Π Α Ν Ε Π Ι Σ Τ Η Μ Ι Ο Π Α Τ Ρ Ω Ν

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
& ΠΛΗΡΟΦΟΡΙΚΗΣ

ΕΡΓΑΣΙΑ ΕΞΑΜΗΝΟΥ
ΓΙΑ ΤΟ ΜΑΘΗΜΑ
ΔΙΚΤΥΑ ΔΗΜΟΣΙΑΣ ΧΡΗΣΗΣ
ΚΑΙ ΔΙΑΣΥΝΔΕΣΗ ΔΙΚΤΥΩΝ

«ΣΧΕΔΙΑΣΜΟΣ ΠΡΩΤΟΚΟΛΛΩΝ»

ΚΑΛΟΓΕΡΟΠΟΥΛΟΣ ΡΑΦΑΗΛ

Α.Μ 5773

ΕΠΙΒΛΕΠΩΝ ΚΑΘΗΓΗΤΗΣ:

Χ. Μπούρας, Καθηγητής

ΠΑΤΡΑ 2017 - 2018

Contents

Ακρώνυμα

- SLP - Session LAYER Protocol
- ALP - Application Layer Protocol
- SM - State Machines
- AI – Arificial Intelligence
- FTAM - (File Transfer, Access and Manager)
- VT - (Virtual Terminal)
- MOTIS - (Message Oriented Text Interchange Standard)
- CMIP - (Common Management Information Protocol)
- JTM - (Job Transfer and Manipulation) a former OSI standard
- MMS - (Manufacturing Messaging Service)
- RDA - (Remote Database Access)
- DTP - (Distributed Transaction Processing)
- ACSE - (Association Control Service Element)[5]
- ROSE - (Remote Operation Service Element)
- CCR - (Commitment Concurrency and Recovery)
- RTSE - (Reliable Transfer Service Element)
- CASE - Common Application Service Element
- SASE - Specific Application Service Element
- RPC - Remote Procedure Calls
- CRC - Cyclic Redundancy Check (CRC)
- ACK - Aknowledgement
- NACK – Negative Acknowledgement
- ARQ - Automatic repeat request
- ECC - error correcting code
- SM – State Machines
- IoT – Internet Of Things
- DDOS - Denial of Service
- IP – Internet Protocol
- IPSec – Internet Protocol Secsurity
- HTTP – HyperText Transfer Protocol
- TLS – Transport Layer Security
- SSL – Secure Sockets Layer
- NBA – Network Based Attacks

- DC – Data Corruption
- DT – Data Theft
- UCT – User-Credential Theft
- ACS – Administrative Control of Servers
- L2TP - Layer Two Tunneling Protocol
- VPN - Virtual Private Network
- PF – Packet Filtering
- ISA – Internet Security and Acceleration
- SSNAT – Server-secured network address translator
- TE – Text Encoding
- LE – Little Endian
- BE – Big Endian
- MSB – Most Significant Bit
- LSB – Least Significan Bit

Κεφάλαιο 1 : Εισαγωγή

Γιατί χρειαζόμαστε πρωτόκολλα – ορισμός πρωτοκόλλου

Από την αρχαιότητα μέχρι και σήμερα, ίσως το πιο σημαντικό επίτευγμα της ανθρωπότητας ήταν η ανάπτυξη της επικοινωνίας. Δεν υπήρξε ποτέ πολιτισμός που να μπορέσει να αναπτυχθεί και να ευημερίσει μόνος του. Προφανώς ώς επικοινωνία δεν ορίζουμε μόνο την λεκτική επικοινωνία, καθώς αυτό θα ήταν κοντόφθαλμο εκ μέρους μας. Το εμπόριο, οι συναλλαγές, η διαμοίραση πληροφοριών ακόμα και οι κοινές στρατηγικές αποτελούν εκλεπτυσμένες μορφές επικοινωνίας. Κοινή προυπόθεση για να επιτευχθούν αυτές οι διαδικασίες είναι να υπάρχει ένα πρωτόκολλο, δηλαδή ένα σύνολο κανόνων το οποίο θα θέτει τις κατευθυντήριες γραμμές προς όλους τους ενδιαφερόμενους, ώστε να είναι σε θέση να πραγματοποιήσουν σωστά την επικοινωνία τους.

Με παρόμοιο σκεπτικό έχει δομηθεί και η επικοινωνία μεταξύ υπολογιστών. Στην πληροφορική, η επικοινωνία μεταξύ υπολογιστών έχει οριστή ώς ή ανταλλαγή πληροφοριών υπό μορφή data μεταξύ υπολογιστικών συστημάτων ή σταθμών εργασίας [10] .Στην πραγματικότητα, η επικοινωνία μεταξύ ενός συνόλου υπολογιστών αποτελεί μια γενίκευση τών παραπάνω κανόνων, καθώς κάθε διαδικασία που απαιτεί την σύνδεση υπολογιστών διέπεται και από κάποιο πρωτόκολλο. Ώς προτόκολλο επικοινωνίας ορίζεται ένα σύστημα κανόνων που επιτρέπουν σε δύο ή περισσότερες οντότητες ενός συστήματος, να εκπέμπουν πληροφορίες μέσω ενός συστήματος. Κάθε προτόκολλο ορίζει τους κανόνες του, την σημασία τών μηνυμάτων , τον συγχρονισμό της επικοινωνίας και πιθανές μεθόδους ανάκτησης λαθών. Τα πρωτόκολλα μπορεί να εφαρμόζονται από το υλικό ή το λογισμικό ενός υπολογιστή , ή άπο ένα συνδυασμό και τών δύο. [1]

Ο Martin Thompson , ειδικός στο high performance computing, ορίζει ώς χαρακτηριστικό παράδειγμα ενός πρωτοκόλλου και της εφαρμογής του, ένα απλό αρχείο. Το πρωτόκολλο για να μπορέσεις να αλληπεπιδράσεις με ένα αρχείο , ορίζεται ώς μια ελεύθερη πράξη που ακολουθείται από (μηδενικά ή μή) δικαιώματα ανάγνωσης/εγγραφής στο αρχείο και καταλήγει σε μία κλειστή πράξη. Δηλαδή το πρωτόκολλο δεν ορίζει τις πράξεις αυτές, καθ’αυτές, παρά ορίζει τους κανόνες που τις διέπουν και τις καθορίζουν. [2]

Τα πρωτόκολλα υπάρχουν για να εξασφαλίζουν ότι κάποιο σύστημα θα λειτουργεί άψογα και δεν θα καταλήγει σε καταστάσεις που δεν έχουν προβλεφθεί. Ωστόσο τα πρωτόκολλα έχουν δημιουργηθεί έτσι ώστε να προσφέρουν ευελιξία και να λειτουργούν συνεργατικά με τα συστήματα. Για αυ΄το το λόγο τα πρωτόκολλα

χρησιμοποιούνται συνεχώς στην καθημαρινή ζωή, και ιδιαίτερα στους υπολογιστές. Σε γενικές γραμμές όλες οι εφαρμογές που αναπτύσσονται βρίσκονται στο $7^{ο}$ επίπεδο πρωτοκόλλων. Τα επίπεδα πρωτοκόλλων υπάρχουν για να διασφαλίζουν την συνέπεια στα συστήματα, και για αυτό το λόγο κάθε αντιστοιχία σε κάθε επίπεδο πρέπει να γίνεται σεβαστή. Χαρακτηριστικό παράδειγμα τέτοιας περίπτωσης αποτελεί το πρωτόκολλο FIX (Financial Instruments Exchange), το οποίο έχει ένα σύστημα αλληλουχίας μηνυμάτων. Οι κατασκευαστές του δεν σεβάστηκαν την ιεραρχία τών επιπέδων και σαν αποτέλεσμα συγχώνευσαν το SLP με το ALP που είχε σαν αποτέλεσμα την αυξημένη δυσκολία ανάκτησης ή υποστήριξης ισχυρών ομάδων (το πρόγραμμα εφαρμόστηκε στην Ιαπωνία και δεν θα εντρυφίσουμε στο είδος ομάδων που δημιουργήθηκαν).

Σχεδιασμός Πρωτοκόλλων

Ακόμα και στην καθημερινή μας ζωή σχεδιάζουμε πρωτόκολλα συνεχώς. Τα πρωτόκολλα ίσως τα αντιλαμβανόμαστε καλύτερα όταν τα συνδέουμε με μοντέλα συναλλαγών καθώς υπάρχουν κανόνες που διέπουν κάθε μία συναλλαγή. Τα πρωτόκολλα επικοινωνίας υπολογιστών ακολουθούν όμοιες πρακτικές και λειτουργούν αξιοποιώντας SM . Ένα πολύ γνωστό εργαλείο που χρησιμοποιείται για τον σχεδιασμό πρωτοκόλλων είναι το SBE. Επιτρέπει πολύ αποτελεσματική κωδικοποίηση και αποκωδικοποίηση μηνυμάτων τα οποία αποτελούν τα ίδια τα πρωτόκολλα. Αυτός είναι και ο λόγος που πολλά συστήματα δεν δημιουργούνται εξ αρχής αποτελεσματικά, καθώς το κόστος της κατάλληλης (απο)κωδικοποίησης μηνυμάτων είναι πολύ υψηλό.

Ο σχεδιασμός πρωτοκόλλων είναι μια απαιτητική εργασία. Προυποθέτει προεργασία και μελέτη και η διαδικασία ανάπτυξης του πρωτοκόλλου περιλαμβάνει την ανάλυση απαιτήσεων και προδιαγραφών του συστήματος στο οποίο θα εφαρμοστεί , την εκτέλεση του πρωτοκόλλου, τον έλεγχο και την εξέλιξη του. Ακόμα και για το πιο απλό πρωτόκολλο κανέναν από τα παραπάνω βήματα δεν μπορεί να παραληφθεί. Ωστόσο η διαδικασία δεν τελειώνει με την παράδοση του πρωτοκόλλου. Θα πρέπει να δοθεί χρόνος ώστε το ίδιο το πρωτόκολλο να ελεγχθεί πάνω στο ίδιο το σύστημα αρχικά σε κάποιο ελεγχόμενο περιβάλλον (alpha testing) μέσω μεθόδων blackbox και whitebox , και στην συνέχεια beta testing, όπου οι ίδιοι οι τελικοί χρήστες έχουν την δυνατότητα να χρησιμοποιήσουν και να ελέγξουν το πρωτόκολλο. Το πρωτόκολλο από την φύση του θα πρέπει να ενισχύει την αποτελεσματικότητα του συστήματος και για αυτό το λόγο θα πρέπει να είναι απλοικό, συνεπές και αποδοτικό.

Κεφάλαιο 2 : Δομή Πρωτοκόλλου – Σχεδίασης τους

Οικογένειες Πρωτοκόλλων

Όπως δεν είναι δυνατόν ένα πρωτόκολλο να έχει γενική χρήση και να εφαρμόζεται σε όλα τα συστήματα, έτσι με την ανάπτυξη των τεχνολογιών και τών διαδικασιών που διέπουν τα πρωτόκολλα γεννιέται η ανάγκη δημιουργίας επιμέρους πρωτοκόλλων. Έτσι δημιουργείται η έννοια της Οικογένειας πρωτοκόλλων (Protocol Family). Πριν τον ορισμό της οικογένειας πρωτοκόλλων, θα πρέπει να οριστούν οι Protocol Suites. Ως Protocol suite (PS) ορίζεται το θεωρητικό μοντέλο και σύνολο πρωτοκόλλων επικοινωνίας τα οποια χρησιμοποιούνται σε διάφορα δίκτυα όπως και το Internet. Τα πιο διαδεδομένα πρωτόκολλα του κλάδου είναι τα TCP και IP . Συχνά αναφέρεται και ώς Department of Defence Model (DoD M) λόγω του ότι στην ανάπτυξη τών μεθόδων δικτύων, χορηγών ήταν το Department Of Defense τών ΗΠΑ μέσω του DARPA. Η χρήση του PS είναι να παρέχει END TO END data communication , ορίζοντας επακριβώς πώς τα δεδομένα θα πρέπει να ομαδοποιούνται , κατευθύνονται , μεταδίδονται,να επιλέγουν το μονοπάτι που θα ακολουθήσουν προς τον προορισμό τους (routed) και πώς να λαμβάνονται. Αυτή η λειτουργία δομείται σε 4 επίπεδα τα οποία ταξινομούν όλα τα πρωτόκολλα σύμφωνα με την εμβέλεια του δικτύου το οποίο χρησιμοποιείται. [3]

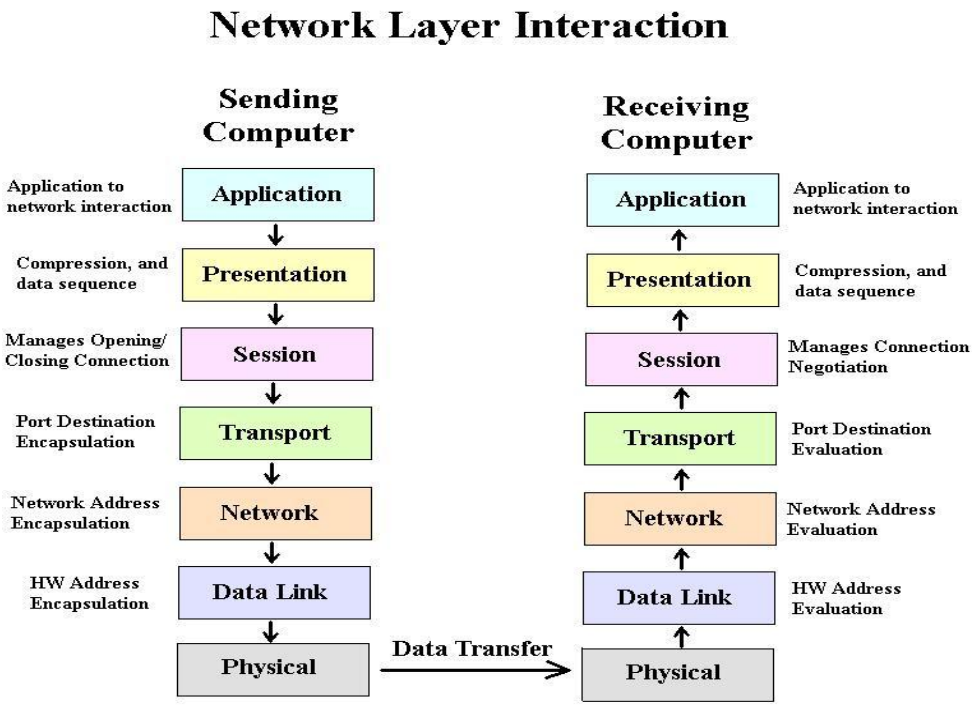
Σύμφωνα με τον ορισμό της η οικογένεια πρωτοκόλλων είναι μία ομάδα λογικών ιδιοτήτων που υπάρχουν σε μια διαμόρφωση διεπαφής . Οι οικογένειες πρωτοκόλων περιέχουν όλα τα πρωτόκολλα τα οποία ορίζουν μια protocol suite. Έτσι για να χρησιμοποιηθεί ένα πρωτόκολλο σε μια συγκεκριμμένη suite , θα πρέπει να διαμορφωθεί ολόκληρη η οικογένεια ώς μια λογική ιδιότητα της διεπαφής. [4] Με πιο απλά λόγια μια οικογένεια πρωτοκόλλων αποτελείται από επιμέρους πρωτόκολλα τα οποία εφαρμόζονται μεμονωμένα και συγκεκριμμένα αντί του γενικού πρωτοκόλου το οποίο σχηματίζουν. Αυτή είναι μία τακτική που χρησιμοποιείται ευρέως στον κλάδο της πληροφορικής και όχι μόνο, καθώς είναι πολλές οι περιπτώσεις όπου ένα πρόβλημα (η λέξη «πρόβλημα» χρησιμοποιείται μεταφορικά) διασπάται σε επιμέρους τμήματα τα οποία είναι ευκολότερο να επιλυθούν μεμονωμένα. Χαρακτηριστικό παράδειγμα αποτελεί η τεχνητή νοημοσύνη ή αλλιώς AI .

