είναι να απενεργοποιηθούν υπηρεσίες που παρέχονται από το διακομιστή και εισβάλλοντας στο δίκτυο, ο εισβολέας στέλνει συνεχώς πακέτα δεδομένων στον διακομιστή, χωρίς να αλλάξουν τα πακέτα δεδομένα, η αποκρυπτογράφηση και οι κόμβοι. Τα πακέτα αυτά καταλαμβάνουν όλο το εύρος ζώνης του δικτύου και καταναλώνουν τους πόρους του διακομιστή. Αυτό το είδος επίθεσης είναι πολύ επικίνδυνο, διότι μπορεί να εφαρμοστεί πολύ εύκολα διότι δεν απαιτεί ακριβό εξοπλισμό.

- Επίθεση Distributed Denial of Services-DDOS: Αποτελεί μια προηγμένη επίθεση, ο στόχος της είναι οι διακομιστές πλημμυρίζοντας τους με ένα μεγάλο αριθμό πολλαπλών δικτύων που τους θέτουν σε κίνδυνο με την απενεργοποίηση των υπηρεσιών τους. Επίσης δημιουργεί περισσότερη επισκεψιμότητα σε σχέση με την επίθεση DOS ώστε ο διακομιστής να μην μπορεί να χειριστεί αιτήματα που το κάνουν να διαφέρει από την επίθεση DOS. Για την εξάλειψη τέτοιων εισβολών χρησιμοποιείται ένα σύστημα ανίχνευσης εισβολών IDS αποτελεί ένα λογισμικό που μελετάει και αναλύει

την κυκλοφορία στο δίκτυο και έπειτα ενεργοποιεί κατάλληλες σχετικές ειδοποιήσεις σχετικά με τις εισβολές. Το μοντέλο αυτό έχει ως στόχο την δημιουργία μετρήσεων προκαλώντας καθυστερήσεις στο δίκτυο. Η συνεργασία και των δυο τεχνικών ως ένα αυτοματοποιημένο εργαλείο μπορεί να φέρει πιο ακριβή αποτελέσματα.

- Επίθεση μέσω IP Address: Κάθε κόμβος δικτύου έχει και μια διεύθυνση IP, η οποία έχει εκχωρηθεί σε έναν συγκεκριμένο χρήστη όταν όμως ο χρήστης κλείνει τη σύνδεση δικτύου η διεύθυνση του ανατίθεται σε έναν νέο χρήστη. Υπάρχει κίνδυνος ο νέος χρήστης να μπορέσει να αποκτήσει πρόσβαση στα δεδομένα του προηγούμενη χρήστη διότι η διεύθυνση εξακολουθεί να υπάρχει στην κρυφή μνήμη DNS.

- Επίθεση Zombie: Στόχος της επίθεσης είναι οι εικονικές μηχανές-Virtual Machines όπου στέλνεται μεγάλος αριθμός αιτήσεων από άλλες εικονικές μηχανές εντός του δικτύου. Τα πολλά αιτήματα που στέλνονται σε σχέση με το μικρό χρονικό διάστημα που υπάρχει δημιουργούν επιθέσεις DOS ή DDOS.

- Επίθεση Sniffer: Η επίθεση πραγματοποιείται μέσω προγραμμάτων σε υπολογιστές που επιτρέπουν τη φιλοξενία πακέτων εντός δικτύου Ethernet, με την προσθήκη Network Interface Card-NIC στο κακόβουλο λογισμικό. Εάν τα πακέτα δεδομένων δεν είναι κρυπτογραφημένα μπορούν να κινδυνέψουν από αυτό το είδος επίθεσης. Προκειμένου να αποτραπούν οι επιθέσεις χρησιμοποιείται το πρωτόκολλο APR όπου ελέγχει εάν τα APR πακέτα διαθέτουν λανθασμένες διευθύνσεις εάν υπάρχει επίθεση μέσω των APR πακέτων. Αφού βρίσκεται ο ύποπτος ελέγχει ένα διαθέτει sniffer δηλαδή “λογισμικό με δυνατότητα παρακολούθησης πακέτων ενός δικτύου”. Άλλη μια τεχνική για την αποτροπή επίθεσης είναι το Round Trip Time-RTT όπου χρησιμοποιούνται μετρήσεις RTT για πακέτα ICMP και βάση στατιστικών προκύπτει μια λύση για το εάν υπάρχει η όχι επίθεση. Ωστόσο για πιο σωστά αποτελέσματα ενδείκνυται η χρήση και των δυο τεχνικών.

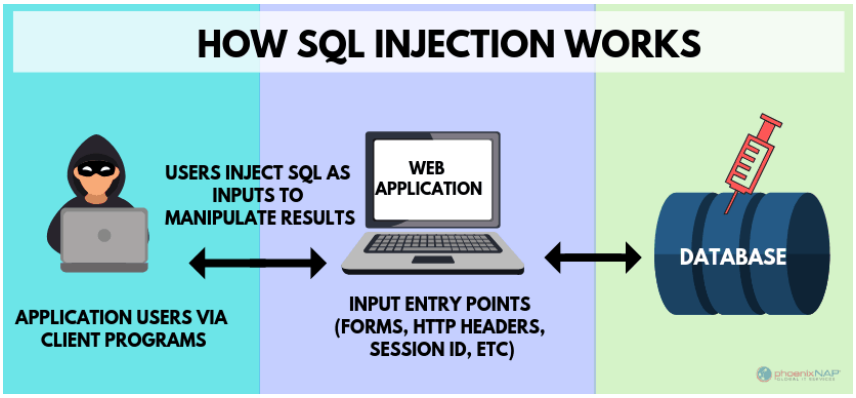
- Επίθεση Wrapping: Όταν ένα χρήστης ζητάει μια εικονική μηχανή VM από έναν περιηγητή, ο περιηγητής στέλνει το αίτημα στον διακομιστή και αυτός του στέλνει ένα μήνυμα SOAP με χρήσιμες πληροφορίες XML. Στην τεχνική αυτή στόχος είναι το μήνυμα SOAP, πιο συγκεκριμένα ο χάκερ αντιγράφει το μήνυμα στην φάση της μετάφρασης και έπειτα το στέλνει στον διακομιστή έχοντας καταφέρει να εισβάλει στις υπηρεσίες υπολογιστικού σύννεφου μέσω κακόβολου κώδικα.

- Επίθεση Cookie Poisoning: Τα cookies " είναι μικρά αρχεία κειμένου τα οποία αποθηκεύονται στον φυλλλομετρητή μας κατά την πλοήγησή μας στο διαδίκτυο. Σκοπός τους είναι να ειδοποιούν τον ιστότοπο που επισκέπτεται ο χρήστης, για την προηγούμενη δραστηριότητά του". Συνήθως στα cookies υπάρχουν αποθηκευμένα στοιχεία σύνδεσης χρηστών, επομένως εάν αποκτηθούν από μη εξουσιοδοτημένα άτομα θα μπορούν να τα χρησιμοποιήσουν παράνομα. Επίσης πολλές φορές αλλάζουν το περιεχόμενο τους ώστε να αποκτήσουν πρόσβαση σε μια μη εξουσιοδοτημένη εφαρμογή.

- Επίθεση CAPTCHA Breaking: Με τον όρο CAPTCHA εννοούμε το τεστ Completely Automated Public Turing που είχε ως σκοπό να αναδείξει στους υπολογιστές εάν οι χρήστες είναι όντως πραγματικοί άνθρωποι ή πρόκειται για κάποιο κακόβουλο λογισμικό. Εάν πρόκειται για κακόβολου λογισμικό παραδείγματος χάριν Trojanioί. Η δοθέν επίθεση έχει ως στόχο να σπάσει τα CAPTCHA μέσω ενός συστήματος ήχου, τα CAPTCHA διαβάζονται εάν μετατραπεί η ομιλία σε κείμενο μέσω κατάλληλου λογισμικού. Η μέθοδος letter overlap χρησιμοποιείται για να αποτρέψουν επιθέσεις, δυσκολεύοντας να παραβιαστεί η συμβολοσειρά του CAPTCHA αλλάζοντας τη γραμματοσειρά και το μήκος της επίσης εντάσσεται στο background του CAPTCHA χρώματα, κουκκίδες και σχήματα.

- Επίθεση Google-Google Dorking: Η επίθεση Google-Dorking έχει ως στόχο τις μηχανές αναζήτησης Google, στοχεύοντας στις ευπάθειες που εμφανίζει στην ασφάλεια, τα λεγόμενα κενά ασφαλείας και να συλλέξουν απαραίτητα χρήσιμα δεδομένα.

- Επίθεση Hypervisor: Η έννοια του hypervisor είναι συνώνυμη με την έννοια Virtual Machine Manager-VMM που έχει ως λειτουργία την εικονοικοποίηση ενός διακομιστή ώστε να διαχωριστεί από τους φυσικούς πόρους και τη λειτουργία λειτουργικών συστημάτων σε μια ενιαία πλατφόρμα υλικού. Ο hypervisor επιτρέπει όταν λειτουργεί κάποιο λειτουργικό σύστημα σε εικονική μηχανή να φέρεται σαν να υπόκειται μόνο έλεγχο υλικού και να μην γνωρίζει εάν υπάρχουν άλλα συστήματα που τη χρησιμοποιούν. Η λειτουργία αυτή είναι πολλές φορές εκτεθειμένη σε επιθέσεις που αφορούν το λειτουργικό σύστημα. Η επίθεση μπορεί να γίνει εάν κάποιο λειτουργικό σύστημα εκτελέσει κάποιο κακόβουλο πρόγραμμα στο κεντρικό υπολογιστή, προσπαθώντας να δημιουργήσει βλάβη, αποκτώντας τον πλήρη έλεγχο και αποκλείοντας την πρόσβαση άλλων λειτουργικών συστημάτων. Επομένως με τη τεχνική αυτή οι εισβολείς θα έχουν πρόσβαση στα δεδομένα.



Εικόνα 25: Επίθεση μέσω της γλώσσας προγραμματισμού SQL.

Κεφάλαιο 5: Θέματα ασφαλείας σε τεχνολογίες Υπολογιστικής Ομίχλης

5.1 Εισαγωγή

Όπως οι υπηρεσίες σύννεφου έτσι και οι τεχνολογίες που βασίζονται στην υπολογιστική ομίχλη ενδέχεται να αντιμετωπίσουν θέματα που αφορούν την ασφάλεια του συστήματος, διότι η κύρια υλοποίηση τους βασίζεται σε περιβάλλοντα που δεν προστατεύονται επαρκώς. Παραδοσιακής επιθέσεις ασφάλειας πραγματοποιούνται για να θέσουν σε κίνδυνο τα συστήματα ομίχλης και τα δεδομένα τους. Σε αυτό το κεφάλαιο θα αναλυθεί η ασφάλεια σε συστήματα fog computing. [23]

5.2 Ασφάλεια σε τεχνολογίες Υπολογιστικής Ομίχλης

Πλέον όλο και περισσότερες αιτήσεις φτάνουν σε κόμβους ομίχλης, αυξάνοντας τον φόρτο τους για επεξεργασία και αποθήκευσης. Ο φόρτος εργασίας πολλές φορές δημιουργεί προβλήματα ασφάλειας. Επίσης οι συσκευές IoT που στέλνουν τις αιτήσεις πρέπει να είναι ασφαλής όπως και η μεταξύ τους επικοινωνία εντός του δικτύου ομίχλης. Η επικοινωνία μεταξύ συσκευών και κόμβων αφορά την επεξεργασία και την αποθήκευση δεδομένων. Οι κόμβοι προκειμένου να ανταπεξέλθουν στον φόρτο αιτήσεων λειτουργούν κατανεμημένα. Αξίζει να τονιστεί ότι πολλές φορές οι συσκευές IoT δεν έχουν γνώση ότι χρησιμοποιούν το δίκτυο ομίχλης και δεν έχουν εφαρμοστεί εκ των προτέρων τεχνικές κρυπτογράφησης. Για να εξασφαλιστεί η ασφάλεια ενδείκνυται η ασύμμετρη κρυπτογραφία και ο μηχανισμός PKI.[23]

5.3 Εμπιστευτικότητα-Trust σε τεχνολογίες Υπολογιστικής Ομίχλης

Όσο αφορά την εμπιστοσύνη, το μοντέλο που ακολουθείται αφορά τους κόμβους στο δίκτυο ομίχλης που έχουν ως κύρια λειτουργία την υλοποίηση υπηρεσιών και θα

πρέπει να είναι σε θέση να αναγνωρίσουν εάν οι συσκευές που ζητάνε τις υπηρεσίες αυτές είναι αξιόπιστες. Επίσης και οι συσκευές IoT που έχουν κύρια λειτουργία την αποστολή δεδομένων και υπηρεσιών πρέπει να είναι σε θέση να αναγνωρίσουν εάν οι κόμβοι ομίχλης είναι ασφαλής. Σε αντίθεση με τις υπηρεσίες υπολογιστικού νέφους, απαιτούνται έμπιστες αλληλεπιδράσεις του παρελθόντος. Οι υπηρεσίες fog computing εάν και διαθέτουν δυνατότητες μέτρησης εμπιστευτικότητας, προκύπτουν προβλήματα διότι δεν γνωρίζουμε εάν διαχειρίζονται σωστά. Επιλέγονται σε μοντέλα υπηρεσιών που βασίζονται στη γενική γνώμη opinion-based model, κυρίως σε μοντέλα ηλεκτρονικού εμπορίου.

Στην αρχιτεκτονική του μοντέλου βασίζεται η υπηρεσία Service Level Agreement-SLA αποτελεί "μια δέσμευση μεταξύ του παρόχου υπηρεσιών και του πελάτη". Εάν όμως ο πελάτης χρησιμοποιήσει κατευθείαν την υπηρεσία σύννεφου το SLA είναι περιορισμένο. [23]

5.4 Αυθεντικοποίηση-Authentication σε τεχνολογίες Υπολογιστικής Ομίχλης

Η έννοια της αυθεντικοποίησης αναφέρεται στον έλεγχο ταυτότητας υπολογιστών που ανήκουν σε δίκτυο ομίχλης. Μια συσκευή πριν εισαχθεί στο δίκτυο και αποκτήσει πρόσβαση στα δεδομένα πρέπει πρώτα να ελεγχθεί η ταυτότητα της, ώστε να επιβεβαιωθεί ότι δεν πρόκειται για μη εξουσιοδοτημένη πρόσβαση. Προκειμένου να ελεγχθεί η ταυτότητα χρησιμοποιείται πολλές ο μηχανισμός Public Key Infrastructure-PKI εάν και υστερεί λόγω των περιορισμένων πόρων που διαθέτει για συσκευές IoT. Εκτός από το PKI έχουν δημιουργηθεί πολλαπλά πρωτόκολλο που έχουν ως υλοποίηση τη δημιουργία ενός δημόσιου κλειδιού ώστε να διασφαλίσουν μια ασφαλή επικοινωνία ελέγχοντας τη ταυτότητα των μελών της. Επίσης στην αυθεντικοποίηση εφαρμόζεται η αρχή Certifying Authority-CA, που απαγορεύει στις μη εξουσιοδοτημένους κόμβους να εισέλθουν στο δίκτυο ομίχλης με αποτέλεσμα να μειωθούν αιτήματα για υπηρεσίες από κακόβουλους κόμβους. Άλλο ένα θέμα που τίθεται είναι ότι το μοντέλο fog computing αποτελείται από δυναμικούς κόμβους που συχνά συνδέονται και αποσυνδέονται από το δίκτυο, οι χρήστες πρέπει όμως να

εξυπηρετούνται ανεξάρτητα από τις κινήσεις των κόμβων. Οι αλγόριθμοι που χρησιμοποιούνται συνήθως για τη διασφάλιση της αυθεντικότητας είναι ο Rivest Shamir Adleman-RSA και ο Elliptic Curve Cryptography-ECC. [23],[21]

5.5 Ιδιωτικότητα-Privacy σε τεχνολογίες Υπολογιστικής Ομίχλης

Η διασφάλιση της ιδιωτικότητας, αποτελεί πιο δύσκολο έργο από την ιδιωτικότητα στις υπηρεσίες υπολογιστικού νέφους, λόγω του σχεδιασμού του που βρίσκεται πιο κοντά στις συσκευές χρηστών και επομένως η συλλογή ευαίσθητων προσωπικών δεδομένων είναι πιο εύκολη.

Η ιδιωτικότητα μπορεί να παραβιαστεί εάν οι κόμβοι δεν είναι καλά προστατευμένοι και έτσι επιτρέψουν την είσοδο από εισβολείς. Οι εισβολείς με την πρόσβαση τους στο δίκτυο ομίχλης καταφέρουν να υποκλέψουν προσωπικά δεδομένα που μεταφέρονται μεταξύ των κόμβων ομίχλης. Επίσης η αρχιτεκτονική του fog computingπου βρίσκεται ανάμεσα σε υπηρεσίες cloud computingκαι σε συσκευές IoTείναι από μόνη της επίφοβη σε επιθέσεις κατά της ιδιωτικότητας. Εκτός από τα προσωπικά δεδομένα κινδυνεύει η αποκάλυψη της τοποθεσίας των χρηστών εάν αποκαλυφθεί η τοποθεσία του εξοπλισμού αλλά και οι συνήθεις τους μέσω του υπολογιστικού νέφους. Όσο εξελίσσεται η τεχνολογία τόσο τίθενται θέματα ιδιωτικότητας, παραδείγματος χάριν στα έξυπνα σπίτια εάν ανακτηθούν τα δεδομένα που στέλνονται στους κόμβους ομίχλης τότε ο επιτιθέμενος θα μπορέσει να αποκτήσει τον έλεγχο όλου του σπιτιού. Πιο αναλυτικά η παραβίαση ιδιωτικότητας μπορεί να χωριστεί σε τέσσερις κατηγορίες:

- Identity privacy: Υποκλοπή στοιχειών που αφορούν τη ταυτότητα του χρήστη όπως ονοματεπώνυμο, αριθμός πιστωτικής κάρτας, διεύθυνση κατοικίας, αριθμός κινητού τηλεφώνου.
- Usage privacy: Αφορά τις πληροφορίες που προκύπτουν από τη χρήση υπηρεσιών υπολογιστικού νέφους. Παραδείγματος χάριν οι υπηρεσίες που χρησιμοποιούνται στα έξυπνα οχήματα μπορούν να εξάγουν κάποιες πληροφορίες σχετικά με τον χρήστη.
- Data privacy: Υποκλοπή δεδομένων, που μεταφέρονται εντός του δικτύου ομίχλης μετατρέποντας το σε αναξιόπιστο.

- Location privacy: Με την εξέλιξη της τεχνολογίας οι περισσότερες κινητές συσκευές διαθέτουν σύστημα εντοπισμού, που χρησιμοποιείται από τις περισσότερες εφαρμογές παραδείγματος χάριν από εφαρμογές κοινωνικής δικτύωσης ή ακόμη και από την εφαρμογή φυλλομετρητή. Οι χρήστες προκειμένου να χρησιμοποιήσουν τις δυνατότητες που δίνει ή ενεργοποίηση τοποθεσίας στις εφαρμογές τη χρησιμοποιούν, παραβιάζοντας έτσι η ιδιωτικότητα της τοποθεσίας τους και εκθέτοντας τους σε κίνδυνο . [23]

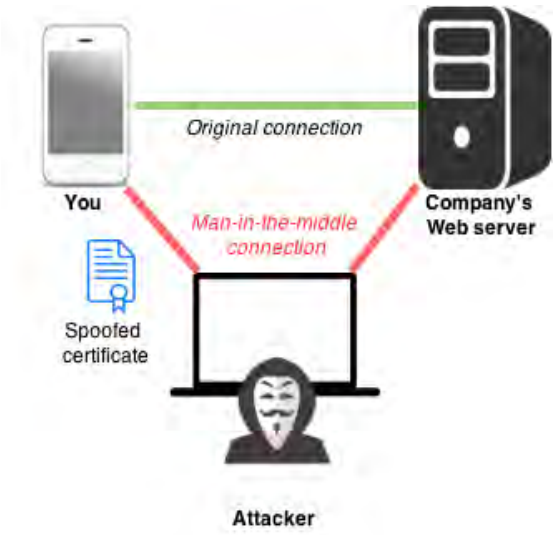
5.6 Τεχνολογία ασφάλειας σε Fog Computing

Η διαφορετικότητα των συστημάτων ομίχλης έκανε απαραίτητα τα περιβάλλοντα multi-OS ώστε μια εφαρμογή να μπορέσει να επικοινωνήσει με τα συστήματα και τους χρήστες. Η τεχνολογία εικονικοποίησης-virtualization δεν επιτρέπει τη πρόσβαση άλλων λειτουργικών συστημάτων, ωστόσο αν προκύψει πρόβλημα σε επίπεδο πυρήνα η πρόσβαση δεν θα αποτραπεί. Σε περιβάλλοντα ομίχλης πολλές πληροφορίες είναι αποθηκευμένες σε κόμβους επομένως εάν υπάρξουν πληροφορίες που διακινδυνεύουν τις αδυναμίες του συστήματος θα υπάρξουν θέματα ασφάλειας. Προκειμένου να αντιμετωπιστούν τα προβλήματα αυτά απαιτείται μια δυναμική τεχνική που μπορεί να αναλύσει τις κινήσεις των κόμβων σε πραγματικό χρόνο, τέτοιες τεχνικές συνήθως έχουν υψηλή απόδοση. Ωστόσο τίθενται θέματα στην εφαρμογή αυτής τεχνική σε περιβάλλοντα multi-OS λόγω της μειωμένης υπολογιστικής ισχύς που διαθέτουν και στην επεξεργασία σε πραγματικό χρόνο. [22]

5.7Τεχνικές δυσκολίες εφαρμογής πολιτικών ασφαλείας σε Fog Computing

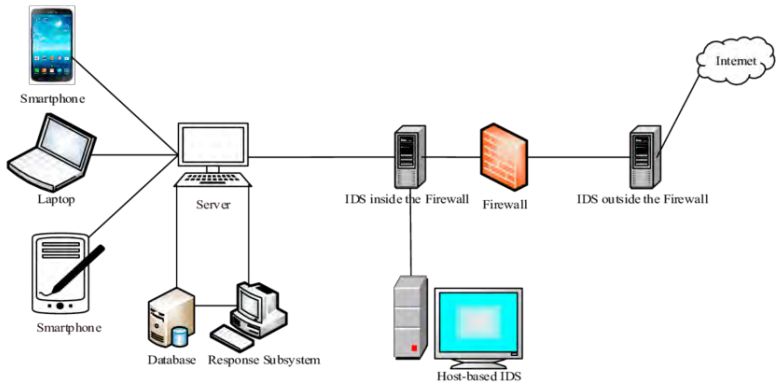
Ορισμένες πολιτικές ασφαλείς που εφαρμόζονται είναι δύσκολο να εφαρμοστούν σε περιβάλλον λόγω της τυπικής επίθεσης σε περιβάλλον υπολογιστικής ομίχλης Man in the Middle που παρουσιάζεται στην εικόνα 26. Σε αυτήν την επίθεση οι πύλες των κόμβων ομίχλης κινδυνεύουν από αντικατάσταση ή από το να διακυβευτούν από εισβολές, με στόχο να τις θέσουν σε κίνδυνο. Επομένως θα τεθούν θέματα

ασφάλειας στην επικοινωνία που πολλές φορές δεν μπορούν να λυθούν μέσω της κρυπτογράφησης όπως στο cloud computing. Ο λόγος που η κρυπτογραφία είναι ανεπαρκής είναι ότι οι συσκευές που χρησιμοποιούνται στο fog computing είναι κινητές με μικρή αντοχή μπαταρίας, επομένως κατά τη διαδικασία κρυπτογράφησης που απαιτείται μεγάλο ποσοστό μπαταρίας πιθανόν να μην υπάρχει σε αυτές τις συσκευές .



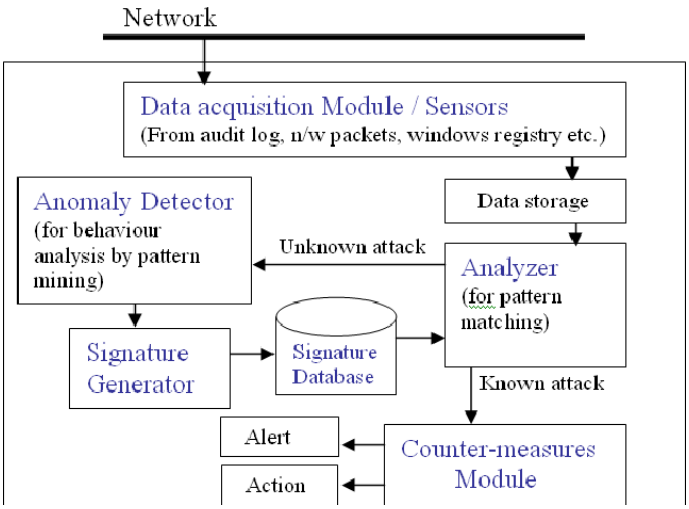
Εικόνα26: Επίθεση Man in the Middle

Επίσης υπάρχει δυσκολία στην εφαρμογή τεχνικών ανίχνευση εισβολής-Intrusion Detection που χρησιμοποιούνται για να ανιχνεύσουν επιθέσεις όπως flooding attack ,εικονικών μηχανών-VM και hypervisor. Πιο συγκεκριμένα οι τεχνικές αυτές παρακολουθούν τις κινήσεις και καταγράφουν των χρηστών εντός του συστήματος, προκειμένου να εξάγει χρήσιμες πληροφορίες προκειμένου να εντοπίσει παράξενες κινήσεις που οδηγούν σε ύπαρξη επίθεσης όπως DoS επίθεση. Η αρχιτεκτονική του μοντέλου ανίχνευσης εισβολής παρουσιάζεται στην εικόνα 27. Ο λόγος που αυτές οι τεχνικές είναι δύσκολο να εφαρμοστούν στο fog computing είναι διότι κατακλύζονται από συσκευές IoT όπως κινητά τηλέφωνα αποτελούμενα από περιορισμένους υπολογιστικούς πόρους. Συνεπώς δεν μπορούν να συμμετέχουν στην διαδικασία εύρεσης απειλής και πιο συγκεκριμένα λογισμικών rootkit που χρησιμοποιούνται από τους εισβολείς ώστε να προκαλέσουν επίθεση και να αποκτήσουν πρόσβαση σε δεδομένα.



Εικόνα 27: Αρχιτεκτονική ανίχνευσης εισβολών

Επίσης και η ανίχνευση εισβολής σε περιβάλλον υπολογιστικής ομίχλης-Malicious Detection in Fog Computing Environments δεν μπορεί να εφαρμοστεί λόγω των συσκευών IoT και της περιορισμένης υπολογιστικής τους ισχύς η ανίχνευση με βάση τη συμπεριφορά των κινήσεων εντός του συστήματος ομίχλης είναι δύσκολη και δαπανηρή και η ανίχνευση τους σε περιβάλλον ομίχλης δεν εντοπίζει πολλά είδη κακόβουλου κώδικα. Σχετικά με την λειτουργία της ανίχνευσης εισβολής όταν οι κόμβοι ομίχλης διακυβεύονται η τεχνική για να εντοπιστεί εάν υπάρχει κακόβουλος κώδικας είναι η Anomaly hybrid detection-ADS η λειτουργία του παρουσιάζεται στην εικόνα 28, όπου μπορεί να εφαρμοστεί και σε συνδυασμό με τις ψηφιακές υπογραφές. Ωστόσο, πολλές τεχνολογίες για να εξαλείψουν αυτό το πρόβλημα κατανέμουν τους κόμβους ομίχλης και αν θεωρηθεί ότι υπάρχει κακόβουλος κώδικας στέλνεται στους κόμβους σύννεφου και σε περίπτωση που όντως υπάρχει στέλνεται το αποτέλεσμα ελέγχου πίσω στους κόμβους ομίχλης.



Εικόνα 28: ΜοντέλοAnomaly hybrid detection-ADS

Οι υπηρεσίες υπολογιστικής ομίχλης βασίζονται σε κόμβους που επεξεργάζονται τις αιτήσεις που προέρχονται από συσκευές IoT. Δημιουργείται πρόβλημα εάν ο φόρτος αιτήσεων στο σύστημα είναι μεγάλος, τότε υπάρχει συνεργασία των κόμβων ομίχλης και η επίθεση είναι πιο εύκολη, το να ανιχνευτούν οι κακόβουλοι κόμβοι είναι δύσκολο. Επίσης σε ένα τέτοιο συμβάν πρέπει να υπάρχει εμπιστοσύνη μεταξύ των κόμβων μέσω ενός πρωτοκόλλου αυθεντικοποίησης και οι κόμβοι αυτοί να βρίσκονται εντός του συστήματος ομίχλης.

Δυσκολία αντιμετωπίζεται και στη προστασία των δεδομένων διότι οι συσκευές IoT δυσκολεύονται να αντιμετωπίσουν ένα μεγάλο όγκο δεδομένων. Η επεξεργασία πραγματοποιείται στους κόμβους ομίχλης, όπου και τίθενται ζητήματα ασφάλειας αφού όταν οι προσωπικές πληροφορίες επεξεργάζονται δεν πρέπει να κινδυνεύουν από επιθέσεις, αφού εκλείπει η τεχνική της κρυπτογράφησης και της αποκρυπτογράφησης λόγω των περιορισμένων πόρων στις συσκευές IoT εκτός και αν πρόκειται για κάποιον ελαφρύ αλγόριθμο κρυπτογραφίας.