Η λειτουργία της οικογένειας πρωτοκόλλων είναι απαράμιλλης σημασίας. Συγκεκριμμένα κάθε ένα από τα επιμέρους πρωτόκολλα θα πρέπει να έχει μία ξεκάθαρα ορισμένη αποστολή και να μην επεμβαίνει στην αποστολή οποιουδήποτε διαφορετικού πρωτοκόλλου. Αυτό όμως δεν συμαίνει πώς δεν θα πρέπει να υπάρχει συνεργασία μεταξύ τών διάφορων πρωτοκόλλων, απλά θα πρέπει κάθε πρωτόκολλο να

έχει καλά ορισμένα όρια. Όλα όμως θα πρέπει να έχουν κοινές δομές δεδομένων και παραγόμενες πληροφορίες. Έτσι όπως είναι καίριας σημασίας κάθε επιμέρους πρωτόκολλο να έχει συγκεκριμμένη λειτουργία έτσι απαιτείται και για κάθε διαργασία να επιλέγεται και η σωστή οικογένεια πρωτοκόλλων.

Τα επίπεδα τών πρωτοκόλλων

Σύμφωνα με το μοντέλο OSI , κάθε δίκτυο πρωτοκόλλων διαχωρίζεται σε επίπεδα. Πιο συγκεκριμμένα το OSI model αποτελεί μία οικογένεια πρωτοκόλλων. Η στοίβα πρωτοκόλλων παρουσιάζεται στην παρακάτω εικόνα:



Εικόνα 2.1 : Protocol Network Layers

Τι αποτελεί όμως ή λίστα πρωτοκόλλων?

Η λίστα πρωτοκόλλων αποτελεί την πλήρη εφαρμογή ενός ps ή μίας οικογένειας πρωτοκόλλων. Στην πράξη μπορούμε να πούμε ότι οι ορισμοί τών πρωτοκόλλων γίνονται στην ps ενώ ή εφαρμογή τους μέσω λογισμικού γίνεται από την στοίβα πρωτοκόλλων.

Όπως έχει ήδη αναφερθεί κάθε πρωτόκολλο σε μία οικογένεια πρωτοκόλλων σχεδιάζεται με βλέψεις για μία συγκεκριμμένη λειτουργία. Συνήθως κάθε επιμέρους πρωτόκολλο επικοινωνεί με δύο άλλα δημιουργώντας έτσι της εντύπωση μίας στοίβας (δηλαδή στην απαικόνιση τους επικοινωνεί με το πάνω επίπεδο και το κάτω επίπεδο του) . Στην πράξη κάθε στοίβα πρωτοκόλλου χωρίζεται σε 3 βασικούς τομείς: media, transport και applications. Κατά κύριο κανόνα το λειτουργικό σύστημα έχει πρόσβαση μέσω λογισμικού σε δύο σημεία στην στοίβα. Το πρώοτ ορίζεται μεταξύ τών media και transport layers και το άλλο μεταξύ τών transport layers και applications. Το πρώτο ορίζει το πώς το λογισμικό του transport protocol χρησιμοποιεί τα διάφορα μέσα (media) και το δεύτερο ορίζει το πώς το λογισμικό μεταφοράς TCP/IP επικοινωνεί με το υλικό Ethernet.

Παρακάτω θα αναλύσουμε μεθοδικά κάθε ένα από τα 7 ορισμένα επίπεδα [5] [6] :

7) Επίπεδο Μεταφοράς

Η ανάγκη ύπαρξης αυτού του επιπέδου δημιουργείται από τις ίδιες τις εφαρμογές στις οποίες αναφέρεται. Λόγω της ύπαρξης πολλών επαναστατικών εφαρμογών έπρεπε να δημιουργηθεί και το κατάλληλο επίπεδο για να τις υποστηρίξει, καθώς πολλές απο άυτές δομούν και το internet. Το επίπεδο αυτό είναι το τελευταίο και περιέχει όλες τις ενέργειες για το πως μια εφαρμογή χρησιμοποιεί ένα δίκτυο. Η μελέτη αυτού του επιπέδου μας δίνει την δυνατότητα να να αναπτύξουμε προγράμματα τα οποίο είναι δικτυακά. Με άλλα λόγια βασίζονται στο δίκτυο και έχουν την δυνατότητα εκτέλεσης σε πολλούς και διαφορετικούς υπολογιστές. Επικοινωνούν πάνω από το δίκτυο π.χ. το λογισμικό του εξυπηρετητή επικοινωνεί με το λογισμικό του browser. Σε αυτό το επίπεδο δεν υπάρχει η ανάγκη να γραφτεί λογισμικό για τις συσκευές του πυρήνα του δικτύου, καθώς αυτές δεν τρέχουν εφαρμογές χρήστη.

Οι αρχιτεκτονικές τών εφαρμογών μπορεί να είναι μεταξύ ενός πελάτη και ενός εξυπηρετητή (client – server) και μεταξύ ομότιμων (peer to peer – P2P). Στην πρώτη περίπτωση ο εξυπηρετητής απαιτείται να είναι ένας συνεχός ενεργός υπολογιστής με μόνιμη – σταθερή διέθυνση IP και στην περίπτωση κλιμάκωσης απαιτούνται data centers. Αντίθετα ο πελάτης επικοινωνεί κατά βούληση με τον εξυπηρετητή, μπορεί να έχει μια δυναμική διαύθυνση IP και δεν επικοινωνεί με άλλους πελάτες. Στην δεύτερη περίπτωση δεν υπάχρει κάποιος συνεχός ενργός εξυπηρετητής καθώς τυχαία τερματικά επικοινωνούν μεταξύ τους. Σε αυτήν την περίπτωση ομότιμοι ζητούν υπηρεσίες από άλλους ομότιμους.

Αυτό προσφέρει και μία επεκτασιμότητα στο σύστημα καθώς υπάρχει η δυνατότητα να παρέχονται και να ζητούνται συνεχώς νέες υπηρεσίες. Το αρνητικό είναι ότι λόγω της μη σταθερής επικοινωνίας υπάρχει διακοπτώμενη επικοινωνία μεταξύ ομότιμων και αλλαγή τών διευθύνσεων IP.

Οι διεργασίες που επικοινωνούν είναι προγράμματα που τρέχουν σε έναν υπολογιστή. Σε διαφορετικούς υπολογιστές οι διεργασίες επικοινωνούν ανταλλάσσοντας μηνύματα ενώ στον ίδιο υπολογιστή μέσω διεργασιακής επικοινωνίας που ορίζεται από το λειτουργικό. Ώς διεργασία πελάτης εννοούμε την διεργασία που εκκινεί την επικοινωνία ενώ ώς διεργασία εξυπηρετητή ορίζουμε την διεργασία ή οποία αναμένει να επικοινωνήσουν μαζί της. Ώς sockets ορίζουμε την διεπαφή ανάμεσα στην εφαρμογή και το δίκτυο μέσω από την οποία η διεργασία στέλνει και λαμβάνει μηνύματα. Τα μηνύματα λαμβάνονται μέσω διευθυνσιοδότησης η οποία διαφέρει μεταξύ διαφορετικών μοντέλων.

Τέλος το πρωτόκολλο εφαρμογής ορίζει τα είδη τών μηνυμάτων που ανταλλάσσονται , την σύνταξη του μηνύματος, την σημασιολογία τών πεδίων,τους κανόνες για το πότε και πώς οι διεργασίες στέλνουν και απαντούν σε μηνύματα και τα πρωτόκολλα.

6) Επίπεδο παρουσίασης

Λειτουργεί ώς ο μεταφραστής τών δεδομένω για το δίκτυο και πολλές φορές καλείται ώς syntax layer. Είναι υπεύθυνο γι ατην παράδοση και ταξινόμηση τής πληροφορίας του επιπέδου εφαρμογής για περαιτέρω επεξεργασία ή παρουσίαση. Η λειτουργία του αντιμετωπίζει τις συντακτικές διαφορές στην απεικόνιση δεδομένων μεταξύ τών συστημάτων τών τελικών χρηστών. Τέτοιο παράδειγμα αποτελεί η διευθέτηση της απεικόνισης και της εναλλαγής μεταξύ ενός EBCDIC – coded αρχείου κειμένου σε ένα ASCII κωδικοποιημένο αρχείο.

Το επίπεδο παρουσίασης είναι το τελεθταίο επίπεσο στο οποίο οι προγραμματιστές εφαρμογών ασχολούνται με δομές δεδομένων και με την παρουσίαση, αντί να στέλνουν απλώς δεδομένα στην μορφή datagrams ή ώς πακέτα μεταξύ ξενιστών (hosts). Πιο συγκεκριμμένα σε αυτό το επίπεδο γίνεται ή απεικόνιση συμβολοσειρών με κυρίαρχη την μέθοδο Pascal. Η βασική ιδέα είναι ότι το επίπεδο εφαρμογής θα υποδεικνύει τα δεδομένα προς μετακίνηση και το επίπεδο παρουσίασης θα αναλαμβάνει τα υπόλοιπα. Επίσης σε αυτό το επίπεδο γίνονται και πολλές διαδικασίες κρυπτογράφησης - αποκρυπτογράφησης, ωστόσο αυτές δεν περιορίζονται εδώ αλλά μπορούν να πραγματοποιηθούν και σε άλλα επίπεδα Το επίπεδο παρουσίασης συχνά αποτελείται από δύο υποεπίπεδα:

CASE

Παρέχει υπηρεσίες για το επίπεδο εφαρμογής και αιτήματα υπηρεσιών από το επίπεδο συνόδου. Προσφέρει υποστήριξη σε πολλές υπηρεσίες όπως:

- ACSE
- ROSE
- CCR
- RTSE

SASE

Προσφέρει συγκεκριμμένα πρωτόκολλα όπως:

- FTAM
- VT
- MOTIS
- CMIP
- JTM
- MMS
- RDA
- DTP

5) Session Layer

Ελέγχει τις συνδέσεις μεταξύ υπολογιστών. Εγκαθιδρύει, διαχειρίζεται και τερματίζει τις συνδέσεις μεταξύ τοπικών και απομακρυσμένων εφαρμογών. Παρέχει υπηρεσίες για full-duplex, half-duplex, simplex διεργασίες και δημιουργεί σημεία ελέγχου , τερματισμού κ.α. Συναντάται συνήθως σε περιβάλλοντα εφαρμογών τα οποία χρησιμοποιούν απομακρυσμένες κλήσεις πρός διεργασίες.

Πιο συγκεκριμμένα το επίπεδο συνόδου παρέχει τον μηχανισμό για να ανοίγει, να κλείνει συνόδους αλλά και τον μηχανισμό διαχείρισης μιας συνόδου μεταξύ τελικών χρηστών. Οι σύνοδοι επικοινωνιών εποτελούνται από απαιτήσεις και απαντήσεις οι οποίες πραγματοποιούνται μεταξύ εφαρμογών. Οι υπηρεσίες που προσφέρει το επίπεδο συνόδου αξιοποιούνται κυρίως από περιβάλλοντα εφαρμογών τα οποία χρησιμοποιούν απομακρυσμένες κλήσεις διεργασιών (RPC) .

Το επίπεδο συνόδου αναλαμβάνει την δημιουργία σημείων ελέγχου και ανάκτησης. Επιτρέπει στις πληροφορίες από διαφορετικές εισροές πιθανών και από διαφορετικούς πόρους να συγχρονίζονται και να συνδυάζονται αποτελεσματικά.Το

επίπεδο συνόδου απαντάει σε ετοίματα που γίνονται από το επίπεδο παρουσίασης και δημιουργεί αιτήματα προς το επίπεδο μεταφοράς.

Διάφορα πρωτόκολλα που χρησιμοποιούνται από το επίπεδο είναι:

- ADSP
- ASP
- H.245
- ISO-SP
- iSNS
- L2F
- L2TP
- NetBIOS
- PAP

4) Επίπεδο Μεταφοράς

Το επίπεδο μεταφοράς είναι υπεύθυνο για την πολύπλεξη και αποπολύπλεξη δεδομένων, και για την αξιόπιστη μεταφορά τους. Σε αυτό το επίπεδο πραγματοποιείται ο έλεγχος ροής τών δεδομένων αλλά και ο έλεγχος συμφόρησης. Οι υπηρεσίες και τα πρωτόκολλα του επιπέδου αυτού παρέχουν λογική επικοινωνία μεταξύ διεργασιών εφαρμογών που τρέχουν σε διαφορετικά τερματικά συστήματα. Τα πρωτόκολλα μεταφοράς τρέχουν σε τερματικά συστήματα και αποτελούνται από την πλευρά αποστολής που σπάει τα μηνύματα της εφαρμογής σε τμήματα και τα περνάει στο επίπεδο δικτύου. Στην πλευρά λήψης γίνεται ανασύνθεση τών τμημάτων σε μηνύματα τα οποία περνάνε στο επίπεδο εφαρμογής.

Ενώ στο επίπεδο δικτύου γίνεται η λογική επικοινωνία μεταξύ υπολογιστών , στο επίπεδο μεταφοράς γίνεται λογική επικοινωνία μεταξύ διεργασιών – Το επίπεδο μεταφοράς συμπληρώνει τις υπηρεσίες του επιπέδου δικτύου. Έπειτα θα αναλύσουμε δύο όρους:

Πολύπλεξη: Πραγματοποιείται στον υπολογιστή λήψης και κατά την διάρκεια της χρησιμοποιούνται οι πληροφορίες κεφαλίδας για την παράδοση τών τμημάτων που λαμβάνονται στο σωστό socket.

Αποπολύπλεξη: Πραγματοποιείται στον υπολογιστή αποστολής και είναι υπεύθυνη για την διαχείριση δεδομένων από πολλαπλά sockets και την προσθήκη κεφαλίδας μεταφοράς.