Τέλος άλλη μια τεχνική ασφαλείας που δεν μπορεί να εφαρμοστεί εύκολα είναι η διαχείριση δεδομένων-Data Management, διότι οι κόμβοι νέφους είναι κατανεμημένοι σε διάφορες γεωγραφικές εκτάσεις επομένως δεν είναι γνωστή η τοποθεσία που βρίσκονται τα δεδομένα. Επίσης οι χρήστες δεν γνωρίζουν τις υπηρεσίες που μπορεί να προσφέρει ο κάθε κόμβος και αν δημιουργηθούν διπλότυπες αιτήσεις θα υπάρξει κατάχρηση των πόρων. Δημιουργούνται θέματα ασφαλείας στους κόμβους συνήθως από μη εξουσιοδοτημένους χρήστες. λόγω λανθασμένης διαχείρισης. [22]

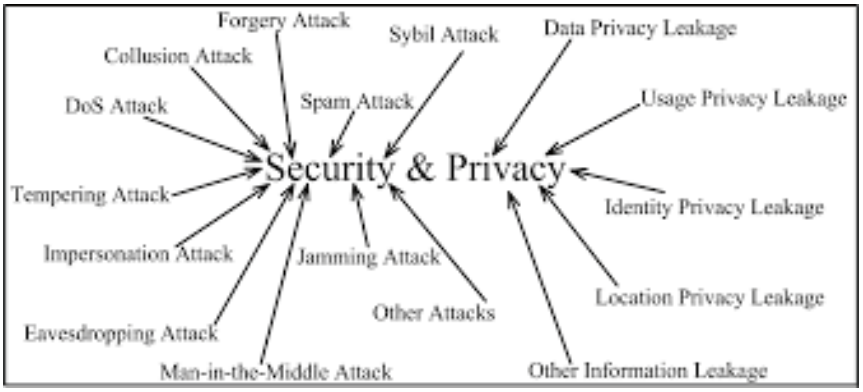
5.8 Επιθέσεις στο Fog Computing

Η τεχνολογία της υπολογιστικής ομίχλης είναι εκτεθειμένη σε ποικίλες κακόβουλες επιθέσεις, ειδικά εάν δεν εφαρμοστούν τα κατάλληλα μέτρα ασφαλείας. Ωστόσο θεωρείται πιο ασφαλής μοντέλο από το cloud computing κυρίως λόγο ότι η αρχιτεκτονική του βασίζεται στην αποθήκευση δεδομένων σε τοπικούς κόμβους,

δυσκολεύοντας έτσι τους εισβολείς να τα υποκλέψουν. Βέβαια αφού το fog computing αποτελεί εξέλιξη του cloud computing, κληρονομούνται και κίνδυνοι από τον προκάτοχο του. Οι επιθέσεις που εντοπίζονται στο fog computing είναι οι ακόλουθες (εικόνα 29):

- Forgery: Οι επιτιθέμενοι δημιουργούν ψεύτικες πληροφορίες προκειμένου να δημιουργήσουν μια μη πραγματική ταυτότητα και στέλνοντας μη εξουσιοδοτημένες αιτήσεις, ώστε να εξαπατήσουν τους χρήστες του δικτύου στέλνοντας μη εξουσιοδοτημένες αιτήσεις. Επίσης μπορούν να καταναλώσουν πόρους δικτύου όπως ενέργεια και εύρος ζώνης μειώνοντας έτσι την απόδοση του συστήματος.
- Tampering: Οι επιτιθέμενοι μεταχειρίζονται τα δεδομένα που μεταφέρονται εντός του δικτύου με το να τα διαγράφουν ή να τα επεξεργάζονται. Αυτή η κίνηση μειώνει την αποδοτικότητα και την αποδοτικότητα του συστήματος ομίχλης. Αξίζει να τονιστεί ότι αυτή η επίθεση είναι επικίνδυνη διότι δεν μπορεί να εντοπιστεί εύκολα λόγω της κινητικότητας της συσκευής.
- Spam: Αφορά ανεπιθύμητα πλαστά δεδομένα, με πληροφορίες λανθασμένες από τους επιτιθέμενους. Εφαρμόζεται συνήθως στο λογισμικό του ηλεκτρονικού ταχυδρομείου. Τα αποτελέσματα αυτής της επίθεσης είναι η κατανάλωση πόρων εντός του δικτύου και η παραβίαση της ιδιωτικότητας
- Sybil: Στόχος της επίθεσης είναι να χρησιμοποιηθεί ψεύτικη ταυτότητα ανακατασκευασμένη ή κλεμμένη από τους χακερ ώστε να αποκτηθεί ο έλεγχος της υπολογιστικής ομίχλης και να μετατραπούν οι κόμβου ομίχλης, μειώνοντας έτσι την αποτελεσματικότητα του δικτύου. Οι επιπτώσεις της επίθεσης εντοπίζονται στην ακεραιότητα των πόρων και στην συνολική επίδοση του δικτύου.
- E-Jamming: Το δίκτυο πλημμυρίζεται με ένα μεγάλο όγκο δεδομένων με στόχο να καταναλωθούν όλοι οι κόμβοι, τα διαθέσιμα κανάλια επικοινωνίας και οι πόροι ώστε η εξυπηρέτηση του χρήστη να μην είναι εύκολη και αξιόπιστη.
- Eavesdropping: Όταν δεν υπάρχει κρυπτογράφηση δεδομένων ή δεν έχει εφαρμοστεί επιτυχημένα μπορεί να αποκτηθεί πρόσβαση στην επικοινωνία διαβάζοντας ή ακόμη και ακούγοντας τη μετάδοση.

- Collusion: Αφορά την ένωση και τη συνεργασία ομάδων ώστε εξαπατήσουν τις νόμιμες και εξουσιοδοτημένες ομάδες και να αποκτήσουν τα πλεονεκτήματα και τα δικαιώματα που έχουν. Πιο συγκεκριμένα η επίθεση αφορά κόμβους IoT, fog και cloud.
- Impersonation: Αφορά τους διακομιστές εντός του δικτύου, οι εισβολείς κάνουν έναν διακομιστή να φαίνεται γνήσιος και αξιόπιστος παρέχοντας υπηρεσίες με κακόβουλες ενέργειες, το ίδιο μπορεί να πραγματοποιηθεί και σε κόμβους ομίχλης.
- Man in the Middle: Η δοθέν επίθεση είναι πολύ γνωστή και δεν μπορεί εύκολα να ανιχνευτεί μέσω των κλασσικών μεθόδων. Καταναλώνει μικρό ποσοστό πόρων στις συσκευές ομίχλης όπως της CPU και της μνήμης. Οι εισβολείς σε αυτή την επίθεση χρησιμοποιούν ένα ψεύτικο σενάριο ώστε να βρεθεί ανάμεσα στους κόμβους ομίχλης και να αναμεταδώσει και να τροποποιήσει δεδομένα. Οι χρήστες δεν γνωρίζουν ότι η επικοινωνία τους έχει παραβιαστεί. Δυο είναι οι τρόποι που μπορεί να εφαρμοστεί η επίθεση ο ένας αφορά την επίθεση σε ευαίσθητες πύλες των κόμβων ομίχλης με το ανανέωση της ROM και με τη προσθήκη ενός ψεύτικου περιβάλλον. Ο άλλος τρόπος αφορά την εισαγωγή κακόβουλου λογισμικού στο σύστημα όπως το πρόγραμμα hookστο πρωτόκολλα TCP/IP. Το hook εισάγει κώδικα εντός μιας κλήσης συστήματος προκειμένου να τροποποιηθεί.
- Denial of Service-DOS: Στέλνονται στους κόμβους ομίχλης, ψεύτικα δεδομένα και ψεύτικες αιτήσεις επεξεργασίας αλλά και αποθήκευσης με αποτέλεσμα να μην εκτελούνται οι αιτήσεις των πραγματικών χρηστών. Καταναλώνετε μεγάλος όγκος πόρων όπως η μπαταρία και ο χρόνος απόδοσης που είναι και περιορισμένοι. [24]



Εικόνα 29: Επιθέσεις σε Fog Computing

Κεφάλαιο 6: Μελέτη αλγορίθμων ασφάλειας σε Fog Computing και IoT περιβάλλοντα

6.1 Αλγόριθμος κρυπτογραφίας σε περιβάλλον Fog Computing και IoT

Το Cloud Computing δεν ενδείκνυται να χρησιμοποιείται σε κρίσιμες εφαρμογές που απαιτούν επεξεργασία σε πραγματικό χρόνο (έξυπνα φανάρια, έξυπνα οχήματα, έξυπνο κτήριο) λόγω της καθυστερημένης μεταφόρτωσης και λήψης δεδομένων. Το Fog Computing, λύνει αυτό το πρόβλημα με την τοπική επεξεργασία. Επίσης όσον αφορά την ασφάλεια υπάρχει κίνδυνος στο να μεταφέρουμε ένα κρυπτογραφημένο αρχείο στο σύννεφο γι' αυτό και χρησιμοποιείται η τοπική επεξεργασία που προσφέρει το Cloud Computing.

Ο δοθέν αλγόριθμος βασίζεται στη κρυπτογράφηση φορητών συσκευών (φορητοί υπολογιστές, κινητά τηλέφωνα, δρομολογητές, κινητοί αισθητήρες). Η κρυπτογράφηση και η αποκρυπτογράφηση γίνεται συνήθως με τους αλγορίθμους AES, DES και πιο συγκεκριμένα με τον αλγόριθμο RSA. Ωστόσο η υλοποίηση σε κινητές συσκευές δεν μπορεί να πραγματοποιηθεί λόγω της περιορισμένης υπολογιστικής ισχύς που διαθέτουν. Στον αλγόριθμο AES έχει προταθεί μια αλλαγή ώστε να μειωθεί η ώρα εκτέλεσης και γύρων μέσω S-Boxes αλλά μόνο για μικρού μεγέθους αρχείου. Η ανάγκη για κρυπτογραφία μεγάλου μεγέθους αρχείου έφερε την ανάγκη για τη δημιουργία του αλγορίθμου.

Η λύση δόθηκε με τη συνεργασία IoT συσκευών που από μόνες τους παρουσιάζουν περιορισμένη υπολογιστική και αποθηκευτική ισχύ. Η συνεργασία αυτή δημιουργεί τέσσερα βασικά θέματα οπού και αντιμετωπίζει και αλγόριθμος:

1. Ετερογένεια: Προκύπτουν θέματα λόγω της διαφορετικότητας στην αρχιτεκτονική υλικού και λειτουργικού συστήματος στις συσκευές.
2. Δυναμική κλιμάκωση: Πρέπει να υπάρχει ευέλικτη διαχείριση εντός του δικτύου IoT, λόγω της αρχιτεκτονικής του δικτύου που επιτρέπει την ένταξη και την αποχώρηση των συσκευών, όποτε το επιθυμούν.
3. Επικοινωνία: Καθίσταται δύσκολη η επικοινωνία των μελών του δικτύου IoT λόγω της δυναμικής διαμόρφωσης τους και της ετερογένειας των συσκευών.

4. Ανάθεση εργασίας: Λόγω των ποικίλων συσκευών IoT στο δίκτυο, η ανάθεση εργασιών για τον κατάλληλο κόμβο πελάτη γίνεται με βάση την επεξεργαστική τους ισχύ και τον φόρτο εργασίας που απαιτείται για του υπολογισμούς. [26]

6.1.1 Αρχιτεκτονική του αλγορίθμου

Όπως φαίνεται και στην εικόνα 30 ο δοθέν αλγόριθμος αποτελείται από τρείς οντότητες. Τους πελάτες, έναν ελεγκτή-controller που ουσιαστικά αποτελεί ένα κεντρικό κόμβο και τους χρήστες. Πιο αναλυτικά παρουσιάζονται παρακάτω οι χρήσεις τους στο δίκτυο.

- Χρήστες: Οποιαδήποτε κινητή συσκευή με περιορισμένη υπολογιστική ισχύ όπως το κινητό τηλέφωνο, μπορεί να θεωρηθεί χρήστης. Οι χρήστες έχουν κάποιες εργασία και επειδή δεν μπορούν να τις διεκπεραιώσουν αυτοί τις στέλνουν σε κάποια άλλη συσκευή μέσω του ελεγκτή.
- Πελάτες: Μπορεί να θεωρηθεί οποιοσδήποτε κόμβος που διαθέτει επεξεργαστική και αποθηκευτική ισχύ. Επίσης οι κόμβοι αυτοί μπορούν να "τρέξουν" προγράμματα lightweight socket (φορητοί υπολογιστές). Ο προγραμματισμός socket απαιτείται για να διασφαλίσει την επικοινωνία των πελατών.
- Ελεγκτής: Αποτελεί έναν ειδικό τύπο πελάτη και πιο συγκεκριμένα έναν κεντρικό κόμβο, υπεύθυνο για τη διαχείριση ολόκληρου του δικτύου συμπεριλαμβανόμενων των κόμβων πελατών και ομίχλης. Επίσης διαθέτει ένα πρόγραμμα στους χρήστες με τους πελάτες που είναι διαθέσιμοι για να διεκπεραιώσουν το αίτημα τους. Η επιλογή των πελατών γίνεται με βάση την υπολογιστική τους ισχύ και την εκτίμηση της, τη χωρητικότητα αποθήκευσης.. βάση αλγορίθμων. Ο ελεγκτής στην υλοποίηση αυτού του αλγορίθμου είναι προκαθορισμένος και ο κάθε χρήστης γνωρίζει τη διεύθυνση του. Η ουσιαστική του λειτουργία εντοπίζεται στη διεπαφή μεταξύ χρήστη και πελάτη. [26]

Figure 1: Basic Architecture of *SYNDEO*

Εικόνα 30: Αρχιτεκτονική αλγορίθμου

6.1.2 Εφαρμογή αλγορίθμου

Ο δοθέν αλγόριθμος βασίστηκε στην αρχιτεκτονική Map Reduce αφού έχουν παρόμοια λειτουργία, με την μόνη διαφορά ότι η δοθέν αρχιτεκτονική υποστηρίζει την ετερογένεια των συσκευών και των υπολοίπων θεμάτων που αναφέρθηκαν στην ενότητα 6.1. Μια από τις κύριες εφαρμογές του αλγορίθμου αυτού εντοπίζεται στον τομέα υγείας ώστε να βελτιωθούν οι υπηρεσίες του και τα τεχνικά θέματα που αντιμετωπίζει όπως η επικοινωνία μεταξύ ασθενών και ιατρών σε πραγματικό χρόνο και η υλοποίηση αλλά και λειτουργία συσκευών υγείας για τη παρακολούθηση των ασθενών.

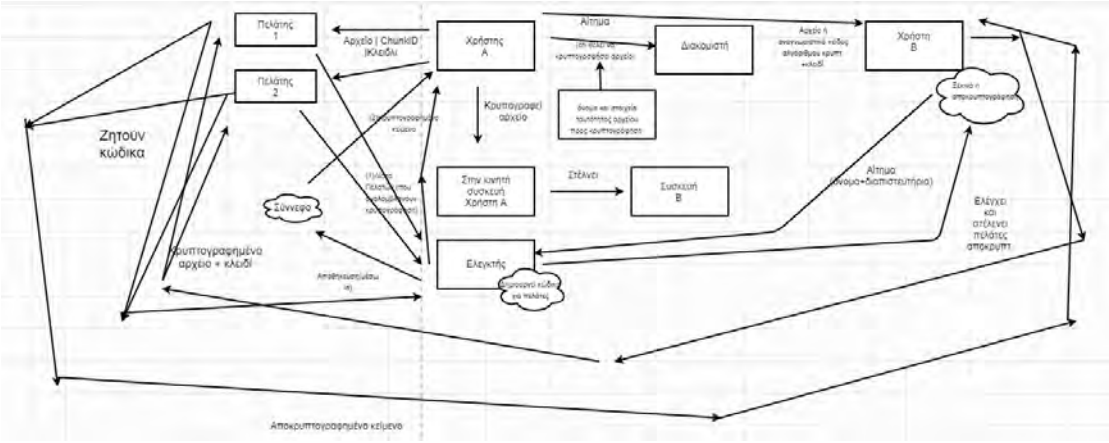
6.1.3 Λειτουργία αλγορίθμου

Προκειμένου να διασφαλιστεί η ενσωμάτωση ετερογενών συσκευών, χρησιμοποιείται ένα πρόγραμμα πελάτη ικανό να λειτουργήσει σε συσκευές IoT.Επίσης όσο αφορά την επικοινωνία των συσκευών παρέχεται μια πλατφόρμα για τη μεταφόρτωση των δεδομένων και τη λήψη αποτελεσμάτων.

Εάν θεωρήσουμε ότι υπάρχουν δύο χρήστες που θέλουν να επικοινωνήσουν μεταξύ τους και να ανταλλάσουν δεδομένα η αναλυτική λειτουργία του αλγορίθμου είναι η εξής:

1. Ο χρήστης Α προκειμένου να στείλει ένα αρχείο στη συσκευή Β, πρέπει να κρυπτογραφήσει το αρχείο αυτό στη κινητή του συσκευή.
2. Ο Α στέλνει αίτημα ότι θέλει να κρυπτογραφήσει ένα αρχείο στον διακομιστή. Η αίτηση περιλαμβάνει το όνομα του αρχείου που θέλει να κρυπτογραφηθεί και στοιχεία ταυτότητας, που θα χρησιμοποιηθούν στην διαδικασία της αποκρυπτογράφησης.
3. Ο ελεγκτής στέλνει στον χρήστη Α μια λίστα με όλους τους πελάτες που μπορούν να αναλάβουν τη κρυπτογράφηση. Ο ελεγκτής διαθέτει τη κατάσταση τη κατάσταση των πελατών και χρησιμοποιεί τα στοιχεία τους για να εκτελέσει τις αιτήσεις. Η επιλογή των πελατών γίνεται με βάση τα χαρακτηριστικά τους.
4. Ο Α χωρίζει το αρχείο σε κομμάτια και τα στέλνει σε πελάτες που μπορούν να αναλάβουν τη κρυπτογράφηση. Το μέγεθος των κομματιών στα αρχεία είναι σε ίσα κομμάτια, ωστόσο σε άλλες υλοποιήσεις εξαρτάται από το τύπο του αλγόριθμου και από το μέγεθος των αρχείων και μπορεί να είναι μεταβλητό το μέγεθος ώστε να διασφαλιστεί η ασφάλεια των δεδομένων. Μαζί τα κομμάτια αρχείου στέλνεται και ένα αναγνωριστικό Chunk ID στους πελάτες για τη συγχώνευση του κρυπτογραφημένου αρχείου αλλά και ένα κλειδί κρυπτογράφησης που έχει δημιουργήσει ο χρήστης.
5. Οι πελάτες στέλνουν στον ελεγκτή για τον κώδικα προγραμματισμού , διότι οι ίδιοι δεν μπορούν να την εκτελέσουν λόγω ότι οι κινητές συσκευές διαθέτουν μικρή χωρητικότητα αποθήκευσης για κώδικα και για την εκτέλεση της αίτησης και αναλαμβάνουν οι ελεγκτές να στείλουν το κατάλληλο κώδικα.
6. Ο ελεγκτής δημιουργεί τον κώδικα για τον πελάτη με βάση την αρχιτεκτονική του.
7. Δημιουργείται το κρυπτογραφημένο κείμενο από τη συνεργασία όλων των πελατών. Όλα τα κομμάτια αρχείων έχουν το ίδιο αναγνωριστικό και στο τέλος στέλνονται όλα μαζί σε έναν πελάτη που είναι υπεύθυνος για τη συγχώνευση και για τη δημιουργία του τελικού κρυπτογραφημένου κειμένου.
8. Ο ελεγκτής στέλνει το κρυπτογραφημένο κείμενο πίσω στο χρήστη Α, είτε αποθηκεύεται στο σύννεφο μέσω ενός αναγνωριστικού-id αρχείου που στέλνεται στον Α.

9. Έπειτα ο Α στέλνει το αρχείο ή το αναγνωριστικό του αρχείου στο Β μαζί με το είδος του αλγορίθμου κρυπτογράφησης και το κλειδί της αποκρυπτογράφησης και έτσι ξεκινάει η διαδικασία αποκρυπτογράφησης.
10. Ο Β στέλνει ένα αίτημα με το όνομα του αρχείου και με διαπιστευτήρια στον ελεγκτή για να πραγματοποιήσει την αποκρυπτογράφηση.
11. Ο ελεγκτής ελέγχει το αίτημα και εάν τα στοιχεία είναι έγκυρα, στέλνει στο Β μια λίστα με τους πελάτες αποκρυπτογράφησης.
12. Ο Β διαιρεί το κρυπτογραφημένο αρχείο και το στέλνει μαζί με το κλειδί της αποκρυπτογράφησης στους πελάτες.
13. Οι πελάτες ζητάνε το κώδικα από τους ελεγκτές.
14. Τέλος στέλνεται το αποκρυπτογραφημένο κείμενο από όλους τους πελάτες στο Β.



Εικόνα 31 : Flowchart Κρυπτογράφησης/Αποκρυπτογράφησης

Επομένως μπορούμε να διακρίνουμε τρείς φάσεις στον αλγόριθμο την εγκατάσταση τα επικοινωνίας, τη κρυπτογράφηση και την αποκρυπτογράφηση. Οι έννοιες τις κρυπτογράφησης και της αποκρυπτογράφησης αναλύθηκαν παραπάνω. Όσο αφορά την εγκατάσταση επικοινωνίας εννοούμε ότι οι πελάτες "ξυπνάνε και στέλνουν ένα μήνυμα "Hello" που περιέχει τη διαμόρφωση του πελάτη και το φόρτο των εργασιών που έχει, στον ελεγκτή. Ο ελεγκτής καταχωρεί τον πελάτη στη λίστα πελατών με μια χρονοσφραγίδα. Το μήνυμα στέλνεται ανά δέκα λεπτά, αν κάποιος πελάτης δεν στείλει μήνυμα ο ελεγκτής τον θεωρεί "νεκρό" και ενημερώνει κατάλληλα τη λίστα. Η λίστα χρησιμοποιείται σε περιπτώσεις περίπλοκων υπολογισμών. [26]

Εικόνα 32 : Flowchart Εγκατάστασης

6.1.4 Αλγόριθμος

Require: ChunkNo(CN), TotalClients(TN), ChunkSize(C),

1: procedure SENDCHUNK

2: Send Request to controller for available client list

3: TN ←getClientsListF romController()

4: Initialization: CN 1

5: Initialization:bufferSize C

6: while (EOF) do

7: ch=read File (fp; bufferSize)

8: clientID=CN%TN

9: send(clienID; ch;CN)

10 : CN + +

11 : end while

12 : end procedure

1: procedure ENCRYPTION/DECRYPTION

2: cm receive Commmand()

3: get CodeFromConroller(cm)

4: while (ch = receiveChunks()) do

5: cipher executeCode(ch; key)

6: end while

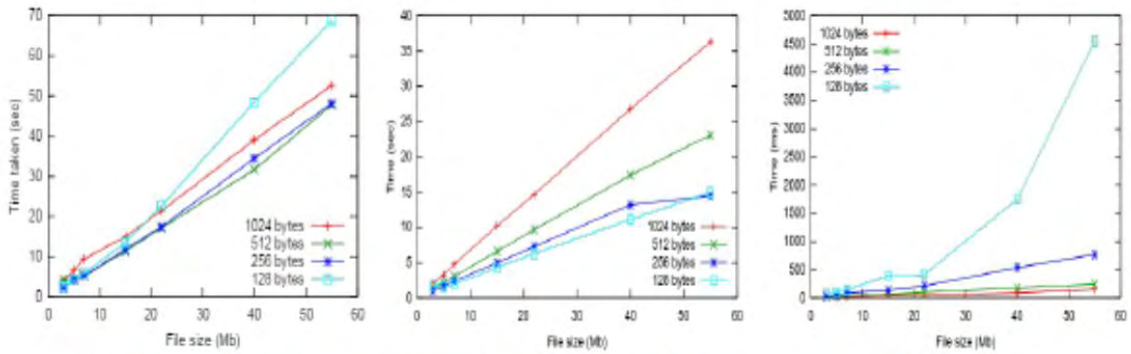
7: send ToMergeClient(cipher)

8: end procedure mobile phone

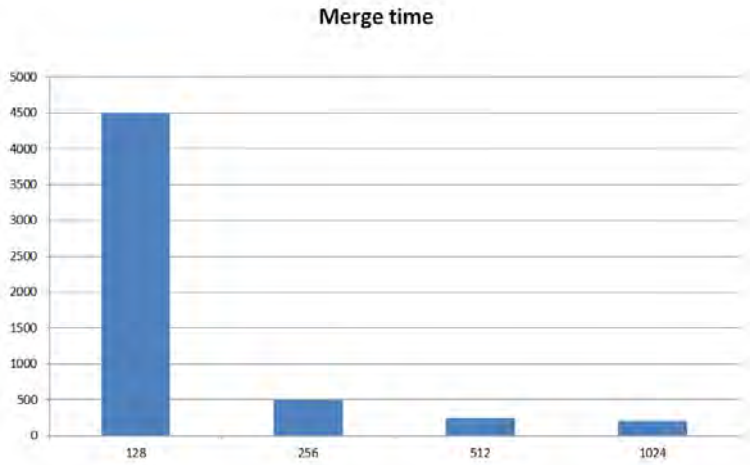
6.1.5 Διαγράμματα και πρακτική προσέγγιση

Προκειμένου να κατανοηθεί η χρήση του δοθέν αλγορίθμους εφαρμόστηκε μια πιο πρακτική προσέγγιση με βάση κάποιες μετρήσεις. Πιο συγκεκριμένα όσο αφορά τα συστατικά μέρη του αλγορίθμου πελάτες αποτελούν φορητοί ηλεκτρινοί υπολογιστές και χρήστες αποτελούν κινητά τηλέφωνα. Ο ελεγκτής και οι πελάτες "τρέχουν" σε κάποια εικονική μηχανή Virtual Machine. Ο αλγόριθμος χρησιμοποιείται για να κρυπτογραφήσει δεδομένα μεγάλου μεγέθους, που κυμαίνετε από 3-55 mb. Ο χρήστης περιμένει από τον ελεγκτή τη λίστα πελατών και την λαμβάνει σε χρόνο 1015. Έπειτα οι χρήστες διαιρούν το αρχείο σε τμήματα από 128-1024 bytes ώστε να τα στείλουν στους πελάτες. Άλλη μια σημαντική διαδικασία είναι η αποστολή προγραμματιστικού κώδικα από τον ελεγκτή στους πελάτες, ο χρόνος που περνάει από την αποστολή έως την λήψη κυμαίνετε από 25 σε 30 ms. Στην εικόνα 33 μπορεί να γίνει αντιληπτή η πολυπλοκότητα του αλγορίθμου που εξαρτάται από το μέγεθος των αρχείων και των μερών της. Στην εικόνα 33(a)και 36 φαίνεται ο συνολικός χρόνος που χρειάζεται για την κρυπτογραφία αρχείων με διαφορετικό μέγεθος . Στην εικόνα 33 (b) και 35υπολογίζεται και πάλι ο χρόνος κρυπτογράφησης αλλά για κάθε κόμβο. Τέλος στην εικόνα 33 (c)και 34παρουσιάζει τον χρόνο που χρειάζεται για τη συγχώνευση των μερών του αρχείου για διάφορα μεγέθη από τους πελάτες. Τα αποτελέσματα των μετρήσεων όσο αφορά την κρυπτογράφηση σε έναν κόμβο πελάτη δείχνουν ότι όσο μεγαλύτερο είναι το αρχείο τόσο αυξάνεται και ο χρόνος κρυπτογράφησης. Αντίθετα στη συγχώνευση τμημάτων αρχείου όσο μικρότερο είναι το μέγεθος των τμημάτων τόσο αυξάνεται χρόνος κρυπτογράφησης. Τέλος

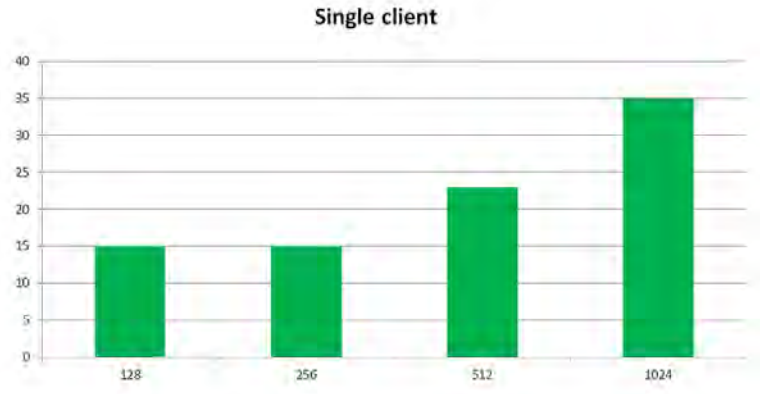
χρησιμοποιήθηκαν τέσσαρα μεγέθη αρχείων 126,256,512 και 1024bytes και το πιο αποδοτικό μέγεθος ήταν το 256 bytes.[26]



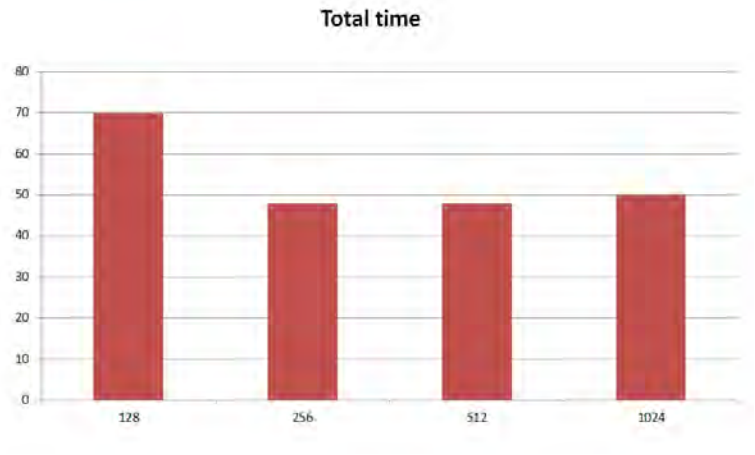
Εικόνα 33 : Διαγράμματα sec/file size : Total Time, Single Client και Merge Time



Εικόνα 34: Ιστογραμματική απεικόνιση merge time



Εικόνα 35: Ιστογραμματική απεικόνιση single client time



Εικόνα 36 : Ιστογραμματική απεικόνιση total time

6.2 Αλγόριθμος για ασφάλεια Fog Computing μέσω κρυπτογράφησης

Το Cloud Computing θεωρείται μια από τις καλύτερες τεχνολογίες χάρις στην ευκαμψία και την επεκτασιμότητα του. Το κύριο ζήτημα το οποίο λαμβάνει χώρα στο σύννεφο, είναι η ασφάλεια. Ωστόσο το ζήτημα αυτό λύνεται με την τεχνική του fogcomputing. Εφόσον υπάρχουν ζητήματα ασφάλειας στο υπολογιστικό ακόμη και μετά τη λήψη των κρυπτογραφημένων δεδομένων από το cloud computing, εφαρμόστηκε η διαδικασία της κρυπτογράφησης μέσω της χρήσης AES αλγορίθμου. Έπειτα από ανάλυση προκύπτει ότι ο AES αλγόριθμος είναι η πιο ασφαλείς διαδικασία κρυπτογράφησης για ασφάλεια.

Στο σύγχρονο κόσμο κάθε οργανισμός, είτε μεγάλης είτε μικρότερης κλίμακας βιομηχανίες, βασίζονται στην διαδικασία του cloud computing για την αποθήκευση των δεδομένων τους. Επίσης βασίζονται στη διαδικασία αυτή για το πως να χρησιμοποιούν ντους πόρους σύμφωνα με τις απαιτήσεις τους. Το cloud παρέχει αμοιβή ανά έννοια χρήσης. Ο αριθμός των συσκευών που είναι συνδεδεμένες στο Διαδίκτυο πλέον είναι διπλάσιος σε σχέση με τον παγκόσμιο πληθυσμό.