Το συγκεκριμμένο επίπεδο χρησιμοποιεί κυρίως δύο πρωτόκολλα.λ Και στα δύο πρωτόκολλα δεν υπάρχουν εγγυήσεις ώς προς την καθυστέρηση τών μηνυμάτων αλλά και εγγυήσεις ώς προς το εύρος ζώνης. Τα πρωτόκολλα που χρησιμοποιούνται από το επίπεδο μεταδοράς είναι τα εξής:

- **TCP**
  Το συγκεκριμμένο πρωτόκολλο προσφέρει αξιόπιστη και σε σωσστή σειρά μεταφορά μέσω του ελέγχου συμφόρησης , του ελέγχου ροής και τηε εγκαθίδρυσης σύνδεσης.
- **UDP**
  Το συγκεκριμμένο πρωτόκολλο προσφέρει μη αξιόπιστη και εκτός σειράς παράδοση μηνυμάτων.

Τέλος θα περιγράψουμε το πώς πετυχαίνεται η αξιόπιστη μεταφορά δεδομένων στο συγκεκριμμένο επίπεδο. Για να γίνει αυτό προαπαιτούμε ότι το κανάλι θα είναι πλήρως αξιόπιστο χωρίς λάθη bit και χωρίς απώλειες πακέτων. Έπειτα ο αποστολέας και ο δέκτης είναι εφοδιασμένοι με FSMs και χρησιμοποιείται checksum για να αναγνωρίζουν ανα το πακέτο έχει μεταφερθεί σωστά. Έπειτα ανταλλάσσουν μεταξύ τους μηνύματα αναγνώρισης σωστής ή μη σωστής μεταφοράς πακέτων , ενώ είναι εφοδιασμένοι και με μηχανισμούς ανάκτησης λαθών (κυρίως μέσω αντιστροφή bit).

4) Επίπεδο δικτύου

Στο επίπεδο δικτύου γίνεται μεταφορά τμήματος από τον υπολογιστή του αποστολέα στον υπολογιστή του δέκτη. Στην πλευρά της αποστολής ενθυλακώνονται τα τμήματα σε datagrams ενώ στην πλευρά του δέκτη παραδίδονται τα τμήματα στο επίπεδο μεταφοράς. Πρωτόκολλα αυτού του επιπέδου υπάρχουν σε κάθε υπολογιστή ανά τον κόσμο και ο δρομολογητής εξετάζει τα πεδία της κεφαλίδας όλως τών IP datagrams που περνούν από αυτόν.

Οι βασικές λειτουργίες που επιτελούνται σε αυτό το επίπεδο είναι:

- Προώθηση
  Κατά την προώθηση γίνεται μετακίνηση πακέτων από την είσοδο του δρομολογητή στην κατάλληλη έξοδο του δρομολογητή. Ώς δρομολόγηση ορίζεται η διαδικασία σχεδιασμού του ταξιδιού από την προέλευση του μηνύματος μέχρι τον προορισμό του.

- Δρομολόγηση
  Κατά την δρομολόγηση γίνεται ο καθορισμός της διαδρομής που ακολουθούν τα πακέτα από την προέλευση στον προορισμό τους.
- Δημιουργία Σύνδεσης
  Είναι μια πολύ σημαντική λειτουργία σε ορισμένες αρχιτεκτονικές όπως η
    1. ATM
    2. Frame Play
    3. X.25

Πριν την ροή τών datagrams οι δύο τερματικοί υπολογιστές και δρομολογητές που μεσολαβούν θα πρέπει να εγκαθιδρύσουν μία εικονική σύνδεση στην οποία θα συμμετέχουν και οι δρομολογητές. Το επίπεδο δικτύου έχει την δυνατότητα να επιλέξει ανάμεσα σε μία από δύο διαθέσιμες επιλογές. Το δίκτυο datagram παρέχει υπηρεσία επιπέδου δικτύου χωρίς σύνδεση ενώ το επίπεδο εικονικού κυκλώματος παρέχειο υπηρεσία επιπέδου δικτύου με σύνδεση. Αυτή η μορφή δικτύων είναι ανάλογη με τις υπηρεσίες του επιπέδου μεταφοράς αλλά σε αυτήν την περίπτωση η υπηρεσία ορίζεται ώς υπολογιστής πρός υπολογιστή (host to host) , η υλοποίηση γίνεται στον πυρήνα του δικτύου και το δίκτυο το ίδιο επιλέγει μία από τις δύο επιλογές.

Τέλος θα αναφερθούμε στις δύο βασικές λειτουργίες που έχει ένας δρομολογητής. Αναλαμβάνει την εκτέλεση αλγορίθμων/πρωτοκόλλων δρομολόγησης και την προώθηση datagrams από εισερχόμενη σε εξερχόμενη ζεύξη. Δεδομένου του προορισμού του datagram γίνεται αναζήτηση της θύας εξόδου με χρήση του πίνακα προώθησης στην μνήμη της θύρας εισόδου.λ Ο σκοπός είνα ινα γίνει ολοκλήρωση της επεξεργασίας της θύρας εισόδου με ταχύτητα γραμμής και έτσι δεν υπάρχει αναμονή αν τα datagrams φτάνουν ταχύτερα από το ρυθμό προώθησης στο δόμημα μεταγωγής.

2) Επίπεδο Ζεύξης

Το επίπεδο ζεύξης πραγματοποιείται σε κάθε υπολογιστή , στον προσωπικό του προσαρμογέα δικτύου δηλαδή την κάρτα δικτύου. Είναι το μόνο επίπεδο που αποτελέι συνδυασμό hardware,software και firmware. Σε αυτό το επίπεδο πραγματοποιείται η ανίχνευση και η διόρθωση σφαλμάτων, μέσω κοινής χρήσης ενός καναλιού ευρύας εκπομπής. Εδώ γίνεται και η διευθυνσιοδότηση του επιπέδου ζεύξης .Σε αυτό το επίπεδο έχουμε την δημιουργία δικτύων στα οποία οι υπολογιστές και οι δρομολογητές αναγνωρίζονται ώς κόμβοι (nodes) και τα κανάλια επικοινωνίας που ενώνουν γειτονικούς κόμβους κατά μήκος της διαδρομής επικοινωνίας ορίζονται ώς ζεύξεις (links). Οι ζεύξεις μπορεί να είναι ενσύρματες, ασύρματες ή LANs.

Στο πλαίσιο του επιπέδου ζεύξης ενθυλακώνεται το datagram το οποίο μεταδφέρεται από διαφορετικά πρωτόκολλα επιπέδου ζεύξης σε διαφορετικές ζεύξεις (πιθανώς). Εδώ πρέπει να αναφερθεί ότι κάθε πρωτόκολλο ζεύξης παρέχει διαφορετικές υπηρεσίες. Αυτές οι υπηρεσίες είναι οι εξής:

- Πλαισίωση – Πρόσβαση στην ζεύξη
- Αξιόπιστη παράδοση μεταξύ γειτονικών κόμβων
- Ελεγχος ροής
- Ανίχνευση σφαλμάτων
- Διόρθωση σφαλμάτων
- Ημι-αμφίδρομη και αμφίδρομη μετάδοση

1) Φυσικό επίπεδο

Το φυσικό επίπεδο αναλαμβάνει την μετάδοση της πληροφορίας μέσω φυσικών μέσων. Μεταφέρει την πληροφορία που δέχεται από το αμέσως ανώτερο επίπεδο και την μεταφέρει στο επόμενο του χρησιμοποιώντας το επιλεγμένο μέσο μετάδοσης. Το φυσικό μέσο μετάδοσης της πληροφορίας μπορεί να αλλάξει από επίπεδο σε επίπεδο οπότε μπορεί να αλλάξει και η τεχνολογία του χρησιμοποιούμενου μέσου, επηρεάζοντας έτσι την ταχύτητα της μετάδοσης.

Το φυσικό επίπεδο αποτελείται από όλες τις τεχνολογίες μετάδοσης σε ένα δίκτυο. Είναι το πιο βασικό επίπεδο καθώς υποστηρίζει τα ανώτερα 6 επίπεδα και λόγω τών διαφορετικών τεχνολογιών που χρησιμοποιούνται μπορεί να θεωρηθεί και το πιο περίτεχνο. Μεταφέρει μεμονωμένα bits σε αντίθεση με τα προηγούμενα τα οποία μετέφεραν πακέτα. Μέσα στα πλαίσιο της αρχιτεκτονικής OSI το φυσικό επίπεδο μεταφράζει αιτήματα επικοινωνίας από το data link επίπεδο σε συγκεκριμμένες εντολές προς το υλικό του υπολογιστή.

Μερικές από τις υπηρεσίες που παρέχει είναι:

- Bit-by-bit παράδοση
- Forward error correction/channel κωδικοποίηση όπως ο ECC
- Bit interleaving για βελτίωση της απόδοσης της διόρθωσης σφαλμάτων
- Auto-negotiation
- Transmission mode control
- Συγχρονισμός bit για σύγχρονες παράλληλες επικοινωνίες
- Κωδικοποίηση γραμμών, που επιτρέπει στα δεδομένα που προέρχονται από το υλικό να βελτιστοποιούνται για ψηφιακή επικοινωνία.

Κεφάλαιο 3 : Αξιοπιστία Πρωτοκόλλου

Τι ορίζεται ώς αξιοπιστία πρωτοκόλου?

Στα δίκτυα υπολογιστών ένα πρωτόκολο θεωρείται αξιόπιστο όταν παρέχει διαπιστευτήρια της παράδοσης τών δεδομένων στους παραλήπτες, σε αντίθεση με ένα μη-αξιόπιστο πρωτόκολλο το οποιο δεν παρέχει καμία πληροφορία. [7] Χαρακτηριστικό παράδειγμα αξιόπιστου πρωτοκόλου είναι το πρωτόκολο tcp το οποίο θεωρείται αξιόπιστο λόγω του ότι το ίδιο το πρωτόκολλο πραγματοποιεί έλεγχο σωστής μετάδοσης δεδομένων στον παραλήπτη (είτε αυτή προέρχεται είτε όχι από χάσιμο μεταδιδόμενων πακέτων). Το tcp επίσης υποστηρίζει την επαναμετάδοση χαμένων πακέτων, παρέχοντας έτσι ασφάλεια ότι εν τέλει , όλα τα πακέτα θα έχουν μεταφερθεί στο προορισμό τους. [8]Η αξιοπιστία χρησιμοποιείται ώς συνόνυμο της διασφάλισης , ο οποίος είναι και ο επίσημος όρος που χρησιμοποιείται από το ATM Forum στο πλαίσιο της ATM Service-Specific Coordination Function.

Γιατί όμως είναι απαραίτητη η αξιοπιστία σε ένα πρωτόκολο?

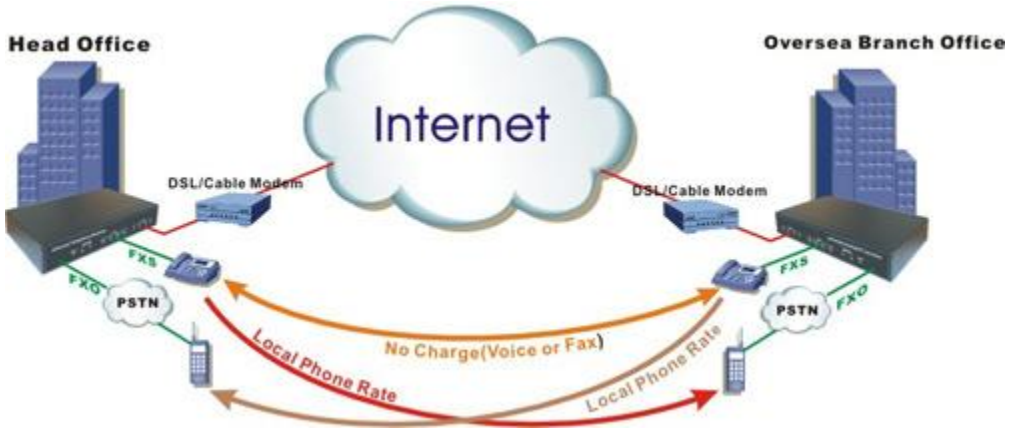
Όπως έχει ήδη αναφερθεί η επικοινωνία τών υπολογιστών γίνεται σε δίκτυα. Ο βασικός λόγος ύπαρξης τών πρωτοκόλλων είναι να συγχρονίζουν τις καταστάσεις στις οποίες βρίσκονται δύο κόμβοι . Η κατάσταση αυτή δεν είναι ένας μονομερής προσδιορισμός, και μπορεί να είναι κάποια δεδομένα, μία σχέση επικοινωνίας ακόμα και δεδομένα μίας βάσης δεδομένων. Έτσι για να είναι αξιόπιστο το πρωτόκολλο θα πρέπει ο συγχρονισμός αυτός να πραγματοποιείται αποτελεσματικά και για λόγους υπολογιστικών πόρων και με τον ελάχιστο αριθμό ανταλλαγής μηνυμάτων.

Ωστόσο κατά τον σχεδιασμό πρωτοκόλλων συχνά καλούμαστε να επιλέξουμε ανάμεσα στην αξιοπιστία και την απόδοση του πρωτοκόλλου που θα δημιουργήσουμε. Εφόσων δεν υπάρχουν σαφείς κανόνες που να καθορίζουν ποιο από τα δύο πρέπει να επιλέξουμε προσπαθούμε να πετύχουμε το “sweet spot” που θα μας εξασφαλίσει τον βέλτιστο συνδυασμό αυτών τών δύο και φυσικά εναλλάσεται ανάλογα με την περίσταση. Η επιβάρυνση λοιπόν είναι κάτι που δεν μπορούμε να αποφύγουμε. Μία διεργασία ή οποία επιβαρύνει το πρωτόκολλο μας είναι η σωστή κωδικοποίηση ή αποκωδικοποίηση της πληροφορίας για τν έλεγχο λαθών και την άυξηση της αξιοπιστίας. Αντίστοιχα οι διάφορες μεθόδοι εύρεσης και διόρθωσης λαθών επιβάλλουν μεγαλύτερο φόρτο επεξεργασίας. Οι διάφοροι μηχανισμοί που θα χρησιμοποιηθούν επιβαρύνουν το πρωτόκολλο διαφορετικά ο κάθε ένας οπότε η απόφαση που θα πάρουμε παίζει σημαντικό ρόλο στην αξιοπιστία και την απόδοση του πρωτοκόλλου μας.

Ποιοι είναι όμως οι μηχανισμοί ελέγχου αξιοπιστίας?