Στο fog computing οι χρήστες ειδοποιούνται σχετικά με το τι είναι οι ενέργειες που πρέπει να ληφθούν για τα δεδομένα. Σε αυτό το σημείο τα αναλυτικά στοιχεία

εφαρμόζονται στα ληφθέντα δεδομένα και αποθηκεύονται στο cloud. Στη διαδικασία του fog computing η εφαρμογή έρχεται στα δεδομένα και όχι στα δεδομένα των εφαρμογών. Είναι γεγονός πως ο αριθμός των συσκευών που συνδέονται με το διαδίκτυο αυξάνεται με γρήγορη ταχύτητα και η πρόοδος στο IoT έχει οδηγήσει και αυτή σε αυτή τη δραστική αύξηση. Το Internet of Things έχει οδηγήσει την εξέλιξη στο fog computing λόγω του αυξημένου αριθμού συσκευών οι οποίες παράγουν τεράστιο όγκο δεδομένων. Η αποθήκευση δεδομένων στο νέφος και η ανάκτηση θα είναι εξαιρετικά δύσκολες. Ως εκ τούτου, η ομίχλη έχει εισαχθεί.

Το υπολογιστικό νέφος παρέχει διάφορες υπηρεσίες για αποθήκευση και πρόσβαση στα δεδομένα στα οποία το κύριο πρόβλημα είναι η μη παροχή ασφάλειας των δεδομένων από τους επιτιθέμενους. Δεν παρέχεται οποιοδήποτε επίπεδο διαβεβαίωσης για τον χρήστη σχετικά με την ασφάλεια των δεδομένων του. Ως εκ τούτου, η ανάπτυξη ενός πιο ασφαλούς νέφους δεν είναι αρκετή επειδή θα υπήρχαν συνεχείς επιθέσεις που συμβαίνουν στο σύννεφο και υπάρχουν πιθανότητες τα δεδομένα να διαρρεύσουν ή να χαθούν για πάντα.
Επί του παρόντος στο υπολογιστικό νέφος υπάρχει, το σύστημα Decoy που χρησιμοποιείται ως τρέχον σύστημα ασφαλείας για την έγκριση των δεδομένων. Το σύστημα Decoy λέγεται ότι είναι ένα σύστημα εξαπάτησης στο οποίο τα φωνητικά στοιχεία είναι ρυθμισμένα για να δελεάσουν μη εξουσιοδοτημένους χρήστες, δίνοντας τρωτά σημεία ενός συστήματος περιορίζοντας έτσι την μη εξουσιοδοτημένη πρόσβαση στο δίκτυο. Αυτό το σύστημα αποτελείται από ψευδείς αρχεία με τα ευαίσθητα ονόματα, όπως ο αριθμός κοινωνικής ασφάλισης και λεπτομέρειες πιστωτικών καρτών ως ονόματα αρχείων. Μόλις το αρχείο κατέβει, η ειδοποίηση θα δημιουργηθεί και το σύστημα θα ειδοποιηθεί για την επίθεση. Αυτή η μέθοδος του συστήματος είναι ενσωματωμένη με τη συμπεριφορά χρηστών όπου υπάρχει η μη εξουσιοδοτημένη πρόσβαση θα ειδοποιηθεί στο σύστημα. [26]

6.2.1Αρχιτεκτονικήαλγορίθμου

Το πρότυπο κρυπτογράφησης δεδομένων (DES) ήταν από τα ευρύτερα χρησιμοποιημένα πρότυπα κρυπτογράφησης, το οποίο χρησιμοποιεί συμμετρικό κλειδί αλγόριθμο κρυπτογράφησης δεδομένων. Ο DES έχει 56 bits του μεγέθους

κλειδιού και ενώ το μέγεθος του μπλοκ είναι 64 bit Για πολλές εφαρμογές όταν θεωρείται DES είναι η πιο ανασφαλής τεχνική. Αυτό είναι λόγω του μεγέθους του κλειδιού το οποίο είναι 56 bit. Το κλειδί του DES έχει παραβιαστεί πολλές φορές σε σύντομο χρόνο, αυτό δείχνει πόσο αδύναμος είναι ο αλγόριθμος. Μερικές από τις επιθέσεις αυτές θα μπορούσαν να σπάσουν το κλειδί γρηγορότερα από τη Brute Force είναι η Διαφορική Κρυπτοαναλύση, Γραμμική Κρυπτογραφία.

Ο προκάτοχος του αλγορίθμου DES είναι το 3DES που ονομάζεται Τριπλό Πρότυπο Κρυπτογράφησης Δεδομένων, όπου 3 υποδείγματα DES είναι κλιμακωτά. Το Triple DES δεν έκανε αλλαγές στον προηγούμενο αλγόριθμο DES εκτός από την αύξηση του μεγέθους του κλειδιού, όπου μπορεί να έχει 56 ή 112 ή168 bits, λόγω της αύξηση της υπολογιστικής δύναμης που έκανε τη Brute Force εύκολη και το μέγεθος του μπλοκ παραμένει ίδιο με 64 bits ως DES. Το τριπλό DES ήταν 2½ φορές περισσότερο ασφαλές από τον αλγόριθμο DES. Ακόμη όμως και το Triple DES είναι ευάλωτο στις επιθέσεις ασφαλείας που συναντώνται στη μεσαία επίθεση. Εφόσον ο αλγόριθμος DES σχεδιάστηκε για υλοποίηση υλικού, όπου εκεί δεν είναι αξιόπιστο κατά τον ίδιο τρόπο το Triple DES δεν λειτουργεί σωστά στις εφαρμογές λογισμικού.

Για να ξεπεραστεί το παραπάνω πρόβλημα το πρότυπο κρυπτογράφησης (AES) θεωρείται περισσότερο αποτελεσματικό. Θεωρείται το πιο προηγμένο και ασφαλές πρότυπο για κρυπτογράφηση ηλεκτρονικών δεδομένων. Το AES θεωρείται διάδοχος του DES που χρησιμοποιεί προκαθορισμένη συμμετρική κρυπτογράφηση κλειδιών. Η AES δέχεται το μέγεθος κλειδιού 128,192, 256 bit μεγέθους. Τα 128bits έχουν ήδη ληφθεί υπόψη ως άθραυστα. Στη σύγκριση όλων των διαθέσιμων αλγορίθμων κρυπτογράφησης, η AES θα είναι η καλύτερη και ο πιο ασφαλής τύπος αλγορίθμου που θα μπορούσε να εφαρμοστεί στα συστήματα fog computingκαι μέχρι τώρα δεν έχει προταθεί τεχνική κρυπτογράφηση για ασφάλεια στους υπολογιστές ομίχλης

Όσο ο αριθμός των συσκευών που είναι συνδεδεμένες στο διαδίκτυο αυξάνεται θα υπάρχει πρόβλημα στην αποθήκευση καθώς και στη διαδικασία ανάκτησης πληροφοριών. Για να λυθεί το πρόβλημα εισήχθηκε το fog computing. Στο cloud computing όλα τα δεδομένα που παράγονται από τους χρήστες αποθηκεύονται απευθείας στο cloud και στη συνέχεια αναλύονται μαζικά, με αναλυτικά στοιχεία πάνω σε αυτό και αποφάσεις που λαμβάνονται για να ενεργεί με δεδομένα.

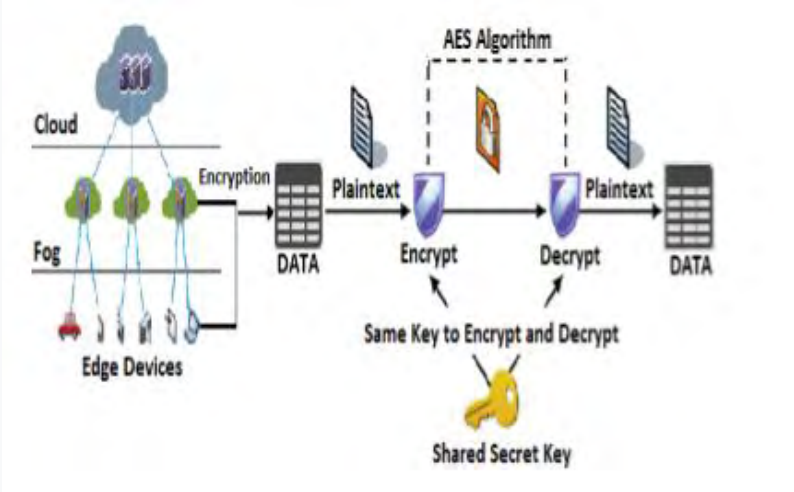
Η ασφάλεια είναι ένα από τα σημαντικότερα προβλήματα, διότι υπάρχουν πολλά ευαίσθητα δεδομένα γύρω μας .Θα μπορούσε να είναι οποιαδήποτε εταιρεία

λεπτομέρειες τιμολόγησης, ή ακόμη θα μπορούσε να είναι εθνικό μυστικό. Όλα τα δεδομένα πρέπει να είναι ασφαλή και πρέπει να βεβαιώνονται ότι έχει όλες τις απαραίτητες μεθόδους σε αυτό, πράγμα που κάνει σε έναν εισβολέα δύσκολο να σπάσει το κλειδί. Η ασφάλεια και το απόρρητο των δεδομένων είναι βασικό κομμάτι. Στο σύστημα του cloud, αν και τα ασφαλή δεδομένα αποστέλλονται στο Fog από σύννεφο, προβλέποντας την απειλή ασφάλειας στην ομίχλη όπως ο άνθρωπος στις μεσαίες επιθέσεις, στόχος είναι να προστεθεί ένα δεύτερο επίπεδο ασφάλειας στο επίπεδο της ομίχλης. [26]

6.2.2 Λειτουργία αλγορίθμου

Στην ομίχλη, το σύστημα Decoy θεωρείται ως μοντέλο ασφαλείας, όπου ο χρήστης του συστήματος πρώτα κάνει εγγραφή. Μετά δίνει τα στοιχεία σύνδεσης του και έπεται πρέπει να απαντήσει σε κάποια ερώτηση ασφαλείας που του δόθηκε κατά τη δημιουργία του λογαριασμού του.

Τα Decoyσυστήματα είναι μια άλλη μέθοδος για την παγίδευση εισβολών με τα ψεύτικα αρχεία, προσπαθώντας να τους εντοπίσει αλλάζοντας το όνομα του αρχείου σε κάποιο λανθασμένο. Οι χρήστες που γνωρίζουν τα δεδομένα του αρχείου μπορούν να καταλάβουν εάν είναι ψεύτικο ή όχι ενώ ο εισβολέας δεν γνωρίζει η διαφορά και μόλις ανοίξει το αρχείο και προσπαθήσει να το κατεβάσει, θα ειδοποιηθεί το σύστημα. Η μέθοδος αυτή δεν είναι κατάλληλη διότι ο επιτιθέμενος μπορεί να προβλέψει ή να υποθέσει την ερώτηση ασφαλείας με αποτέλεσμα να αποκτήσει πρόσβαση στα δεδομένα. Ως λύση χρησιμοποιούμε τον αλγόριθμο AES, όπου τα δεδομένα θα είναι κρυπτογραφημένα έτσι ώστε ακόμα και αν ο εισβολέας θέλει να έχει πρόσβαση στα δεδομένα από την αρχιτεκτονική Decoy, η κρυπτογράφηση τον δυσκολεύει περισσότερο. Υπάρχουν πιθανότητες υποκλοπής από το σύννεφο έως την ομίχλη αλλά προσθέτοντας την κρυπτογράφηση στα ήδη κρυπτογραφημένα δεδομένα , η υποκλοπή είναι αδύνατη και ξεπερνιέται ο κίνδυνος για επίθεση man in the middle. [26]



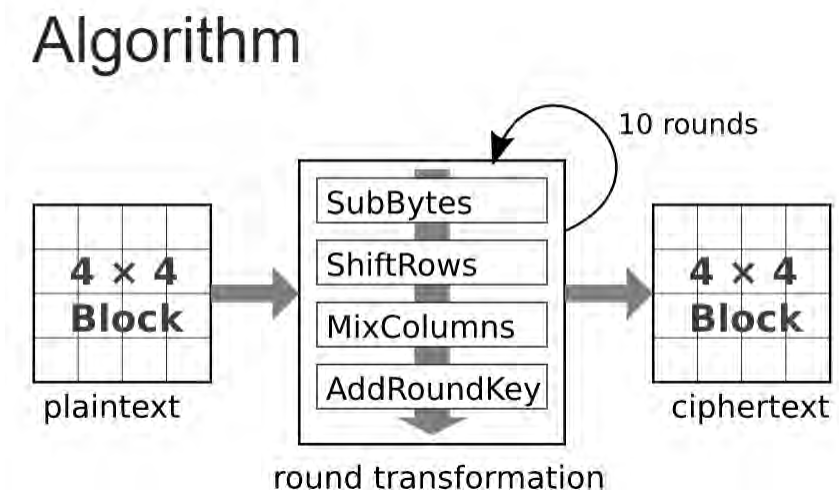
Εικόνα 37: Αρχιτεκτονική συστήματος

Στην εικόνα 37 έχει επιλεγεί τεχνική κρυπτογράφησης ώστε τα δεδομένα σε fogcomputingνα έχουν μεγαλύτερη ασφάλεια. Ο ερευνητικός στόχος είναι να επιτευχθεί ασφάλεια στο fog η οποία είναι το δεύτερο επίπεδο του συστήματος fog με τη χρήση του AES αλγόριθμου κρυπτογράφησης και την εφαρμογή του πάνω από το επιλεγμένα σύνολα δεδομένων μέσω της ανάπτυξης σε μια κινητή συσκευή και συνεπώς να συλλέξει τις μετρήσεις απόδοσης. Τρία σύνολα δεδομένων έχουν δημιουργηθεί για να αξιολογήσουν τις καλύτερες και τις χειρότερες περιπτώσεις σε όλες τις πτυχές των συνόλων δεδομένων. Για την ανάλυση του σε περιβάλλον ομίχλης, η edge συσκευή του κινητού και η εφαρμογή θεωρείται και έχει σχεδιαστεί έτσι ώστε να κρυπτογραφεί και να αποκρυπτογραφεί τα σύνολα δεδομένων που επιλέγονται χρησιμοποιώντας την τεχνική κρυπτογράφησης του AES που είναι μια συμμετρική κρυπτογράφηση κλειδιών κάνοντας χρήση ενός κοινού μυστικού κλειδιού για κρυπτογράφηση και αποκρυπτογράφηση. Διαφορετικά σύνολα δεδομένων που έχουν διαφορετικά μεγέθη δεδομένων και το κείμενο, οι συμβολοσειρές και οι εικόνες επιλέγονται για τη δοκιμή του στη μέθοδο κρυπτογράφησης. Η απόδοση αξιολογείται γι 'αυτά τα σύνολα δεδομένων που δοκιμάζονται στην άκρη της κινητής συσκευής. Κατά την αξιολόγηση διαφόρων παραγόντων όπως η κρυπτογράφηση, χρόνος αποκρυπτογράφησης, χρήση μνήμης, χρόνος απόκρισης για κάθε σειρά δεδομένων, οι καλύτερες και οι χειρότερες πιθανές περιπτώσεις παρακολουθούνται. Με άλλα λόγια, ο κύριος στόχος της έρευνας είναι να παρέχει ασφάλεια στο δεύτερο στρώμα του συστήματος σύννεφων-ομίχλη χρησιμοποιώντας τη τεχνική της κρυπτογράφησης AES. Ακουστική συσκευή κινητή

με 3 σειρές δεδομένων επιλέγεται για τη δοκιμή της περίπτωσης ασφάλειας σε ομίχλη. Τέλος, απόδοση για κάθε πτυχή του συνόλου δεδομένων σε σχέση με μετρήσεις όπως ο χρόνος, η χρήση της μνήμης, ο χρόνος απόκρισης αξιολογείται και οι καλύτερες, χειρότερες περιπτώσεις παρακολουθούνται. [26]

6.2.3 Αλγόριθμος

Το AES είναι γνωστό για την αρχή βασισμένη στο σχεδιασμό που έχει υποκατάσταση και μεταλλάξεις και λέγεται ότι είναι γρήγορη τόσο στο λογισμικό όσο και στο υλικό. Έχει σταθερό μέγεθος μπλοκ 128 bit και μέγεθος κλειδιού 128, 192 ή 256 bits. Οι AES λειτουργούν με την κύρια σειρά 4x4.
10 κύκλοι επανάληψης για κλειδί 128 bit
12 κύκλοι επανάληψης για κλειδί 192 bit
14 κύκλοι επανάληψης για πλήκτρο 256 bit
Υπάρχουν τέσσερις γύροι βημάτων που εκτελούνται στο
σύνολο δεδομένων που απεικονίζονται στην εικόνα 35. [26]



Εικόνα 38 : Αλγόριθμος

TheSubBytesstep:

Το μέγεθος του πίνακα αναζήτησης είναι 16 × 16. Το υποκατάστατο byte για δεδομένο θα μπορούσε να βρεθεί διαιρώντας το byte σε δύο 4-bit πρότυπο, με

αποτέλεσμα μια ακέραια τιμή από 0 έως 15. Αυτά θα μπορούσαν να αναπαρασταθούν από δεκαδικό τιμές από 0 έως F. Όταν χρησιμοποιείται ένα από αυτά ο δείκτης γραμμών και ένας άλλος χρησιμοποιείται για τη στήλη

για να μπείτε στον πίνακα 16 × 16 Lookup. Κάθε βήμα SubBytes του συνόλου δεδομένων αντικαθίσταται με τον πίνακα αναζήτησης 8-bit. Το βήμα αντικατάστασης επικεντρώνεται στη μείωση της συσχέτισης μεταξύ bit εισόδου και εξόδου σε byte επίπεδο. Ο αλγόριθμος του σταδίου SybBytes step:

```
VoidSubByte(byte[][] state) {
    for (intrw=0; rw<4: rw++)
        for (int cl=0; cl<N;cl++)
state[rw][cl]=SBox[state[rw][cl]];
}
```

The ShiftRows step:

Ο μετασχηματισμός ShiftRow συμπεριφέρεται ως εξής

1) Δεν θα μετατοπίσει καθόλου τον πίνακα κατάστασης στην πρώτη σειρά.
2) Η κυκλική δεύτερη σειρά θα μετατοπιστεί κατά ένα byte προς τα αριστερά.
3) Στην τρίτη σειρά κυκλικά μετατοπίζεται δύο byte προς τα αριστερά.
4) Στην τέταρτη σειρά θα μετατοπίσει κυκλικά τρία bytes προς τα αριστερά.

Στο Shift το βήμα κάθε σειράς θα αντικατασταθεί στα αριστερά του ανάλογα με τον δείκτη της σειράς. Με τον ίδιο τρόπο για την αποκρυπτογράφηση, οι αντίστοιχες σειρές θα είναι μετατοπισμένες προς την αντίθετη κατεύθυνση. Η πρώτη σειρά παραμένει αμετάβλητη, στη δεύτερη σειρά η σειρά θα μετακινηθεί προς τα δεξιά από ένα byte. Στη τρίτη σειρά θα μετακινήθηκαν προς τα δεξιά με 2 bytes και στην τέταρτη σειρά κάνουν shift σε 3 byte προς τα δεξιά. Ο αλγόριθμος του σταδίου ShiftRow:

```
VoidShiftRow(byte[ ][ ] state) {
```

```
byte[ ] s= newbyte[4];
        for (int t=1; t<4;t++)
                for (int d=0; d<N;d++)
s[d]=state[t](d+t])%N];
for(int d=0;d<N;d++)
        state[t][d]=s[d];
        }
}
```

The MixColumns step:

Στη στήλη Mix κάθε byte της στήλης στο σύνολο δεδομένων αντικαθίσταται από τη λειτουργία όλων των bytes στην υπάρχουσα στήλη. Το πιο σημαντικό, κάθε byte στη στήλη θα αντικατασταθεί από τις δύο φορές αυτού του byte, συν τρεις φορές του επόμενου byte, συν το byte που έρχεται στη συνέχεια, συν το byte που ακολουθεί. Ο αλγόριθμος του σταδίουMixColumn:

```
Void MixColumn(byte[ ][ ] st) {
byte [ ] p= newbyte[4];
for (int cl=0; cl<4;cl++) {
        p[0]=(0x02 #st[0][c] ^ (0x03 # st[1][c]) ^st[2][c] ^st[3][c];
        p[1]= st[0][c] ^ (0x02 # st[1][c]) ^ (0x03 # st[2][c]) ^ st[3][c];
        p[2]= st[0][c] ^ st[1][c] ^ (0x02 # st[2][c]) ^ (0x03 # st[1][c]);
        p[3]=(0x03 # st[0][c]) ^ st[1][c] ^ st[2][c] (0x02 # st[3][c]);
                for(int j=0; j<4;j++)
                        st[i][cl]=p[j];
```

}}}

The AddRoundKey step:

Εδώ κάθε ένα από τα byte συνδυάζεται με bytes του AddRoundKey με πράξηXOR. Ο αλγόριθμος του σταδίουAddRoundKey

Void AddRoundKey (byte[][]sta)

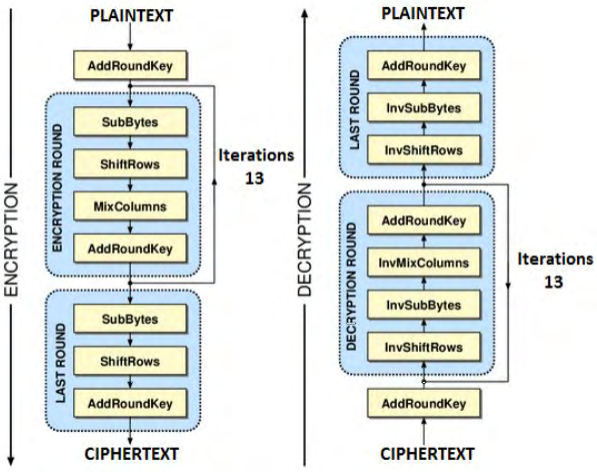
{

For(int cl=0;cl<N;cl++)

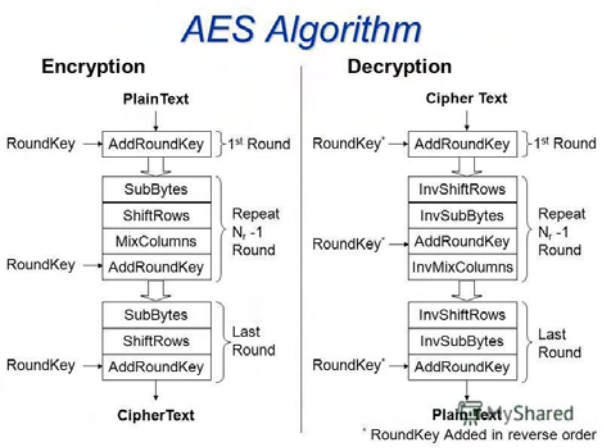
For(intrw=0;rw<4;rw++)

Sta[r][c] = sta[r][c]^n[nCount++]

}



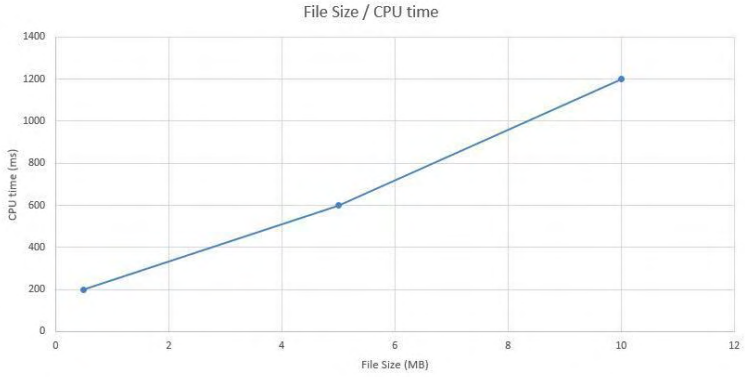
Εικόνα 39 : Αρχιτεκτονική αλγόριθμου AES



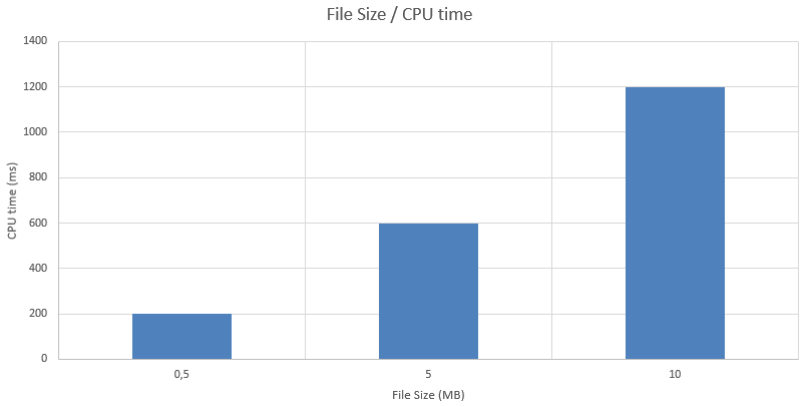
Εικόνα 40 : Σχηματική απεικόνιση κρυπτογράφησης και αποκρυπτογράφησης στον αλγόριθμο AES

6.2.4 Διαγράμματα και πρακτική προσέγγιση

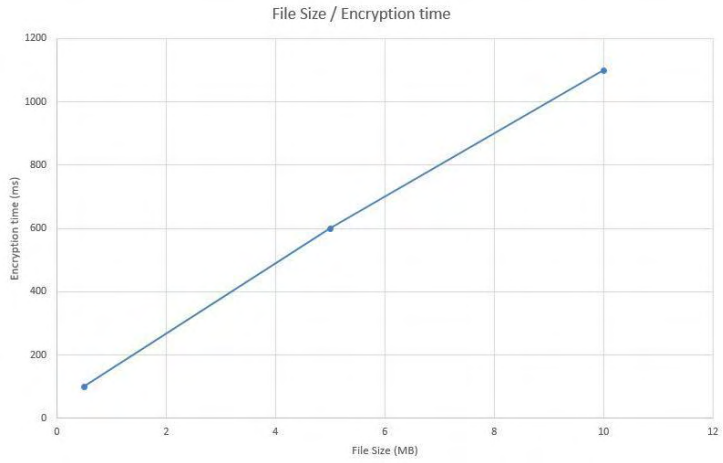
Για την καλύτερη κατανόηση του μοντέλου εφαρμόστηκαν μετρήσεις στην κρυπτογράφηση AESσε δεδομένα που ήδη είναι κρυπτογραφημένα, εφαρμοσμένος σε συσκευές ομίχλης. Όσο αφόρα το μέγεθος αρχείων σε με σχέση τον χρόνο εκτέλεσης στη CPU (εικόνα 41 και 42). Επίσης το χρόνο κρυπτογράφησης για διάφορα μεγέθη αρχείων, παρουσιάζεται (εικόνα 43 και 45), αλλά και το χρόνο αποκρυπτογράφησης για διάφορα μεγέθη αρχείων (εικόνα 44 και 45). Τέλος υπολογίστηκε ο χρόνος που χρησιμοποιείται η μνήμη για τα διάφορα αρχεία (εικόνα 46 και 47). [26]



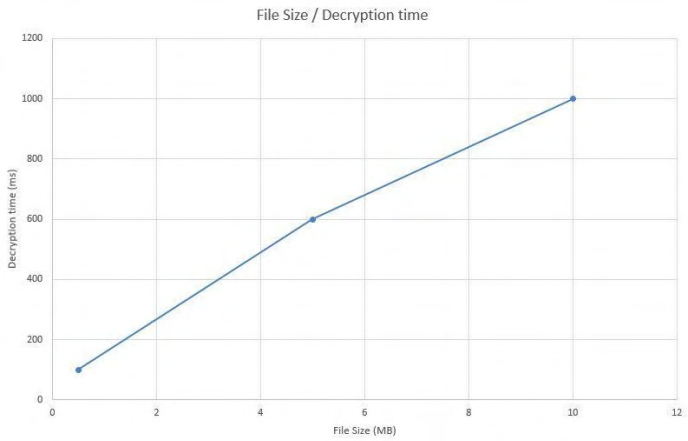
Εικόνα 41: Μέτρηση χρόνουCPU.



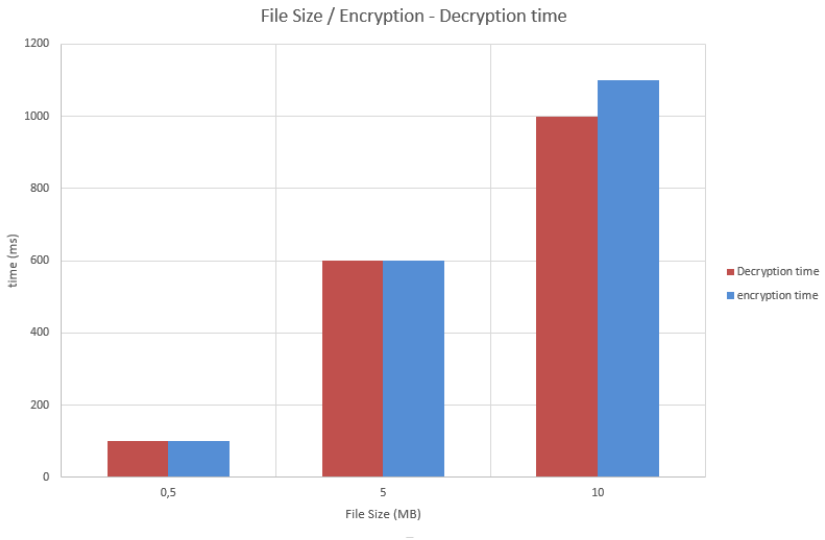
Εικόνα 42: Ιστογραμματική απεικόνιση μέτρησης CPU.



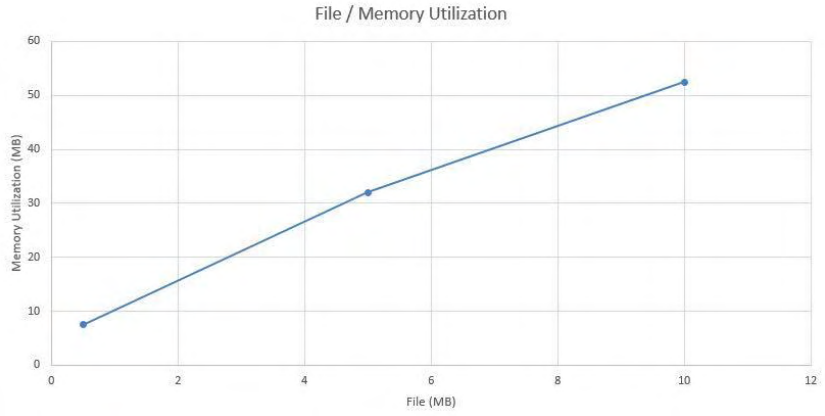
Εικόνα 43: Μέτρηση χρόνου κρυπτογράφησης.