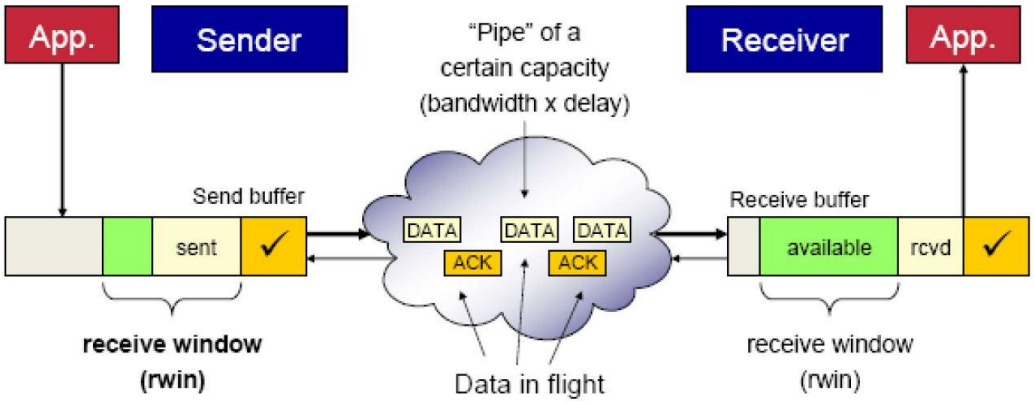
Οι μηχανισμοί ελέγχου αξιοπιστίας είναι οι μηχανισμοί που έχουν επιλεγεί για να ελέγξουν άν το πρωτόκολλα μας πληροί τις προυποθέσεις που απαιούνται και εξασφαλίζουν την συμμόρφωση του με αυτές. Δεν είναι γενικοί που σημαίνει πώς ή επιλογή τους ποικίλει ανάλογα με την περίσταση και την εφαρμογή στην οποία θα χρησιμοποιηθούν, το περιβάλλον στο οποίο θα λειτουργήσουν (είδη λαθών, συχνότητα σφαλμάτων κ.α.), το μοντέλο το οποίο χρησιμοποιούν (π.χ. αν βασίζεται σε κάποιο πρωτόκολλο) και την σταθερότητα του συχστήματος (δηλαδή ο αριθμός τών λαθών είναι γνωστός ή ποικίλει από στιγμή σε στιγμή).

Όπως έχει ήδη αναφερθεί ένα από τα πιο αξιόπιστα πρωτόκολλα που χρησιμοποιούνται είναι το πρωτόκολλο tcp. Για αυτό το λόγο θα βασιστούμε σε αυτό προκειμένου να εξάγουμε συμπεράσματα τα οποία να αποδεικνύουν τα λεγόμενα μας. Το πρωτόκολλο tcp χωρίζεται σε 3 φάσεις. Πρώτα γίνεται η σωστή εγκαθίδρυση επικοινωνίας μέσω μιας πολυεπίπεδης διαδικασίας handshake. Ακολουθεί η φάση μετάδοσης δεδομένων, και μόλις ολοκληρωθεί τερματίζεται η σύνδεση και απελευθερώνονται οι αξιοποιούμενοι πόροι.



Εικόνα 3.1 : Παράδειγμα δικτύου επικοινωνίας

Παρατηρώντας την παραπάνω εικόνα κατανοούμε ότι το πρωτόκολλο δεν εφαρμόζεται μόνο σε ένα σημείο κατά την διάρκεια της επικοινωνίας. Ανά πάσα στιγμή πολλές συσκευές συνδέονται με πολλές συσκευές οπότε σε κάθε μία από αυτές τις συνδέσεις χρησιμοποιείται και το κα΄ταλληλο πρωτόκολλο.



Εικόνα 3.2 : Ανάλυση πρωτοκόλλου tcp

Ποιοι είναι οι μηχανισμοί του tcp protocol? [11]

Slow Start

Αργή εκκίνηση, μια απαίτηση για υλοποιήσεις λογισμικού TCP είναι ένας μηχανισμός που χρησιμοποιείται από τον αποστολέα για τον έλεγχο του ρυθμού μετάδοσης, αλλιώς γνωστός ως έλεγχος ροής με βάση τον αποστολέα. Αυτό επιτυγχάνεται μέσω του ρυθμού επιστροφής των αναγνωρίσεων από τον δέκτη. Με άλλα λόγια, ο ρυθμός των αναγνωρίσεων που επιστρέφει ο δέκτης καθορίζει το ρυθμό με τον οποίο ο αποστολέας μπορεί να μεταδώσει δεδομένα.Όταν ξεκινά αρχικά μια σύνδεση TCP, ο αλγόριθμος Slow Start αρχικοποιεί ένα παράθυρο συμφόρησης σε ένα τμήμα, το οποίο είναι το μέγιστο μέγεθος τμήματος (MSS) που προετοιμάστηκε από τον δέκτη κατά τη φάση εγκατάστασης σύνδεσης. Όταν οι παραλήπτες επιστρέφουν αναγνωρίσεις, το παράθυρο συμφόρησης αυξάνεται κατά ένα τμήμα για κάθε επιβεβαίωση που επιστρέφεται. Έτσι, ο αποστολέας μπορεί να μεταδώσει το ελάχιστο παράθυρο συμφόρησης και το διαφημιζόμενο παράθυρο του δέκτη, το οποίο απλά ονομάζεται παράθυρο μετάδοσης.

Congestion Avoidance

Κατά τη διάρκεια της αρχικής φάσης μεταφοράς δεδομένων μιας σύνδεσης TCP χρησιμοποιείται ο αλγόριθμος Slow Start. Εντούτοις, κατά τη διάρκεια αργής εκκίνησης ενδέχεται να υπάρχει σημείο κατά το οποίο το δίκτυο αναγκάζεται να αποβάλει ένα ή περισσότερα πακέτα λόγω υπερφόρτωσης ή συμφόρησης. Εάν συμβεί αυτό, η Αποφυγή συμφόρησης χρησιμοποιείται για να επιβραδύνει την ταχύτητα μετάδοσης. Ωστόσο, η αργή εκκίνηση χρησιμοποιείται σε συνδυασμό με τη Αποφυγή συμφόρησης ως μέσο για να γίνει η μεταφορά δεδομένων ξανά, ώστε να μην επιβραδυνθεί και να

παραμείνει αργή. Στον αλγόριθμο αποφυγής συμφόρησης, ο χρονομετρητής αναμετάδοσης που λήγει ή η λήψη διπλών ACK μπορεί να υποδηλώσει σιωπηρά στον αποστολέα ότι συμβαίνει μια κατάσταση συμφόρησης δικτύου. Ο αποστολέας θέτει αμέσως το παράθυρο μετάδοσης στο μισό του τρέχοντος μεγέθους παραθύρου (το ελάχιστο παράθυρο συμφόρησης και το διαφημιζόμενο μέγεθος παραθύρου του δέκτη), αλλά σε τουλάχιστον δύο τμήματα. Εάν η συμφόρηση υποδείχθηκε από ένα χρονικό όριο, το παράθυρο συμφόρησης επαναφέρεται σε ένα τμήμα, το οποίο θέτει αυτόματα τον αποστολέα σε λειτουργία αργής εκκίνησης.

Fast Retransmit

Όταν λαμβάνεται ένας διπλότυπος λογαριασμός ACK, ο αποστολέας δεν γνωρίζει αν αυτό οφείλεται στο γεγονός ότι χάθηκε ένα τμήμα TCP ή απλά ότι ένα τμήμα καθυστέρησε και έλαβε εκτός σειράς τον παραλήπτη. Εάν ο δέκτης μπορεί να ανακατασκευάσει τμήματα, δεν πρέπει να είναι πολύ πριν ο δέκτης στείλει την τελευταία αναμενόμενη επιβεβαίωση. Συνήθως δεν θα πρέπει να παραλαμβάνονται περισσότερα από ένα ή δύο αντίγραφα ACK όταν υπάρχουν απλές συνθήκες εκτός λειτουργίας. Εάν, ωστόσο, ληφθούν από τον αποστολέα περισσότερα από δύο αντίγραφα ACK, αποτελεί ένδειξη ότι έχει χαθεί τουλάχιστον ένα τμήμα. Ο αποστολέας TCP θα αναλάβει αρκετό χρόνο έχει λήξει για όλα τα τμήματα να διορθωθούν σωστά από το γεγονός ότι ο δέκτης είχε αρκετό χρόνο για να στείλει τρεις διπλές ACKs.

Fast Recovery

Εφόσον ο αλγόριθμος γρήγορης αναμετάδοσης χρησιμοποιείται όταν λαμβάνουν διπλότυπα ACK, ο αποστολέας TCP έχει σιωπηρή γνώση ότι υπάρχουν δεδομένα που εξακολουθούν να ρέουν στον δέκτη. Γιατί; Ο λόγος είναι επειδή οι διπλοί ACK μπορούν να δημιουργηθούν μόνο όταν λαμβάνεται ένα τμήμα. Αυτό αποτελεί ισχυρή ένδειξη ότι δεν υπάρχει σοβαρή συμφόρηση του δικτύου και ότι το χαμένο τμήμα ήταν ένα σπάνιο γεγονός. Έτσι, αντί να μειωθεί απότομα η ροή των δεδομένων μεταβαίνοντας σε αργή εκκίνηση, ο αποστολέας εισέρχεται μόνο στη λειτουργία Αποφυγής συμφόρησης.

Μηχανισμοί ελέγχου και διόρθωσης σφαλμάτων

Χρησιμοποιούνται διότι μας δίνουν την δυνατότητα να ελέγξουμε το άν έχει πραγματοποιηθεί αλλοίωση κάπου τμήματος δεδομένων που έχουν οριστεί για μεταφορά, τον εντοπισμό τους και εν τέλει την διόρθωση τους. [9]
Για να επιτευχθεί αυτό έχει οριστεί προκαταβολικά τοσο από τους αποστολείς όσο και από τους παραλήπτες ότι όλα τα προς μετάδοση πακέτα θα πρέπει να έχουν προκαθορισμένες ιδιότητες. Ο αποστολέας μετατρέπει κάθε πακέτο προς αποστολή ώστε να πληροί αυτές τις ιδιότητες και ο παραλήπτης απορρίπτει οποιοδήποτε πακέτο δεν τις πληροί.

Μια καλή ιδιότητα ή οποία μπορεί να χρησιμοποιηθεί για αυτόν ακριβώς τον σκοπό, θα πρέπει εύκολα να μπορεί να εφαρμοστεί σε οποιοδήποτε αυθαίρετο πακέτο και επίσης θα πρέπει να μπορούμε εύκολα να ελέγξουμε αν κάποιο πακέτο πληροί αυτή την ιδιότητα είτε χρησιμοποιώντας το λογισμικό είτε το υλικό του υπολογιστή. Καλό θα ήταν να τονιστεί ότι και τα λάθη φαίνεται να διατηρούν αυτές τις ιδιότητες οπότε θα πρέπει να αναπτύξουμε ακόμα περισσότερο τους μηχανισμούς αναγνώρισης τών λαθών.

Η πιο ευρέως διαδεδομένη ιδιότητα είναι το parity. Σύμφωνα με αυτή την ιδιότητα όλα τα μεταδιδόμενα πακέτα θα έχουν κατά την παράδοση του έναν περιττό αριθμό από άσσους (1) – τα πακέτα είναι της μορφής 0-1 . Έτσι χρησιμοποιείται ένα bit το οποίο αφιερώνεται για αυτόν ακριβώς τον λόγο. Με αυτόν τον τρόπο όχι μόνο έχουμε ένα αφιερωμένο διφύο για αυτον τον λόγο αλλά επίσης είναι και εύκολος ο έλεγχος για το αν πληρείται αυτή η ιδιότητα.

Μία άλλη ιδιότητα που χρησιμοποιείται είναι η απόσταση Hamming. Σύμφωνα με αυτή έστω ότι έχουμε 2 λέξεις τών n bit: w , w’ . Ώς απόσταση Hamming ορίζεται η HD(w,w') , η οποία ισούται με το πλήθος τών θέσεων στις οποίες διαφέρουν. Για τα περισσότερα κανάλια η πιθανότητα ένα λάθος να μετατρέπει μία προς αποστολή λέξη w σε μία ληφθείσα λέξη w’ , μειώνεται όσο αυξάνεται η απόσταση Hamming.

Επίσης χρησιμοποιείται και η μέθοδος του CheckSum, κατά την οποία το πακέτο που στέλνεται διαβάζεται ώς μία ακολουθία από λέξεις προκαθορισμένου μεγέθους, το άθροισμα τών οποίων είναι μία προκαθορισμένη τιμή, χωρίς περιορισμόστην χρησιμοποιούμενη αριθμητική και το χρησιμοποιούμενο σύστημα. Δηλαδή έστω ότι έχουμε τις προς αποστολή ακολουθίες 010001110 και 111001011. Ο αποστολέας υπολογίζει το άθροισμα τους στο δυαδικό σύστημα και ο παραλήπτης θα πρέπει να ελέγξει για το άν οι λέξεις που έχει λάβει πληρούν αυτή την προυπόθεση (δηλαδή το άθροισμα τους ισούτε με αυτό που του υπέδειξε ο αποστολέας). Το checksum είναι μία ιδιότητα η οποία χρησιμοποιείται και στα πρωτόκολλα UDP,IP,TCP .

Η τελευταία ιδιότητα που θα μελετήσουμε για την αναγνώριση λαθών είναι η μέθοδος CRC. Αυτή χρησμοποιείται ευρέως στα προτόκολλα του επιπέδου datalink και θεωρείται ώς η πιο αποδοτική στην εύρεση λαθών. Συγγεκριμμένα μία r-bit CRC μπορεί να αναγνωρίζει όλες εκτός από $\frac{1}{2^r}$ μοντέλα λαθών. Για παράδειγμα μία 16 bit CRC μπορεί να αναγνωρίσει λάθη μεγέθους έως και 16 bit , όλα τα λάθη με «βάρος» περιττό και όλα τα λάθη, εκτός από $\frac{1}{2^{16}}$ εξάψεις λαθών μεγαλύτερων τών 16 bit, κάτι το οποίο την κάνει εξαιρετικά αποδοτική καθώς η πιθανότητα να εμφανιστούν τέτοια λάθη είναι πολύ μικρή.

Η λειτουργία της CRC βασίζεται στο ότι ο παραλήπτης και ο αποστολέας συμφωνούν σε μία γεννήτρια πολυωνύμου βαθμού G(x) . Η ιδιότητα είναι ότι η λέξη , η οποία αναγνωρίζεται ώς ένα πολυώνυμο θα μπορεί να διαιρείται με την G(x) . Κάθε μεταδιδόμενο πακέτο θα πρέπει δηλαδή να είναι πολλαπλάσιο της G(x), και ο παραλήπτης διαιρεί κάθε ληφθέν πακέτο με την G(x), και απορρίπτει όσα αφήνουν υπόλοιπο διαφορετικό του μηδέν (0) .

Μέθοδοι Διόρθωσης Λαθών

Τα πρωτόκολλα τα οποία έχουν συμφωνηθεί ώς προς χρήση και χρησιμοποιούνται για την εύρεση και διόρθωση λαθών έχουν εξελιχθεί ώς προς την ακρίβεια, την ταχύτητα και την αποδοτικότητα τους από το 1978 , όποτε και το πρωτόκολλο Xmodem έγινε το standard πρωτόκολλο πρός χρήση [12] . Περιληπτικά μπορούμε να πούμε ότι σε όλα τα πρωτόκολλα τα δεδομένα συμπιέζονται σε block ενός προκαθορισμένου μεγέθους σε byte και στέλνονται προς τον κόμβο προορισμού τους και ανάλογα με τα αποτελέσματα του ελέγχου τους ο παραλήπτης στέλνει πίσω θετικά (ACK) ή αρνητικά (NACK) διαπιστευτήρια.