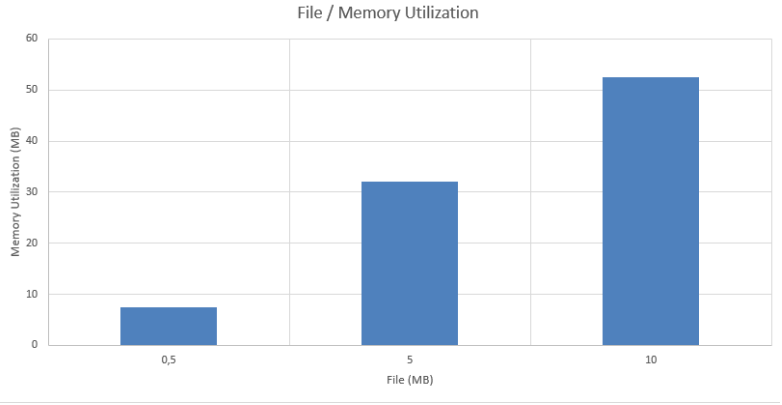
Εικόνα 44: Μέτρηση χρόνου αποκρυπτογράφησης.



Εικόνα 45: Ιστογραμματική απεικόνιση κρυπτογράφησης και αποκρυπτογράφησης



Εικόνα 46: Μέτρηση χρόνου χρήσης της μνήμης.



Εικόνα 47: Ιστογραμματική απεικόνιση χρόνου χρήσης της μνήμης

6.3Αλγόριθμος ιδιωτικότητας Lightweight Privacy-Preserving Data Aggregation σε περιβάλλον Fog Computing και IoT

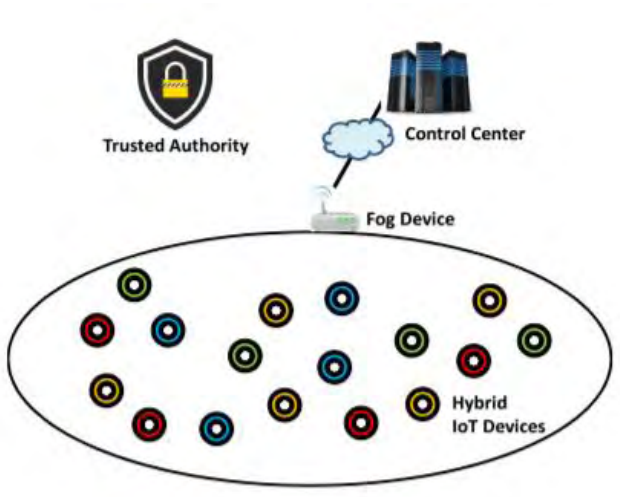
Ο δοθέν αλγόριθμος είναι επικεντρωμένος στις συσκευές IoT εντός του δικτύου ομίχλης. Η προστασία των ιδιωτικών δικαιωμάτων είναι μια από τις πιο σημαντικές εφαρμογές στο fog computing. Η εξέλιξη των συσκευών IoT έχει δημιουργήσει προβλήματα, λόγω των ποικίλων IoTεφαρμογών που παράγουν έναν τεράστιο όγκο δεδομένων σε πραγματικό χρόνο. Επομένως χρειάζονται εξειδικευμένες τεχνικές εξόρυξης δεδομένων. Επίσης εντοπίζεται δυσκολία στη μεταφορά των δεδομένων από συσκευές IoTστο κέντρο ελέγχου, διότι η μεταφορά αυτή η κοστίζει αρκετά και ακόμη χειρότερα εάν τα δεδομένα είναι ψευδή.

Έχουν δημιουργηθεί ποικίλα σχήματα για την προστασία της ιδιωτικότητας, ωστόσο τα περισσότερα δεν αφορούν τα δεδομένα υβριδικών συσκευών και τη συγκέντρωση τους σε μια συσκευή IoT. Τη λύση την έδωσε ο αλγόριθμος Lightweight Privacy-preserving Data Aggregation-LPDA όπου χρησιμοποιείται η κρυπτογράφηση Paillier, το κινέζικο θεώρημα Chinese Remainder Theorem για τη συγχώνευση των δεδομένων στις υβριδικές συσκευές και συναρτήσεις κατακερματισμού one-way hash chain για την ύπαρξη φίλτρου στην άκρη του δικτύου, ώστε να ελεγχθεί η εγκυρότητα των δεδομένων.

Αξίζει να τονιστεί ότι η υλοποίηση αυτού του μοντέλου απαιτεί να επιτευχθούν τέσσερις τομείς στο υπολογιστικό νέφος η ασφάλεια, η ιδιωτικότητα, ανθεκτικότητα σε σφάλματα και η αποδοτικότητα. Η ασφάλεια μπορεί να εντοπιστεί στο φιλτράρισμα στην άκρη του δικτύου και μπορεί να αποτρέψει επιθέσεις τύπου injection attack.Η ιδιωτικότητα εντοπίζεται στην προστασία των δεδομένων στο κέντρο ελέγχου. Το σύστημα πρέπει να είναι ανεκτικό στα σφάλματα σε περιπτώσεις όπου οι συσκευές IoTσταματήσουν την λειτουργία τους, το κέντρο ελέγχου να συνεχίσει τους υπολογισμούς στα δεδομένα. Τέλος η αποδοτικότητα εστιάζεται στην μείωση του κόστους σε υπολογισμούς και σε επικοινωνίες στο δίκτυο ομίχλης. Ο σχηματισμός LPDA, θεωρήθηκε ασφαλής και πολύ πιο ελαφρύς από άλλα παρόμοια μοντέλα. [23], [33]

6.3.1 Αρχιτεκτονική του αλγορίθμου

Το υβριδικό δίκτυο IoT αποτελείται από τρία συστατικά μέρη όπως παρουσιάζεται και στην εικόνα 48 τις υβριδικές συσκευές IoT, μια συσκευή ομίχλης και ένα κέντρο δεδομένων. Όσο αφορά τις υβριδικές συσκευές IoT λόγω της ετερογένειας τους μπορούν να κατηγοριοποιηθούν. Μια συσκευή ομίχλης χρησιμοποιείται στο να παρέχει επικοινωνία μεταξύ των συσκευών και του κέντρου ελέγχου και πιο συγκεκριμένα υλοποιεί κάποιες συγκεκριμένες λειτουργίες ώστε να προωθούνται τα δεδομένα από τις συσκευές IoT στο κέντρο ελέγχου και να ελέγχονται τα δεδομένα από το κέντρο ελέγχου ώστε να είναι ασφαλής. Το κέντρο ελέγχου διαθέτει δεδομένα από τις συσκευές IoT που στέλνονται μέσω της συσκευής ομίχλης. Η χρήση του εντοπίζεται στις αναλύσεις δεδομένων ανά σύνολο λόγω της ετερογένειας που εμφανίζουν. Στην αρχιτεκτονική του αλγορίθμου απαιτείται να υπάρχει εμπιστοσύνη, ώστε το σύστημα να είναι αξιόπιστο. Για να διασφαλιστεί η εμπιστοσύνη υπάρχει ανάλογη διαχείριση των υλικών του συστήματος και η εκχώρηση κλειδιών σε όλα τα μέρη του δοθέν μοντέλου συσκευές IoT, συσκευή ομίχλης και κέντρο ελέγχου. [23], [33]

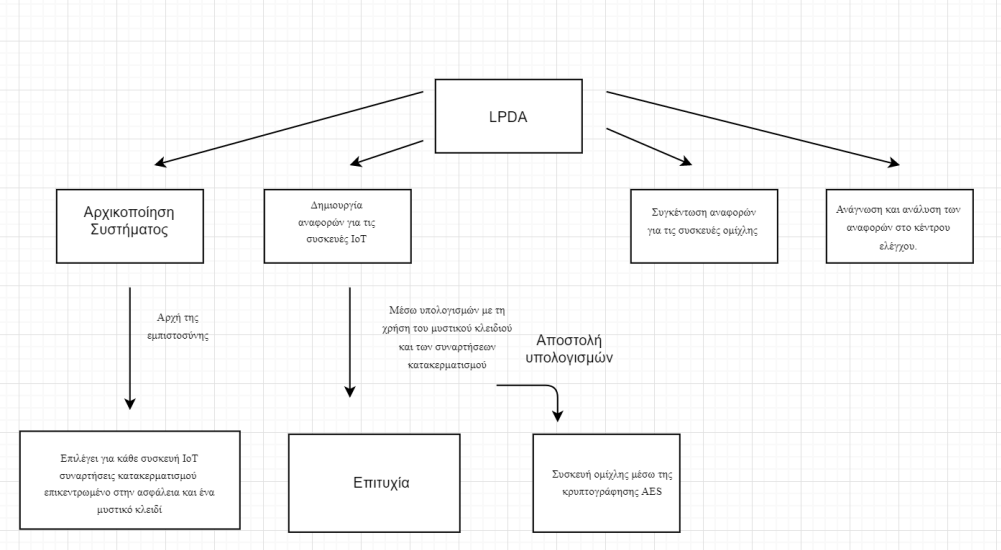


Εικόνα48: ΑρχιτεκτονικήLightweight Privacy-preserving Data Aggregation

6.3.2 Λειτουργία αλγορίθμου

Το μοντέλο LPDA αποτελείται από τέσσερα κύρια μέρη την αρχικοποίηση του συστήματος, τη δημιουργία αναφορών για τις συσκευές IoT, τη συγκέντρωση αναφορών για τις συσκευές ομίχλης και την ανάγνωση και ανάλυση των αναφορών στο κέντρου ελέγχου. Η αρχικοποίηση του συστήματος είναι επικεντρωμένη στην αρχή της εμπιστοσύνης, που αποτελεί συστατικό μέρος του δοθέν μοντέλου και επιλέγει για κάθε συσκευή IoT συναρτήσεις κατακερματισμού επικεντρωμένους στην ασφάλεια και ένα μυστικό κλειδί. Για τη συσκευή ομίχλης εκχωρείται ένα τυχαίο κοινό κλειδί μαζί με το κέντρο ελέγχου που ανατίθεται στις συναρτήσεις κατακερματισμού. Επίσης ανατίθενται στη συσκευή ομίχλης μυστικά κλειδιά και δημόσιου παράμετροι. Τέλος στο κέντρο ελέγχου υπάρχει το κοινό κλειδί που προαναφέρθηκε και μυστικά κλειδιά και δημόσιου παράμετροι. Η δημιουργία των αναφορών για συσκευές IoT επιτυγχάνεται μέσω υπολογισμών με τη χρήση του μυστικού κλειδιού και των συναρτήσεων κατακερματισμού. Οι υπολογισμοί στέλνονται στη συσκευή ομίχλης μέσω της κρυπτογράφησης AES.Το τρίτο στάδιο αφορά τη συγκέντρωση των αναφορών από τις συσκευές IoT. Η συσκευές ομίχλης ελέγχουν την εγκυρότητα των δεδομένων που λάβανε, αν δεν τα έχουν λάβει ξανά τα κρατάνε, διαφορετικά τα απορρίπτουν. Μετά την έγκριση συγκεντρώνονται τα δεδομένα από όλες τις συσκευές και με τη χρήση του μυστικού κλειδιού ενοποιούνται και στέλνονται στο κέντρο ελέγχου.

Σχετικά με την ασφάλεια στο μοντέλο LPDA, μπορούν να αποτραπούν εξωτερικές επιθέσεις τύπου injection attacks. Προκειμένου να ελεγχθούν τα δεδομένα με βάση την αρχή της αυθεντικοποίησης εφαρμόζεται η συνάρτηση κατακερματισμού για κάθε συσκευή IoT. Η συσκευή ομίχλης μπορεί να επικοινωνήσει με κάθε συσκευή IoTεάν αποκριθεί σε συγκεκριμένο χρονικό περιθώριο, εάν αποκριθεί μετά το χρονικό περιθώριο μπορεί να πρόκειται για συσκευή εισβολέα που στοχεύει σε επίθεση injection attack και δεν γνωρίζει τα μυστικά κλειδιά. Επίσης το δοθέν σχήμα στοχεύει στην επίτευξη της ιδιωτικότητας με τη χρήση κρυπτογραφημένων κειμένων μέσω της κρυπτογράφησης Paillier. [23], [33]



Εικόνα 49:Flowchart LPDA

6.3.3Αλγόριθμος

Στο δοθέν σχήμα χρησιμοποιείται το κινέζικο θεώρημα-Chinese Remainder Theorem, αποτελεί θεώρημα αριθμών που χρησιμοποιείται κυρίως για τον υπολογισμό μεγάλων ακεραίων και βασίζεται στην μοναδική λύση οποιοδήποτε ζευγαριού module, ώστε να δημιουργεί ένα σχήμα υβριδικών IoTσυσκευών σε ένα δίκτυο ομίχλης:

Δοθέντος n1,n2,…,nk θετικοί ακέραιοι όπου στον καθένα δίνεται ένας αυθαίρετος ακέραιος a1,a2,..,ak τότε υπάρχει το σύστημα:

$x \equiv a1 \bmod n1$

$x \equiv a2 \bmod n2$

…….

$x \equiv ak \bmod nk$

Όπου η λύση του είναι ένα μοναδικό $modN=n1*n2*…nk$

Επίσης χρησιμοποιείται η συνάρτηση κατακερματισμού One Way Hash Chain, αποτελεί μια τεχνική ασφαλείας για τον έλεγχο της ταυτότητας στα δεδομένα. Στο παρόν σχήμα προτείνετε ο ελαφρύς έλεγχος ταυτότητας σε υβριδικές συσκευές.

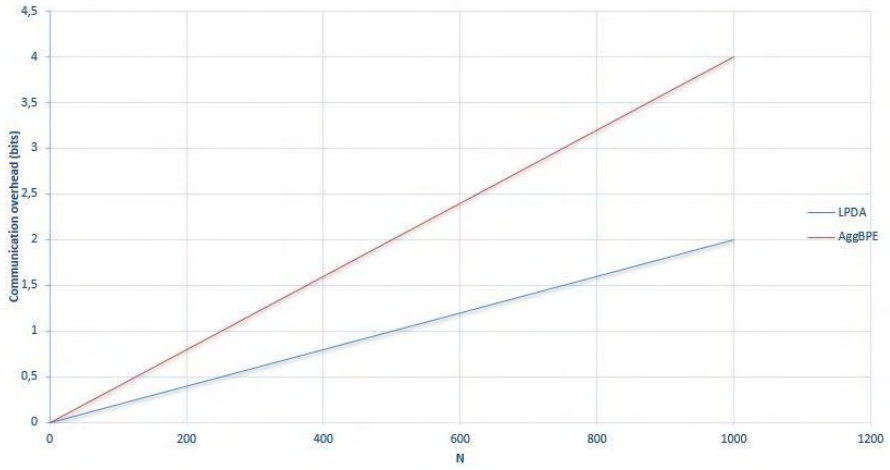
Με βάση τη συνάρτηση κατακερματισμού h, μια αλυσίδα κατακερματισμού αποτελείται από ένα σύνολο τιμών n1,n2,..,nk. Τυχαία επιλέγεται μια τιμή και υπολογίζεται το ni=h(ni+1) όπου το i εκτελεί επανάληψη από 0 έως k-1. Έπειτα πρέπει να υπολογιστεί η τιμή nj που λόγω της μονόδρομης λειτουργίας του αλγορίθμου είναι εύκολος ο υπολογισμός.