Για την διόρθωση λαθών χρησιμοποιούνται πρωτόκολλα, τα οποία αφού πρώτα αναγνωριστεί το λάθος χρησιμοποιούνται εντοπισμένα και το διορθώνουν. Συνήθως το λάθος είναι μεμονωνμένο bit , οπότε αρκεί μια αντιστροφή bit ωστόσο το σύνολο τών μηχανισμών δεν είναι τόσο απλοικό. Κάποια πρωτόκολλα που χρησιμοποιούνται είναι τα εξής:

ARQ

Το ARQ είναι μια μέθοδος ελέγχου σφαλμάτων για τη μετάδοση δεδομένων που χρησιμοποιεί κωδικούς ανίχνευσης σφαλμάτων, μηνύματα επιβεβαίωσης ή / και αρνητικής επιβεβαίωσης και χρονικά όρια για την επίτευξη αξιόπιστης μετάδοσης δεδομένων. Μια επιβεβαίωση είναι ένα μήνυμα που αποστέλλεται από τον δέκτη για να υποδείξει ότι έχει λάβει σωστά ένα πλαίσιο δεδομένων. Συνήθως, όταν ο πομπός δεν λαμβάνει την επιβεβαίωση πριν εμφανιστεί το χρονικό όριο (δηλ. Εντός εύλογου χρονικού διαστήματος μετά την αποστολή του πλαισίου δεδομένων), αναμεταδίδει το πλαίσιο μέχρι να ληφθεί σωστά ή το σφάλμα να παραμείνει πέρα από έναν προκαθορισμένο αριθμό αναμεταδοτήσεων. Τρείς τύποι πρωτοκόλλων που ανήκουν σε αυτή την μέθοδο είναι:

1) Stop-and-wait ARQ
2) Go-Back-N ARQ
3) Selective Repeat ARQ

Το ARQ είναι κατάλληλο αν το κανάλι επικοινωνίας έχει διαφορετική ή άγνωστη χωρητικότητα, όπως συμβαίνει στο Internet. Ωστόσο, το ARQ απαιτεί τη διαθεσιμότητα ενός οπίσθιου καναλιού, οδηγεί σε πιθανή αύξηση της καθυστέρησης λόγω αναμετάδοσης και απαιτεί τη διατήρηση των buffer και των χρονομετρητών για αναμετάδοση, η οποία στην περίπτωση της συμφόρησης του δικτύου μπορεί να βλάψει τον server και τη συνολική χωρητικότητα του δικτύου. [13]

ECC

Ένας κωδικός διορθώσεως σφάλματος (ECC) ή κώδικας διόρθωσης σφάλματος (FEC) είναι μια διαδικασία προσθήκης περιττών δεδομένων ή δεδομένων ισοτιμίας σε ένα μήνυμα, έτσι ώστε να μπορεί να ανακτηθεί από ένα δέκτη, ακόμη και όταν έχουν σημειωθεί διάφορα σφάλματα η δυνατότητα χρήσης του κώδικα) εισήχθησαν είτε κατά τη διάρκεια της διαδικασίας μετάδοσης είτε κατά την αποθήκευση. Δεδομένου ότι ο δέκτης δεν χρειάζεται να ζητήσει από τον αποστολέα την αναμετάδοση των δεδομένων, δεν απαιτείται οπτικός καναλιού στη διόρθωση σφαλμάτων προς τα εμπρός και επομένως είναι κατάλληλη για απλή επικοινωνία όπως η εκπομπή. Οι κώδικες διόρθωσης σφαλμάτων χρησιμοποιούνται συχνά στην επικοινωνία κατώτερου στρώματος, καθώς και για αξιόπιστη αποθήκευση σε μέσα όπως CD, DVD, σκληρούς δίσκους και μνήμη RAM. Οι ECC συνήθως χωρίζονται σε:

1) *Convolutional codes : Ελέγχονται bit προς bit. Είναι κατάλληλοι για εφαρμογή σε hardware και ο* Viterbi decoder *επιτρέπει την βέλτιστη δυνατή αποκωδικοποίηση.*
2) Block Codes : Ελέγχονται μπλοκ ανά μπλοκ. Πρώιμα παραδείγματα block codes είναι οι repetition codes, Hamming codes και multidimensional parity-check codes.

Hybrid Schemes

Αποτελούν ένα συνδυασμό τών ARQ και Forward Error Correction. Υπάρχουν δύο βασικές προσεγγίσεις:

1) Τα μηνύματα πάντα μεταδίδονται με FEC parity data και με error correction πληροφορίες . Ο παραλήπτης αποκωδικοποιεί τα μηνύματα χρησιμοποιώντας parity και απαιτεί επαναμετάδωση χρησιμοποιώντας μόνο ARQ μόνο άν τα δεδομένα parity δεν ήταν επαρκή για ικανοποιητική αποκωδικοποίηση.
2) Τα μηνύματα μεταδίδονται χωρίς FEC parity data, παρά μόνο με error correction πληροφορίες .Αν ο παραλήπτης παρατηρήσει κάποιο λάθος απαιτεί FEC πληροφορίες από τον αποστολέα χρησιμοποιώντας ARQ και τις χρησιμοποιεί για να αποκαταστήσει το απεσταλμένο μήνυμα.

Κεφάλαιο 4 : Κανόνες και μεθοδολογία σχεδιασμού πρωτοκόλλου

Εισαγωγή

Ο όρος πρωτόκολλο στην πραγματικότητα αντιστοιχεί σε μία συλλογή από «συμφωνίες», που τίθενται στα δύο μίας μορφής επικοινωνίας. Στην πρώτη φάση για να εκκινήσουμε την επικοινωνία τών κόμβων που συμμετέχουν θα ορίσουμε μία χειρονομία handshake. Η διαδικασία αυτή δεν είναι απαραίτητα επιτυχήςκαθώς μπορεί να αγνοηθεί (Server Time-out), είτε να γίνει άρνηση της από το άλλο μέρος (Connection Closed By Host) ακόμη και να διακοπεί κατά την διάρκεια της (Connection Reset By Peer) είτε να προσπαθήσουμε να εκκινήσουμε αυτή την διαδικασία προς λάθος ξενιστή (Connection Denied) . [25]Εφόσων αυτή η διαδικασία πραγματοποιηθεί επιτυχώς, ξεκινάει η ανταλλαγή πληροφοριών και η διαδικασία ολοκληρώνεται με το κλείσιμο της επικοινωνίας.

Σαν παράδειγμα μπορούμε να έχουμε την ανθρώπινη επικοινωνία. Το στάδιο handshake μπορεί να θεωρηθεί ώς ένας απλό χαιρετισμός και η ανταλλαγή τών πληροφοριών, επιτυγχάνεται μέσω της γλώσσας που χρησιμοποιείται. Όπως και για την ανθρώπινη γλώσσα πήρε χιλιάδες χρόνια μέχρι να πάρει την σημερινή της μορφή (και ακόμα εξελίσσσεται), έτσι και τα πρωτόκολλα συνεχώς εξελίσσονται , αλλάζουν μορφή και προσαρμόζονται στα νέα δεδομένα.

Παραδοχές σχεδιασμού

Ίσως το πιο σημαντικό κομμάτι με το οποίο πρέπει να ξεκινάει κάθε σχεδιασμός πρωτοκόλλου είναι η μελέτη άλλων πρωτοκόλλων.Η πρώτη απόφαση που πρέπει να παρθεί για το νέο πρωτόκολλο είναι εάν θα είναι binary ή text-based.

Text based

Τα Text-based πρωτόκολλα συμπεριλαμβάνουν το HTTP πρωτόκολλο, το οποίο χρησιμοποιείται και για την περιήγηση σε ιστοσελίδες. Το μεγαλύτερο όφελος από την χρήση του είναι ότι είναι αρκετά κατανοητό από τους ανθρώπους, κάτι το οποίο κάνει την δημιουργία και το de-bugging σε αυτό, εύκολο. Το κύριο μειωνέκτημα του είναι ότι είναι λιγότερο ακριβές και αρκετά χρονοβόρο στην δημιουργία του, δεν είναι άμεση η δυνατότητα πιστοποίησης ότι ανταποκρίνεται στις απατήσεις μας , συν ότι είναι αρκετά επιρρεπές σε προβλήματα που προκύπτουν από λανθασμένη κωδικοποίηση κειμένου (TE). Δυστυχώς αρκετά δυσπρόσητο και επιρρεπές στα λάθη σαν πρωτόκολλο επικοινωνίας κάτι το οποίο έχει σαν αποτέλεσμα να χρησιμοποιείται σπάνια , και κυρίως σε δικτυακές εφαρμογές.

Ένα μεγάλο προσόν που έχουν τα συγκεκριμμένα πρωτόκολλα είναι ότι δεν επιρρεάζονται από την κατάσταση endianness (που προκύπτει ανάλογα με το αν χρησιμοποιείται Little/Big Endian), που είναι και οι διατάξεις που χρησιμοποιούνται από τις ηγετικές εταιρίες στον χώρο τών επεξεργαστών.Little Endian διάταξη σημαίνει ότι τα λιγότερα σημαντικά bits χρησιμοποιούνται στο μπροστινό μέρος ενός byte, ενώ η Big Endian είναι το ακριβώς αντίθετο. Σαν παράδειγμα έστω ό αριθμός 14 (hexidecimal 0x0E), στην little endian είναι ένας 4-byte ακέραιος 0E 00 00 00, ενώ στην big endian είναι : 00 00 00 0E. Προφανώς μία σύγχιση τών δύο μεθόδων θα έχει καταστρεπτικά αποτελέσματα για το πρωτόκολλο μας.

Binary

Χρησιμοποιώντας τα binary πρωτόκολλα επιλύουμε αυτό το πρόβλημα μέσω της χρήσης ανάμεικτων endian περιβάλλοντων, προσθέτοντας έναν αριθμό στο αρχικό μέρος του αριθμού συνήθως 2 bytes με γνωστές τιμές. Μέσω της ανάγνωσης τους γνωρίζουμε ποια τεχνική endianness χρησιμοποιείται για την αναπαράσταση τών δεδομένων. Ένα παράδειγμα είναι η χρήση ‘MM’ όπως σε TIFF κεφαλίδες αρχείων για να ενδείξουμε σε big endian (MSB) και ‘II’ για να δείξουμε σε little endian (LSB) σειρά byte. Μπορούμε στην συνέχεια να χρησιμοποιήσουμε διαφορετική ρουτίνα parsing ή να εναλλάξουμε τις δύο τεχνικές κατά βούληση κατά την διάρκεια του parsing.

Έπειτα θα πρέπει να οργανωθούν τα μηνύματα έτσι ώστε οι ιδιότητες που χαρακτηρίζουν το πρωτόκολλο να είναι σαφείς σε κάθε ένα από τα σταλθέντα μηνύματα, έτσι ώστε ο παραλήπτης να μην χρειάζεται να κάνει υποθέσεις για το περιεχόμενο τους. Αυτό σημαίνει ότι θα πρέπει να έχουν χρησιμοποιηθεί σαφείς και ξεκάθαρες αρχές κρυπτογράφησης στα μηνύματα, και ποιες απαιτήσεις υπάρχουν από αυτά. Π.χ. Είναι απαραίτητο τα μηνύματα να είναι εμπιστευτικά (confidentiality) ?

Χαρακτηριστικά σχεδιασμού

Υπάρχουν ορισμένα χαρακτηριστικά και ιδιότητες που θα πρέπει να πληροί το πρωτόκολλο που επιχειρούμε να δημιουργήσουμ προκειμένου να είναι σωστά και πλήρως σχεδιασμένο, αλλά και να μπορεί να χρησιμοποιηθεί και να ενσωματωθεί σε συστήματα συμβάλλοντας θετικά στην λειτουργία τους.

Ανθεκτικότητα

Ώς ανθεκτικότητα πρωτοκόλλου ορίζουμε την ικανότητα τα του να διατηρεί την συνοχή του και την αδιάκοπη και σωστή λειτουργία του ανεξαρτήστως του περιβάλλοντας, δηλαδή να είναι ανθεκτικά απέναντι σε δυσλειτουργίες. Τις δυσλειτουργίες αυτές θα διαχωρίσουμε σε δύο κατηγορίες.

Ανθεκτικότητα σε Απλές αποτυχίες

Χαρακτηριστικά παραδείγματα αποτελούν οι απώλειες πακέτων την καθυστερημένη παράδοση πακέτων και τα timeouts . Αυτό σημαίνει πώς τα timeouts τών διαφορετικών παραληπτών θα πρέπει να έχουν οριστεί έχοντας υπόψιν πιθανά σενάρια διαφορών στην επικοινωνία και να προσαρμόζονται έτσι ώστε να ανταπεξέρχονται σε διαφορετικές περιστάσεις επαρκώς. Θα πρέπει να λαμβάνονται υπόψης επαναλαμβανόμενες απώλειες πακέτων και να έχουν τεθεί οι κατάλληλες προυποθέσεις ώστε τέτοια φαινόμενα να αντιμετωπίζονται, π.χ. μέσω επαναμετάδοσης χαμένων πακέτων είτε εγκαίρως είτε μετά το πέρας της αποστολής τών υπόλοιπων.

Ανθεκτικότητα σε δυσλειτουργίες

Σε αυτό το κομμάτι θα αναλύσουμε τυχόν δυσλειτουργίες που ενδέχεται να παρουσιαστούν στο hardware, τόσο κάποιου υπολογιστή όσο και σε οποιοδήποτε τμήμα του δικτύου (κόμβο ή κανάλι). Σε αυτή την περίπτωση το σύστημα θα πρέπει να έχει προκαθορισμένες μεθόδους ώστε να είναι σε θέση να αυτο-σταθεροποιηθεί. Παραδείγματος χάριν στην περίπτωση που καταρεύσει ένα τμήμα του δικτύου, τότε βέλτιστα θα πρέπει να μην ακινητοποιείται το δίκτυο αλλά να μπορεί να εποκλειστεί το συγκεκριμμένο τμήμα ώστε να μην προσθέτει καθυστερήσεις και να μπορεί να βρεθεί εγκαίρως ένα νέο μονοπάτι για να εξυπηρετήσει τους κόμβους που χρησιμοποιούσαν το καταρριφθέν μονοπάτι, με την ελάχιστη επιβάρυνση τών υπόλοιπων τμημάτων του δικτύου.

Ένα ακόμα συμβάν που θα πρέπει να έχει ληφθεί υπόψιν είναι το γεγονός ότι σχεδόν πάντα θα υπάρχει κάποιο πρόβλημα στην μετάδοση πακέτων. Αυτό μπορεί να είναι είναι μη μετάδοση τους , είτε λανθασμένη μετάδοση τους. Σε αυτή την περίπτωση θα πρέπει να έχουν παρθεί εξ αρχής μέτρα ώστε να μπορεί και στα δύο μέρη του δικτύου που προσπαθούν να επικοινωνήσουν, ώστε να μπορούν να αναγνωρίσουν το λάθος έγκαιρα και να το διορθώσουν. Σε αυτό το κομμάτι συμβάλλουν οι τεχνικές αναγνώρισης και διόρθωσης σφαλμάτων που αναλύωωνται μετέπειτα.

Σημαντικός είναι ο τομέας της εμπιστοσύνης. Αυτό σημαίνει ότι στο πρωτόκολλο θα πρέπει να υπάρχουν σαφείς οδηγίες πρός όλα τα αναμεμειγμένα μέλη ώς προς το ποιες υποθέσεις θα πρέπει να λαμβάνουν κατά την διάρκεια της επικοινωνίας. Χαρακτηριστικό παράδειγμα αποτελεί ένας Server , οποίος θα πρέπει να είναι σίγουρο ότι εκδίδει τα σωστά timestamps για τα μηνύματα τα οποία αποστέλει.

Τέλος θα πρέπει το πρωτόκολλο να σχεδιαστεί ώστε να είναι αδιαπέραστο από επιθέσεις [26]. Ένα από τα καλύτερα μέσα αντιμετώπισης επιθέσεων είναι η κρυπτογράφηση τών μηνυμάτων. Οι επιθέσεις συνήθως δεν στοχεύουν στην κατάρριψη πακέτων διότι κάτι τέτοιο θα είχε απλά ώς αποτέλεσμα ο δέκτης να ζητήσει την επαναποστολή του, αλλά τεμαχίζουν τα μηνύματα και μετά ενώνουν τμήματα από διαφορετικά μηνύματα , αναμειγνύοντας έτσι τα αρχικά μηνύματα. Το πιο αποτελεσματικό μέσο αντιμετώπισης μίας τέτοιας επίθεση είναι η χρήση τών MAC τα οποία δεσμεύουν τμήματα πακέτων μεταξύ τους ώστε αυτά να μην μπορούν να διαχωριστούν.

Ασφάλεια

Η ασφάλεια ενός πρωτοκόλλου είναι ίσως από τις πιο σημαντικές ιδιότητες που θα πρέπει να το χαρακτηρίζουν. Εικότερα όταν τα μηνύματα μπορεί να έχουν χαρακτηριστεί ώς εμπιστευτικά είναι απαραίτητο να υπάρχει εμπιστοσύνη στο χρησιμοποιούμενο πρωτόκολλο να μεταδώσει σωστά και με ασφάλεια τα μηνύματα.

Η κρυπτογραφία παίζει έναν από τους πιο σημαντικούς ρόλους στην ασφάλεια του πρωτοκόλου. Για αυτό τον λόγο στην κρυπτογραφία έχουμε την ύπαρξη κλειδιών αποκρυπτογράφησης αλλά και ορισμένες αρχές οι οποιες την διέπουν. Στν κρυπτογραφία υπάρχουν πολλές μεθόδοι και αρχές που ακολουθούνται. Παραδείγματος χάριν υπάρχει το sign (signing). Θεωρούμε ότι όταν κάποιος παραλήπτης μηνύματος κάνει signing σε κάποιο ήδη κρυπτογραφημένο μήνυμα , τότε δεν έχει γνώση του περιεχομένου του.

Το θέμα όμως παραμένει ότι η κρυπτογραφία παρά τα προτερήματα της μπορεί να οδηγήσει σε συγχίσεις μηνυμάτων. Παραδείγματος χάριν δεν υπάρχει εγγύηση ότι το κρυπτογραφημένο μήνυμα δύο συγχωνευμένων μηνυμάτων θα είναι ίδιο με την συγχώνευση τών δύο κρυπτογραφημένων μηνυμάτων. Αυτό μας οδηγεί και σε μία άλλη αρχή που υπάρχει στην κρυπτογραφία η οποίο καθιστά αναγκαίο να είναι γνωστές όλες οι αρχές που διέπουν την κρυπτογράφηση τών μηνυμάτων.

Στο επόμενο κεφάλαιο θα εντρυφίσουμε στον τομέα της ασφάλειας.

Κεφάλαιο 5 : Εμβάθυνση στην ασφάλεια

Ασφάλεια Δικτύων

Τι είναι και πώς λειτουργεί?

Σύμφωνα με την CISCO, ηγετική εταιρία στον τομέα τών δικτύων και τηλεπικοινωνιών, ώς ασφάλεια δικτύου ορίζεται οποιαδήποτε δραστηριότητα που έχει σχεδιαστεί για την προστασία της χρηστικότητας και της ακεραιότητας του δικτύου και των δεδομένων που διακινούνται σε αυτό. Περιλαμβάνει τεχνολογίες υλικού και λογισμικού. Η αποτελεσματική ασφάλεια δικτύου διαχειρίζεται την πρόσβαση στο δίκτυο. Στόχος αντιμετώπισης είναι μια ποικιλία απειλών και τους εμποδίζει να εισέλθουν ή να εξαπλωθούν στο δίκτυό . [20] Η ασφάλεια δικτύου συνδυάζει πολλαπλά στρώματα άμυνων στα όρια του δικτύου και στο ίδιο το δίκτυο. Κάθε στρώμα ασφάλειας δικτύου εφαρμόζει πολιτικές και ελέγχους. Οι εξουσιοδοτημένοι χρήστες αποκτούν πρόσβαση σε πόρους δικτύου, αλλά οι κακόβουλοι χρήστες εμποδίζονται από το να πραγματοποιούν εκμεταλλεύσεις και απειλές στο δίκτυο.

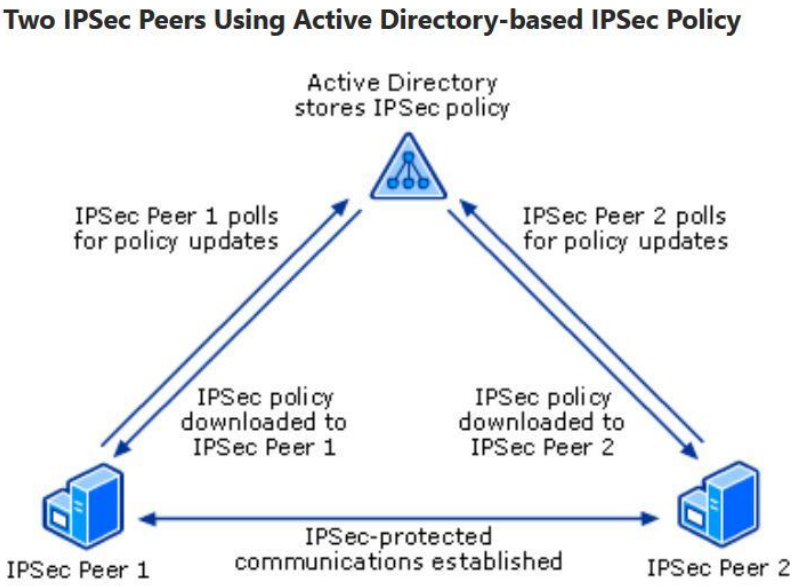
Ο ορισμός της CISCO αν και πλήρης είναι αρκετά απλοικός δεδομένου ότι πρέπει να γίνεται κατανοητός από τους δυνητικούς πελάτες της. Η ασφάλεια δικτύων αποτελείται από τις πολιτικές και τις πρακτικές που υιοθετούνται για την πρόληψη και την παρακολούθηση μίας ενδεχόμενης μη εξουσιοδοτημένης πρόσβασης, κατάχρησης, τροποποίησης ή άρνησης ενός δικτύου υπολογιστών και πόρων που είναι προσβάσιμοι από το δίκτυο. Η ασφάλεια δικτύου περιλαμβάνει την εξουσιοδότηση πρόσβασης σε δεδομένα του δικτύου, το οποίο ελέγχεται από το διαχειριστή του δικτύου.

Η ασφάλεια δικτύου καλύπτει μια ποικιλία δικτύων υπολογιστών, τόσο δημόσιων όσο και ιδιωτικών, που χρησιμοποιούνται σε διεργασίες στην καθημερινή ζωή. Αυτές περιλαμβάνουν ενδεικτικά την διεξαγωγή συναλλαγών και επικοινωνιών μεταξύ επιχειρήσεων, κυβερνητικών φορέων και ατόμων. Τα δίκτυα μπορούν να είναι ιδιωτικά, όπως μέσα σε μια εταιρεία, και άλλα ενδέχεται να είναι ανοικτά στην πρόσβαση του κοινού. Η ασφάλεια δικτύων εμπλέκεται σε οργανισμούς, επιχειρήσεις και άλλους τύπους ιδρυμάτων, αλλά το πιο σημαντικό είναι ότι περιλαμβάνει τους ίδιους τους χρήστες. [21] Καταλαβαίνουμε ότι η ασφάλεια δικτύων είναι άρρηκτα συνδεδεμένη με την ασφάλεια πρωτοκόλλων καθώς τα πρωτόκολλα είναι αυτό που συμβάλλουν σε μεγάλο βαθμό στην επίτευξη ασφαλείας στα δίκτυα.

Πώς ορίζεται η ασφάλεια πρωτοκόλλων?

Η ασφάλεια πρωτοκόλλου Internet (IPSec) είναι ένα πλαίσιο ανοικτών προτύπων για τη διασφάλιση ιδιωτικών και ασφαλών επικοινωνιών μέσω δικτύων πρωτοκόλλου Internet (IP) μέσω της χρήσης κρυπτογραφικών υπηρεσιών ασφαλείας. [23] Αρχικά τέτοιου είδους ασφάλεια περιοριζόταν στο επίπεδο Εφαρμογών. [24] Το IPSec υποστηρίζει την ακεραιότητα δεδομένων σε επίπεδο δικτύου, την εμπιστευτικότητα των δεδομένων, την εξακρίβωση προέλευσης δεδομένων και την προστασία επανάκλησης. Επειδή το IPSec είναι ενσωματωμένο στο στρώμα Internet (layer 3), παρέχει ασφάλεια σχεδόν για όλα τα πρωτόκολλα της σουίτας TCP / IP και επειδή η εφαρμογή IPSec εφαρμόζεται με διαφάνεια στις εφαρμογές, δεν χρειάζεται να διαμορφώσετε ξεχωριστή ασφάλεια για κάθε εφαρμογή που χρησιμοποιεί TCP / IP. [22]. Είναι ιδιαίτερα αποτελεσματική ενάντια σε NBAs , DC , DT, UCT , ACS και επιβάλλεται μέσω του συνδυασμού βασιζόμενων σε ξενιστή IPSec packet φίλτρων πακέτων και την χρήση εμπιστευτικών γραμμών.

Η ασφάλεια πρωτοκόλλων ενσωματώνεται σε μεγάλο βαθμό μέσα στο ίδιο το λειτουργικό σύστημα τών υπολογιστών που συμμετέχουν σε ένα δίκτυο. Παρακάτω βλέπουμε την Βασισμένη σε Active – Directory πολιτική που διαμοιράζεται σε δύο IPSec ομότιμους και τα IPSec προστετευόμενα σενάρια επικοινωνίας μεταξύ τους.



Εικόνα 5.1 : – Directory πολιτική που διαμοιράζεται σε δύο IPSec ομότιμους

Το IPSec είναι μια τεχνολογία ασφάλειας γενικής χρήσης , που μπορεί να χρησιμοποιηθεί για την ασφάλεια της κυκλοφορίας δικτύου σε πολλά σενάρια.

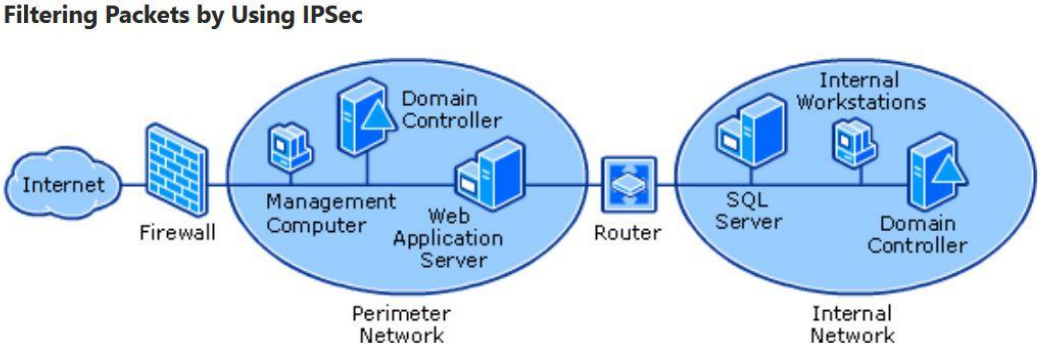
Ωστόσο, πρέπει να εξισορροπιστεί η ανάγκη για ασφάλεια με την πολυπλοκότητα της ρύθμισης των πολιτικών IPSec. Ως ώρας λόγω της έλλειψης κατάλληλων προτύπων, το IPSec δεν είναι κατάλληλο για ορισμένους τύπους σύνδεσης . Παρακάτω περιγράφονται ορισμένα σενάρια IPSec που συνιστώνται και για τα οποία συνίσταται η χρήση IPSec .

- Φιλτράρισμα πακέτων (PF)
- Ασφάλεια End-to-end μεταξύ διαφορετικών φιλοξενιτών (ξενιστών)
- Ασφάλιση server
- End-to-end κίνηση μέσω ενός SSNAT
- L2TP Μέσω (L2TP/IPSec) , για απομακρυσμένη είσοδο και εικονική ιδιωτικού δικτύου (VPN) σύνδεση
- Site-to-site IPSec διοχέτευση μέσω IPSec gateways που δεν προέρχονται από την Microsoft

Τεχνικές ασφάλειας πρωτοκόλων

Packet Filtering

Αξίζει για να κατανοήσουμε πώς λειτουργεί η ασφάλεια πρωτοκόλλων να μελετήσουμε το φιλτράρισμα πακέτων. Το IPSec μπορεί να εκτελέσει φιλτράρισμα πακέτων με βάση τον ξενιστή για να παρέχει περιορισμένες δυνατότητες τείχους προστασίας για τελικά συστήματα. Μπορείτε να ρυθμίστεί το IPSec ώστε να επιτρέπει ή να αποκλείει συγκεκριμένους τύπους επισκεψιμότητας IP unicast με βάση τους συνδυασμούς διευθύνσεων προέλευσης και προορισμού και συγκεκριμένα πρωτόκολλα και συγκεκριμένες θύρες. Για παράδειγμα, σχεδόν όλα τα συστήματα που απεικονίζονται στο παρακάτω σχήμα μπορούν να επωφεληθούν από το φιλτράρισμα πακέτων για να περιορίσουν την επικοινωνία μόνο σε συγκεκριμένες διευθύνσεις και θύρες. Μπορείτε να ενισχύσετε την ασφάλεια χρησιμοποιώντας το φιλτράρισμα πακέτων IPSec για να ελέγξετε ακριβώς τον τύπο επικοινωνίας που επιτρέπεται μεταξύ των συστημάτων.