Τέλος στο μοντέλο Lightweight Privacy–preserving Data Aggregation χρησιμοποιείται ο αλγόριθμος Pailier αποτελεί έναν ασύμμετρο αλγόριθμο παραγωγής δημόσιου κλειδιού και αποτελεί ένα κρυπτοσύστημα. Βασίζεται στο γεγονός ότι δύο κρυπτογραφήσεις δεδομένων ισοδυναμεί με την άθροιση των δύο τιμών κρυπτογράφησης .Αποτελείται από τρία στάδια την παραγωγή κλειδιού, την κρυπτογράφηση και την αποκρυπτογράφηση. Πιο συγκεκριμένα στη κρυπτογράφηση, επιλέγονται δύο μεγάλοι πρώτοι αριθμοί τυχαία pκαι qέπειτα υπολογίζεται το n=p*gκαι λ= lcm(p-1,q-1), έτσι ώστε να ισχύει gcd=(p*q,(p-1)*(q-1))=1. Έπειτα διαλέγεται τυχαία ένας ακέραιος g, και ένας αριθμός που να μπορεί να τον διαιρέσει n και υπολογίζεται η συνάρτηση μ=L(g^λmodn^2)) ^(-1) όπου το L(x)=x-1/nκαι από τα αποτελέσματα μπορούμε να βρούμε το δημόσιο κλειδί= (n,g) και το ιδιωτικό=(λ,μ). Όσο αφορά τη κρυπτογράφηση ορίζουμε μια μεταβλητή m ως το αρχικό μήνυμα και μια τιμή r και δημιουργούμε το κρυπτογραφημένο κείμενο με βάση τη συνάρτηση c=(g^m * r^m)modn^2. Τέλος στην αποκρυπτογράφηση ορίζουμε μια μεταβλητή c που αποτελεί το κρυπτογραφημένο κείμενο δημιουργούμε το αρχικό κείμενο με βάση τη συνάρτηση m=L(c^λmodn^2) * μmodn. [32],[33]

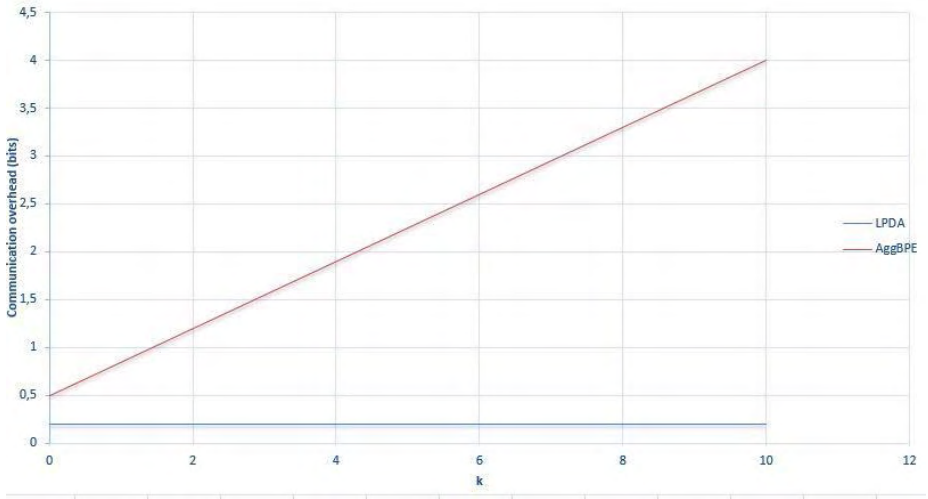
6.3.4 Διαγράμματα και πρακτική προσέγγιση

Το δοθέν μοντέλο διασφαλίζει την ιδιωτικότητα κατά την ενσωμάτωση πολλών υβριδικών IoTσυσκευών σε μια ενιαία IoT συσκευή μέσω του κέντρου ελέγχου που πραγματοποιεί υπολογισμούς για κάθε σύνολο συσκευών όπως το μέσο όρο του. Το κόστος επικοινωνίας είναι μικρότερο για το δοθέν μοντέλο, με σχέση άλλα παρόμοια μοντέλα όπως το AggBPE αφού ορίζεται για N συσκευές IoT ίσο μεN*n^2 όπου nτο μέγεθος του κρυπτογραφημένου κειμένου σε ένα αρχικό κείμενο. Ενώ στον αλγόριθμο AggBPE πραγματοποιείται κρυπτογραφία σε δύο αρχικά κείμενα και ο χρόνος ορίζεται ως N*2n^2 επομένως απαιτεί διπλάσιο κόστος επικοινωνίας. Για την

καλύτερη κατανόηση του μειωμένου κόστους επικοινωνίας στο LPDA δημιουργήθηκε το διάγραμμα στην εικόνα 50 όπου παρουσιάζεται η επιβάρυνση επικοινωνίας μεταξύ συσκευών IoT και Fog και αντίστοιχα για τις Fogσυσκευές και το κέντρο διαχείρισης στην εικόνα 51.



Εικόνα 50: Επιβάρυνση επικοινωνίας μεταξύ IoT και Fog συσκευών για LPDA και AggBPE.



Εικόνα 51: Επιβάρυνση επικοινωνίας μεταξύ Fog συσκευών και κέντρου διαχείρισης για LPDA και AggBPE.

Σχετικά με το υπολογιστικό κόστος που απαιτείται, το μοντέλο LPDA δημιουργήθηκε ώστε να είναι ελαφρύ. Προκειμένου να επιβεβαιωθεί η αποτελεσματικότητα του LPDA όσο αφορά το υπολογιστικό κόστος,

πραγματοποιήθηκε επανάληψη 1000 φορών και ο χρόνος λειτουργίας για το υπολογιστικό κόστος καταγράφηκε 0.328 ms για τις συσκευές IoT ακόμη και εάν χρησιμοποιήθηκαν τεχνικές ιδιωτικότητας, 0.470msγια συσκευές ομίχλης και 0.578 ms εάν έχουν εφαρμοστεί τεχνικές ιδιωτικότητας και για το κέντρο διαχείρισης 0.062 ms και 0.156ms εάν έχουν εφαρμοστεί τεχνικές ιδιωτικότητα. Συμπερασματικά οι μετρήσεις είναι εξίσου αποδοτικές ακόμη και ένα έχουν εφαρμοστεί περαιτέρω τεχνικές για την διαφύλαξη των ιδιωτικών δεδομένων.

Βιβλιογραφία

1. Jensen M., "On technical security issues in cloud computing", 2009.

2. Meryeme Alouane, Hanan El Bakkali, "Security, Privacy and Trust in Cloud Computing: A Comparative Study", International Conference on Cloud Technologies and Applications (Cloud Tech), 2015.

3. Zhidong Shen, Li Li, Fei Yan, Xiaoping Wu , "Cloud Computing System Based on Trusted Computing Platform", International Conference on Intelligent Computation Technology and Automation, 2010.

4. Rajat Soni, Smrutee Ambalkar, Pratosh Bansal , "Security and Privacy in Cloud Computing", Symposium on Colossal Data Analysis and Networking (CDAN), 2016.

5. Yoshita Sharma, Himanshu Gupta, Sunil Kumar Khatri, "A Security Model for the Enhancement of Data Privacy in Cloud Computing", Amity International Conference on Artificial Intelligence (AICAI), 2019.

6. Xiaojun Yu, Qiaoyan Wen (,"A View about Cloud Data Security from Data Life Cycle", International Conference on Computational Intelligence and Software Engineering, 2010.

7. Seung Jo, Heang Soo, Jongan Park(1996),“The improved data encryption standard (DES) algorithm”, Proceedings of ISSSTA'95 International Symposium on Spread Spectrum Techniques and Applications, 1996.

8. Shady Mohamed Soliman, Baher Magdy, Mohamed A. Abd El Ghany, “Efficient implementation of the AES algorithm for security application”,29th IEEE International System-on-Chip Conference (SOCC), 2016.

9. Omar Elkeelany, Adegoke Olabisi“Case study: integrates design of RC5 encryption”,IEEE SoutheastCon ,2007.

10. Yang Jun, Li Na, Ding Jun “A Design and Implematation og High-Speed 3DES Algorithm System”, Second International Conference on Future Information Technology and Management Engineering, 2009.

11. Tingyuan Nie, Teng Zhang, “A study of DES and Blowfish encryption algorithm”,IEEE Region 10 Conference, 2009.

12. Shikha Mathur, Deepika Gupta, Vishal Goar, Manoj Kuri (2017) “Analysis and design of enhanced RSA algorithm to improve to security”,3rd International Conference on Computational Intelligence & Communication Technology (CICT) ,2017.

13. Nan L, “Research on Diffie-Hellman key exchange protocol”, Information Engineering Teaching and research section, 2010.

14. Zengqiang Wu, Di Su, Gang (2014), “ElGamalalgorithm for encryption of data transmission”,International Conference on Mechatronics and Control (ICMC) ,2014 .

15. Babitha M.P, K.R Remesh Babu, “Secure Cloud Storage Using AES Encryption”, International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), 2016.

16. Naseer Amara, Huang Zhuqui, Awais Ali, “Cloud Computing Security Threats and Attacks with their Mitigation Techniques”,International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, 2017.

17. Jay Singh, Brajesh Kumar, Asha Khatri, “Improving stored data security in Cloud using Rc5 algorithm”, Nirma University International Conference on Engineering (NUiCONE), 2012.

18. Opeyememi Osanaiye, Shuo Chen, Zheng Yan, Rongxing Lu, Kim-Kwang Raymond Choo, Mqhele Dlodlo, “From Cloud to Fog Computing: A Review and a Conceptual Live VM Migration Framework”, IEEE Access, 2017.

19. Naidila Sadashiv, S. M Dilip Kumar, “Cluster, Grid and Cloud Computing: A Detailed”,The 6th International Conference on Computer Science & Education (ICCSE), 2011.

20. Rochwerger B., Caceres J., Montero JS., Breitgand D., Elmroth E., Galis A., Levy E., Llorente IM., Nagin K., Wolfsthal Y., Elmroth E., Caceres J., BenYehuda M., Emmerich W., Galan F., "The RESERVOIR Model and Architecture for Open Federated Cloud Computing", IBM Journal of Research and Development, Vol. 53, 2009

21. Gohar Rahmn, Chuah Chai Wen , “Fog Computing, Applications , Security and Challenges, Review” ,International Journal of Engineering & Technology,2018.

22. Kanghyo Lee, Donghyun Kim, Dongsoo Ham, Ubaidullah Rajputm Heekuck Oh (2015), “On Security and Privacy Issues of Fog Computing supported Internet of Things Environment”, 6th International Conference on the Network of the Future (NOF) , 2015.

23. Mithun Mukherjee, Rakesh Matam, Lei Shu, Leandros Maglaras, Mohamed Amine Ferrag, Nikumani Choudhury, Vikas Kumar, “Security and Privacy in Fog Computing: Challenges”, IEEE Access ,2017.

24. Abdullah Aljumah, Tariq, Ahamed Ahanger, “Fog Computing and Security Issues: A review”, 7th International Conference on Computers Communications and Control (ICCCC), 2018.

25. Akhilesh Vishwanath, Ramya Peruiri, Jing Selene He, “Securrity in Fog Computing through Encryption”, I.J Information Technology and Computer Science, 2016.

26. Auhidur Rahman, Hasnat Riaz, Nishu Nath, “A Fog Baed Encryption Algorithm for IoT Network”, Internation Journal of Computer Science and Information Security (IJCSIS), 2019.

27. Karthija T., A.S Radhamani, V.G Anish Gnana Vincy, L. Amutha Swaminathan, “An Overview of Security Algorithms in Cloud Computing”, International Journal of Recent Trends in Engineering & Research (IJRTER), 2017.

28. Khalid El Makkaoui, Abderrahim Beni-Hssane, Abdellah Ezzati, “Cloud-ElGamal: An Efficient Homomorphic Encryption Scheme”, International Conference on Wireless Networks and Mobile Communications (WINCOM), 2016.

29. Divya Prathana Timothy, Ajit Kumar Santra, “A hybrid cryptography algorithm for cloud computing security”,International conference on Microelectronic Devices, Circuits and Systems (ICMDCS),2012.

30. Tony Durgadas Jagyasi, Jagdish Pimple, “Security Enhancement in Cloud Computing Using Triple DES Encryption Algorithm”, International Conference on Industrial Automation And Computing (ICIAC), 2014.

31. Ivan Stojmenovic, Sheng Wen, Xinyi Huang, Hang Luan, “An overview of Fog computing and its security issues”, Wiley Online Library, 2015.

32. Yao Liu, Shuai Xue, “Accelerate the Paillier Cryptosystem in CryptDB by Chinese Remainder Theorem”, International Conference on Advanced Communications Technology (ICACT), 2018

33. Rongxing Lu, Arash Habibi Lashkari, Ali A. Ghorbani,“A lightweight Privacy-Preserving Data Aggregation Scheme for Fog Computing-Enhanced IoT”, IEE Access, 2017.

34. Hannah Williams, “The history of cloud computing: A timeline of key moments from the 1960s to now”, Computerworld, 2018. Ανάκτηση από https://www.computerworld.com/article/3412271/the-history-of-cloud-computing--a-timeline-of-key-moments-from-the-1960s-to-now.html#slide5

35. Antony T. Velte, Toby J. Velte, Robert Elsenpete, “Cloud Computing: A Practical Approach”, 2009.

36. Kris Jamsa, “Cloud Computing: SaaS, PaaS, IaaS, Virtualization, Business Models, Mobile, Security and More”, 2013.

37. Jerry Archer, Alan Boehme, Dave Cullianane, Paul Kurtz, Nils Puhlmann, Jim Reavis, “Security Guidance for Critical Areas of Focus in Cloud Computing”,Cloud Security Alliance, 2009.

38. Rehan S., “Cloud computing’s effect on enterprises, Lund University, 2011.

39. Rittinghous J, Ranson J, “Cloud Computing Implementation Management, and Security”, 2010.

40. Peter Mell, Timothy Grance, “The NIST Definition of Cloud Computing”, Computer Security Division Information Technology Laboratory National Institute of Standards and Technology Gaithersburg, 2011.

41. Mithun Mukjerjee, Lei Shu, Di Wang, “Survey ofFog Computing: Fundamental, Network Application and Research Challenges”, IEEE COMUMUNICATIONS SURVEYS & TUTORIALS, 2018.

42. Songquing Chen, Tao Zhang, Weisong Shi, “Fog Computing”, IEEE Computer Society, 2017.

43. Mohammad Aazam, Sherali Zeadally, Khaled A. Harras, “Fog Computing Architecture, Evaluation, and Future Research Directions”,IEEE Communications Magazine, 2018.

44. Danel Happ, “Cloud and Fog Computing in the Internet of Things”, Telecommunication Networks Group (TKN), 2018.

45. Seung Woo Kum, Jaewon Moon, Tae-Beom Lim, “Design of Fog Computing based IoT Application Architecture”, Internation Conference on Consumer Electronics-Berlin (ICCE –Berlin), 2017.

46. Ashkan Yousefpour , Genya Ishigaki, Riti Gour, Jason P. Jue, “On Reducing IoT Service Delay via Fog Offloading”, IEEE Internet of Things Journal, 2018.