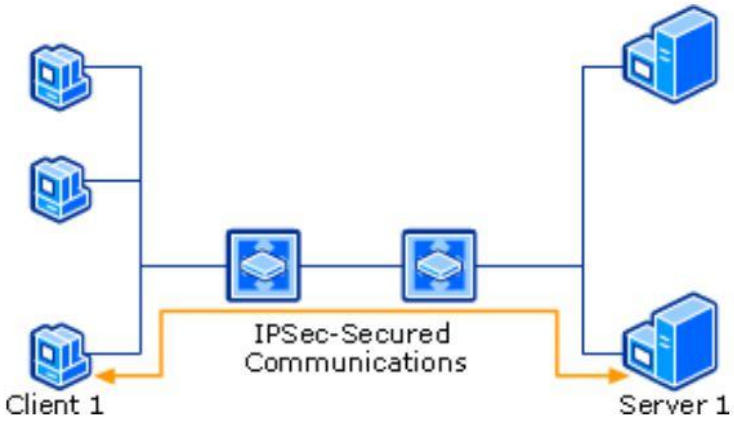
Εικόνα 5.2 : Απεικόνιση του Φιλτραρίσματος Πακέτων

End-to-End Ασφάλεια μεταξύ διακεκριμμένων ξενιστών

Το IPSec δημιουργεί εμπιστοσύνη και ασφάλεια στην επικοινωνία από διεύθυνση IP πηγής unicast σε διεύθυνση IP προορισμού unicast (end-to-end). Για παράδειγμα, το IPSec μπορεί να βοηθήσει στην ασφαλή κυκλοφορία μεταξύ διακομιστών Web (Web Servers) και διακομιστών βάσεων δεδομένων (DB Servers) ή ελεγκτών τομέα σε διαφορετικούς ιστότοπους. Όπως φαίνεται στο παρακάτω σχήμα, μόνο οι υπολογιστές αποστολής και λήψης πρέπει να γνωρίζουν το IPSec. Κάθε υπολογιστής χειρίζεται την ασφάλεια στο αντίστοιχο άκρο του και υποθέτει ότι το μέσο στο οποίο πραγματοποιείται η επικοινωνία δεν είναι ασφαλές. Οι δύο υπολογιστές μπορούν να τοποθετηθούν κοντά ο ένας στον άλλο, όπως σε ένα μόνο τμήμα δικτύου ή στο Internet.

Οι υπολογιστές ή τα στοιχεία δικτύου που δρομολογούν δεδομένα από πηγή σε προορισμό δεν απαιτούνται για την υποστήριξη του IPSec. Ωστόσο μπορεί να χρησιμοποιηθρί η λειτουργία μεταφοράς IPSec για να βοηθήσει στην ασφαλή κυκλοφορία κεντρικού υπολογιστή μέσω υπολογιστή που εκτελεί ISA Server και λειτουργεί ως μεταφραστής διεύθυνσης δικτύου εάν το ISA (ή οποιαδήποτε άλλη συσκευή NAT) δεν χρειάζεται να επιθεωρήσει την κίνηση μεταξύ του δύο οικοδεσπότες. Η λειτουργία μεταφοράς IPSec χρησιμοποιείται για την προστασία της κυκλοφορίας μεταξύ υπολογιστών και μπορεί να παρέχει ασφάλεια μεταξύ υπολογιστών που βρίσκονται στο ίδιο τοπικό δίκτυο (LAN) ή συνδέονται μέσω ιδιωτικών συνδέσεων δικτύου WAN.

Παρακάτω βλέπουμε το πώς οι Domain Controllers καλούνται στα διαφορετικά άκρα ενός firewall. Η IPSec , μπορεί να χρησιμοποιηθεί τόσο για την εξασφάλιση της κίνησης στο δίκτυο , όσο και για την εξασφάλιση της κίνησης μεταξύ δύο domain controllers στο ίδιο domain.



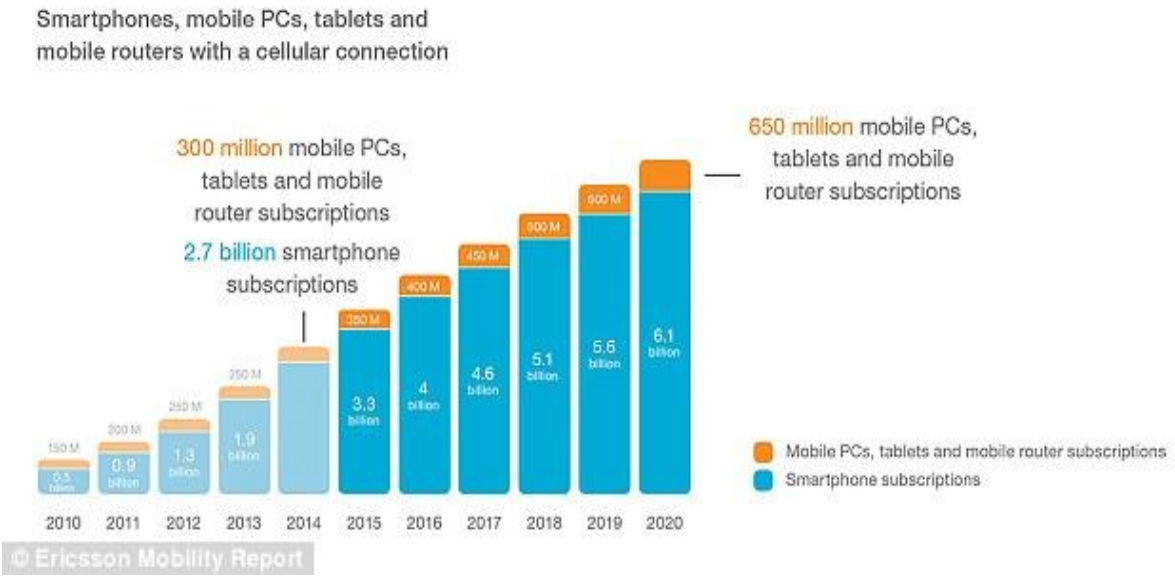
Εικόνα 5.3 : Κλήση Domain Controllers στα διαφορετικά άκρα ενός firewall

Κεφάλαιο 6 : Που θα κινηθεί ο τομέας στο μέλλον?

Εισαγωγικά

Πριν μπορέσουμε να εμβαθύνουμε σε αυτή την ερώτηση θα πρέπει να κατονομάσουμε και να εξηγήσουμε ορισμούς και τεχνολογίες που εφαρμόζονται.

Αρχικά θα πρέπει να αναγνωρίσουμε το ότι η ανάπτυξη της τεχνολογίας τών συσκευών κινητής επικοινωνίας έχει αυξήσει σημαντικά το ποσοστό χρήσης τους στην καθημερινή ζωή. Με την συνεχή μείωση του κόστους απόκτησης αλλά και τών υπολογιστικών πόρων που διαθέτουν, το ποσοστό τών ανθρώπων που επενδύουν σε τέτοιες συσκευές έχει αυξηθεί σημαντικά τα τελευταία χρόνια. Αυτό έχει ώς αποτέλεσμα πολλές από αυτές τις συσκευές να μπορούν να χρησιμοποιηθούν κακόβουλα. Ωστόσο ή αύξηση τους σημαίνει ότι τα παγκόσμια δίκτυα θα πρέπει να είναι σε θέση να προσαρμοστούν έτσι ώστε να μπορούν να ενσωματώσουν όλες αυτές τις συσκευές. [14] Αυτό σημαίνει είτε ότι θα πρέπει να μεγαλώσουν σε μέγεθος , είτε ότι θα αναβαθμίσουν και ενημερώσουν τα πρωτόκολλα τους.



Εικόνα 6.1: Πλήθος κινητών συσκευών σε χρήση τα τελευταία χρόνια

Η ανάπτυξη τών δικτύων θα έχει ώς αποτέλεσμα και την αύξηση τών οικονομικών κεφαλαίων που δαπανώνται στον τομέα . Αυτό θα γίνει εμφανές κυρίως αφού οι επιχειρήσεις συνειδητοποιήσουν ότι μεγάλο μέρος της αξίας επιχειρήσεων έγκειται στις πληροφορίες που διαχειρίζονται , και αυτό ισχύει κυρίως για τις πληροφορίες που κρατάνε για τους χρήστες τους . Και όσο τα δίκτυα μεγαλώνουν και ταυτόχρονα όλο και περισσότερες συσκευές αντιπροσωπεύουν ένα χρήστη σε αυτά, η αξία τών πληροφοριών πολλαπλασιάζεται .

Προβλέψεις για το μέλλον του Internet και τών Δικτύων

Πιο συγκεκριμμένα οι προβλέψεις από το γνωστό λογισμικό καταπολέμησης κακόβουλου λογισμικού Norton καταγράφονται στην ακόλουθη σελίδα: [15]

- **Μέχρι το 2020 υπολογίζεται ότι μπορεί να υπάρχουν έως και 21 δισεκατομμύρια συσκευές συνδεδεμένες στο ίντερνετ.**
  Οι συσκευές αυτές όμως δεν περιορίζονται στα κινητά , λάπτοπ, τάμπλετ και υπολογιστές. Τα τελευταία χρόνια οι «smart» τεχνολογίες που επιτρέπουν στις συσκευές να έχουν δυνατότητες υπολογιστών έχουν ενθυλακωθεί ακόμη και σε ψυγεία ή πλυντήρια. [18] [19] Για να γίνει κατανοητό το μέγεθος αυτής της εξέλιξης, το 2015 υπήρχαν μόλις 4,9 δισεκατομμύρια διασυνδεδεμένες συσκευές.

- **Hackers will continue to use IoT devices to facilitate DDoS attacks**
  Τον Οκτώβριο του 2016, ο κόσμος εισήχθη στο πρώτο "κακόβουλο λογισμικό" του IoT, το οποίο είναι ένα στέλεχος κακόβουλου λογισμικού που μπορεί να μολύνει συνδεδεμένες συσκευές όπως DVR, κάμερες ασφαλείας και άλλα. Το κακόβουλο λογισμικό Mirai έχει πρόσβαση στις συσκευές χρησιμοποιώντας τον προεπιλεγμένο κωδικό πρόσβασης και τα ονόματα χρηστών. Το κακόβουλο πρόγραμμα μετατρέπει έπειτα τις επηρεαζόμενες συσκευές σε ένα botnet για να διευκολύνει μια επίθεση Distributed Denial of Service (DDoS). Αυτή η επίθεση κατέληξε να πλημμυρίζει μία από τις μεγαλύτερες εταιρίες φιλοξενίας ιστοσελίδων στον κόσμο, φέρνοντας πολλούς σημαντικούς, γνωστούς ιστοτόπους και υπηρεσίες, σε ένα σβήσιμο για ώρες.

- **Πολλές πόλεις θα γίνουν «smart»**
  Οι καταναλωτές δεν θα είναι οι μόνοι που χρησιμοποιούν συσκευές IoT. Οι πόλεις και οι επιχειρήσεις, προσπαθώντας πάντα να γίνουν πιο αποδοτικές και να εξοικονομήσουν χρόνο και χρήμα, θα αρχίσουν επίσης να υιοθετούν "έξυπνες" τεχνολογίες. Σημαίνει ότι οι πόλεις θα είναι σε θέση να αυτοματοποιήσουν, να διαχειρίζονται εξ αποστάσεως και να συλλέγουν δεδομένα μέσω περιπτέρων επισκεπτών, συστημάτων παρακολούθησης βιντεοκάμερας, σταθμών ενοικίασης ποδηλάτων και ακόμη και ταξί.

- **Ανάπτυξη της AI**
  Έξυπνοι οικιακοί κόμβοι, θερμοστάτες, συστήματα φωτισμού και ακόμη και καφετιέρες συλλέγουν δεδομένα σχετικά με τις συνήθειες και τα πρότυπα χρήσης. Οι συσκευές ελέγχου φωνής καταγράφουν πραγματικά αυτά που τους λέτε και στη συνέχεια αποθηκεύουν αυτές τις εγγραφές στο σύννεφο. Όλα αυτά τα δεδομένα συλλέγονται για να διευκολύνουν τη λεγόμενη μηχανική μάθηση. Η μηχανική μάθηση είναι ένας τύπος τεχνητής νοημοσύνης που βοηθά τους υπολογιστές να «μαθαίνουν» χωρίς να χρειάζεται να προγραμματιστούν από ένα άτομο. Αυτοί οι υπολογιστές προγραμματίζονται κατά τρόπο που επικεντρώνεται στα δεδομένα που λαμβάνουν. Αυτά τα νέα δεδομένα μπορούν στη συνέχεια να βοηθήσουν το μηχάνημα να «μάθει» ποιες είναι οι προτιμήσεις σας και να προσαρμόσει ανάλογα.

- **Routers will become more secure and “smarter”**
  Δεδομένου ότι η πλειοψηφία αυτών των συσκευών χρησιμοποιούνται στο σπίτι και δεν μπορούν να έχει εγκατασταθεί λογισμικό ασφαλείας σε αυτά, είναι πολύ ευάλωτα σε επιθέσεις. Με την ανάπτυξη του IoT στην καταναλωτική αγορά, πολλοί κατασκευαστές εργάζονται για να προωθήσουν γρήγορα το προϊόν τους στην αγορά, έτσι μερικές φορές η ασφάλεια μπορεί να αγνοηθεί. Αυτό είναι παίζει πολύ σημαντικό ρόλο. Ο δρομολογητής είναι ουσιαστικά το σημείο εισόδου του Διαδικτύου στο σπίτι. Ενώ οι συνδεδεμένες συσκευές δεν μπορούν να προστατευθούν από μόνες τους , ο δρομολογητής έχει τη δυνατότητα να παρέχει προστασία στο σημείο εισόδου.

  Αν και ο σημερινός τυπικός δρομολογητής παρέχει κάποια πρόσθετη ασφάλεια (όπως προστασία με κωδικό πρόσβασης, τείχη προστασίας και δυνατότητα ρύθμισης των ρυθμίσεων ώστε να επιτρέπονται μόνο ορισμένες συσκευές στο δίκτυό σας) , δεν συνοδεύεται από εγκατεστημένο λογισμικό ασφαλείας. Αυτό σημαίνει ότι το κακόβουλο λογισμικό μπορεί ακόμα να τον στιγματήσει . Με τη δημοτικότητα των συσκευών διαδικτύου και τις υψηλές ευπάθειες που μεταφέρουν, οι επιτιθέμενοι επικεντρώνονται ήδη σε τρόπους εκμετάλλευσης τους.

Η ασφάλεια όμως στο διαδίκτυο δεν επιτυγχάνεται χωρίς περιορισμούς στους βασικούς (τουλάχιστον) χρήστες [16]. Και πάλι, η έκθεση τονίζει ότι το Διαδίκτυο πρέπει να παραμείνει ασφαλές και εύκολο στη χρήση. Οι χρήστες διαμαρτύρονται σήμερα για ορισμένα βασικά μέτρα ασφαλείας, όπως η επαλήθευση ταυτότητας δύο παραγόντων, οπότε είναι λογικό να αναρωτηθούμε αν τα μελλοντικά μέτρα ασφαλείας θα αποθαρρύνουν τους χρήστες. "Υπάρχει πολλή συζήτηση γύρω από την ασφάλεια και την κρυπτογράφηση, αλλά οι χρήστες δεν είναι πρόθυμοι να χρησιμοποιήσουν κάτι που είναι ακόμη ελαφρώς ενοχλητικό. Υποψιάζομαι ότι σε πέντε χρόνια θα συζητήσουμε ακόμα πόσο σημαντική είναι η ασφάλεια και τα πράγματα θα είναι ακόμα πιο ανασφάλιστα ", δήλωσε ένας συμμετέχων στη μελέτη.

Ο Bommelaer de Leusse λέει ότι δεν χρειάζεται να συμβεί εάν οι οργανώσεις επικεντρωθούν στην ευαισθητοποίηση σχετικά με την ασφάλεια και να υιοθετήσουν μια προσέγγιση εστιασμένη στη λύση. "Η μεγαλύτερη ανησυχία που βλέπουμε σήμερα είναι η έλλειψη ασφάλειας, η οποία μεταφράζεται σε υπονόμευση της εμπιστοσύνης στο Διαδίκτυο. Οι χρήστες και οι επιχειρήσεις πρέπει να αισθάνονται σίγουροι ότι η ακεραιότητα των δεδομένων τους προστατεύεται και η τάση αύξησης των επιθέσεων κατά του δικτύου και των υπηρεσιών του ενδέχεται να υπονομεύσει αυτή την εμπιστοσύνη. Το κλειδί εδώ είναι η διαχείριση των κινδύνων και η ελαχιστοποίηση των κινδύνων μέσω καλύτερων πρακτικών ασφαλείας και παράλληλα η προσπάθεια βελτιστοποίησης των οφελών που είναι εγγενείς στο ανοικτό και παγκόσμιο χαρακτήρα του Διαδικτύου. Η συνεργατική ασφάλεια είναι το κλειδί για την αποτελεσματική ελαχιστοποίηση αυτού του κινδύνου. "

Πως θα επηρεάσουν όμως όλα τα παραπάνω τα πρωτόκολλα του Internet?

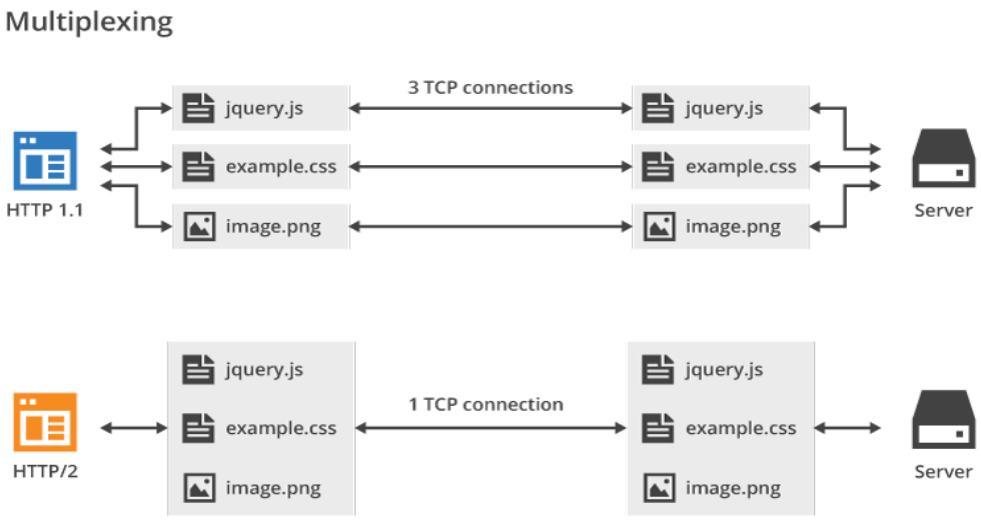
Η ανάλυση αυτή δεν είναι δυνατόν να παραμείνει σε γενικά επίπεδα για αυτό θα την εξηγήσουμε προσδιορίζοντας την εξέλιση ήδη υπάρχοντων πρωτοκόλλων. [17]

- **HTTP/2**
  Το HTTP/2 (βασισμένο στο Google SPDY) ήταν ή πρώτη μεγάλη αλλαγή του πρωτοκόλλου — έγινε το βασικό μέτρο σύγκρισης το 2015. Περιπλέκει πολλαπλά αιτήματα σε μία TCP σύνδεση, αποφεύγοντας την ανάγκη εναπόθεσης σε ουρά τών αιτημάτων τών πελατών χωρίς το ένα να εμποδίζει το άλλο. Τώρα χρησιμοποιείται ευρέως και υποστηρίζεται από όλους τους δημοφιλείς περιηγητές και web servers.

Από την οπτική τών δικτύων το HTTP/2 έφερε σημαντικές αλλαγές.Πρώτων είναι δυαδικό πρωτόκολλο και έτσι οποιαδήποτε συσκευή δεν το αναγνωρίσει δεν θα μπορέσει να λειτουργήσει Αυτός είναι και ένας από τους κύριους λόγους για τους οποίους έπρεπε να αναβθμιστεί το HTTP/2 καθώς απαιτεί κρυπτογράφηση. Αυτό το προστατεύει καλύτερα από παρεμβολές από μεσάζοντες που υποθέτουν ότι είναι HTTP/1.1 .

Το HTTP/2 επίσης απαιτεί την χρήση του TLS/1.2 όταν είναι κρυπτογραφημένο, και απομακρύνει μεθόδους αποκρυπτογράφησης οι οποίες έχουν οριστεί ώς μη ασφαλείς επιτρέποντας μόνο την χρήση κλειδιών. Τέλος το HTTP/2 επιτρέπει περισσότερα από αιτήματα του ενός host να ενοποιηθούν σε μία σύνδεση, προκειμένου να αυξήσει την απόδοση, μειώνοντας το πλήθος τών συνδέσεων . Πέραν αυτών τών αλλαγών αξίζει να τονίσουμε ότι το HTTP/2 δεν φαίνεται να αντιμετοπίζει σοβαρά προβλήματα διαλειτουργικότητας ή παρεμβολές μέσα σε οποιοδήποτε δίκτυο.

Στην παρακάτω εικόνα μπορούμε να διακρίνουμε τις πιο αξιοσημείωτες διαφορές που δημιουργήθηκαν κατά την εξέλιξη του πρωτοκόλλου HTTP. Όπως ήδη αναφέραμε, το HTTP/2 κατάφερε να μειώσει σημαντικό το πλήθος τών TCP συνδέσεων που απαιτούνται.



Εικόνα 6.2 : Διαφορές στα HTTP 1.1 και HTTP/2

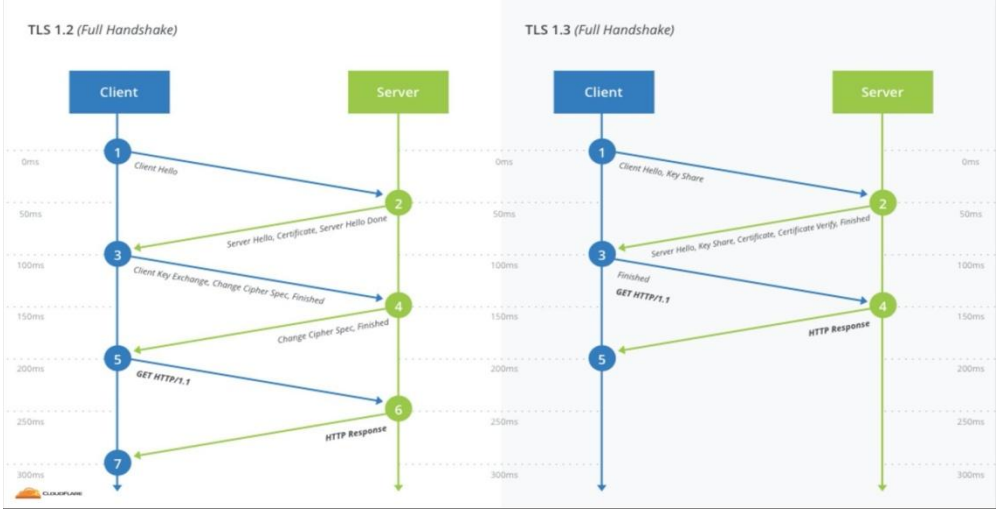
- **TLS 1.3**

  Το TLS 1.3 βρίσκεται ήδη στα τελευταία στάδια της τελειοποίησης του και ήδη υποστηρίζεται από πολλούς φυλλομετρητές και servers . Δεν είναι μια απλή αναθεώρηση παρά μια πλήρης νέα έκδοση του TLS, με καλύτερα πρωτόκολλα σύναψης , τα οποία επιτρέπουν τα δεδομένα εφαρμογών να κατευθύνονται από την έναρξη (0RTT) προς τον προορισμό τους. Ο νέος σχεδιασμο΄ς βασίζεται στην ανταλλαγή εφήμερων κλειδιών , απομακρύνοντας πλέων τα στατικά κλειδιά. Αυτό έχει προκαλέσει ανησυχία σε ορισμένους φορείς εκμετάλλευσης δικτύων και πωλητές - ιδίως εκείνους που χρειάζονται ορατότητα σε αυτό που συμβαίνει μέσα σε αυτές τις συνδέσεις.

  Για παράδειγμα, εξετάστε το κέντρο δεδομένων για μια τράπεζα που έχει ρυθμιστικές απαιτήσεις για προβολή. Με την παρεμπόδιση της κυκλοφορίας στο δίκτυο και την αποκρυπτογράφηση με τα στατικά κλειδιά των διακομιστών τους, μπορούν να καταγράψουν τη νόμιμη κυκλοφορία και να εντοπίσουν την επιβλαβή κίνηση, είτε πρόκειται για εισβολείς από το εξωτερικό είτε για υπαλλήλους που προσπαθούν να διαρρεύσουν δεδομένα από μέσα.

  Το TLS 1.3 δεν υποστηρίζει αυτή τη συγκεκριμένη τεχνική για την παρεμπόδιση της κυκλοφορίας, καθώς είναι επίσης μια μορφή επίθεσης που προστατεύουν τα εφήμερα κλειδιά. Ωστόσο, δεδομένου ότι διαθέτουν κανονιστικές απαιτήσεις τόσο για τη χρήση σύγχρονων πρωτοκόλλων κρυπτογράφησης όσο και για την παρακολούθηση των δικτύων τους, αυτό θέτει αυτούς τους φορείς εκμετάλλευσης δικτύου σε μια δύσκολη θέση.

  Υπήρξε μεγάλη συζήτηση σχετικά με το εάν οι κανονισμοί απαιτούν στατικά κλειδιά, εάν οι εναλλακτικές προσεγγίσεις θα μπορούσαν να είναι εξίσου αποτελεσματικές και εάν η εξασθένηση της ασφάλειας για ολόκληρο το Διαδίκτυο προς όφελος σχετικά μικρών δικτύων είναι η σωστή λύση. Πράγματι, εξακολουθεί να είναι δυνατή η αποκρυπτογράφηση της επισκεψιμότητας στο TLS 1.3, αλλά χρειάζεστε πρόσβαση στα εφήμερα κλειδιά για να το κάνετε αυτό και από τη σχεδίαση δεν είναι μακροχρόνια.Σε αυτό το σημείο δεν φαίνεται ότι το TLS 1.3 θα αλλάξει για να φιλοξενήσει αυτά τα δίκτυα, αλλά υπάρχουν περιπλανήσεις για τη δημιουργία ενός άλλου πρωτοκόλλου που επιτρέπει σε ένα τρίτο μέρος να παρατηρήσει τι συμβαίνει - και ίσως περισσότερο - για αυτές τις περιπτώσεις χρήσης. Το αν αυτό θα γίνει πραγματικότητα θα φανεί στην πορεία.



Εικόνα 6.3 : TLS 1.3

Έτσι γίνεται πλήρως κατανοητό ότι δεν υπάρχει καμία σταθερότητα στα πρωτόκολλα που χρησιμοποιούνται όσο τα δίκτυα σταοποία εφαρμόζονται δεν παραμένουν σταθερά. Και αυτό είναι λογικό. Οι ανάγκες για επικοινωνία πολλαπλασιάζονται και για να προοδεύσουμε πρέπει να συμβαδίσουμε, έτσι και τα πρωτόκολλα ακολουθούν αυτή την πορεία , συμβάλλοντας έτσι σε ένα πιο ασφαλές (δια)δίκτυο.

Βιβλιογραφία

[1] : Licesio J. Rodríguez-Aragón: Tema 4: Internet y Teleinformática

[2] : https://www.infoq.com/news/2014/07/protocol-design-sbe-thompson

[3] : https://en.wikipedia.org/wiki/Internet_protocol_suite

[4] : https://www.juniper.net/documentation

[5] : Λάζαρος Μεράκος – Καθηγητής Τμήματος Πληροφορικής και Τηλεπικοινωνιών ΕΚΠΑ

[6] : A. Tannenbaum – Computer Networks

[7] : https://en.wikipedia.org/wiki/Reliability_(computer_networking)

[8] : http://searchnetworking.techtarget.com/answer/How-can-we-say-TCP-is-a-reliable-protocol

[9] : http://protocols.netlab.uky.edu/~calvert/classes/571/lectureslides/ErrorDetection.pdf

[10] : Τηλεπικοινωνίες και Δίκτυα Υπολογιστών – Άρης Αλεξόπουλος ,Γιώργος Λαγογιάννης

[11] : Διαδίκτυα με TCP/IP Αρχές, Πρωτόκολλα και Αρχιτεκτονικές 4th Edition – Douglas E. Comer

[12] : http://searchnetworking.techtarget.com/definition/modem-error-correcting-protocols

[13] : A. J. McAuley, Reliable Broadband Communication Using a Burst Erasure Correcting Code, ACM SIGCOMM, 1990.

[14] : https://www.quora.com/What-is-the-future-of-internet-security

[15] : https://us.norton.com/internetsecurity-iot-5-predictions-for-the-future-of-iot.html

[16] : https://www.csoonline.com/article/3226392/security/future-cyber-security-threats-and-challenges-are-you-ready-for-whats-coming.html

[17] : https://blog.apnic.net/2017/12/12/internet-protocols-changing/

[18] : https://www.cso.com.au/article/573522/how-internet-things-reshaping-future-security/

[19] : https://www.theinquirer.net/inquirer/feature/2433753/the-future-of-internet-security

[20] : https://www.cisco.com/c/en/us/products/security/what-is-network-security.html

[21] : https://en.wikipedia.org/wiki/Network_security

[22] : https://technet.microsoft.com/en-us/library/cc776369(v=ws.10).aspx

[23] : https://technet.microsoft.com/en-us/library/cc179879.aspx

[24] : http://searchmidmarketsecurity.techtarget.com/definition/IPsec

[25] : https://mayaposch.wordpress.com/2011/10/03/design-your-own-protocol-in-five-minutes/

[26] : http://www.cs.cornell.edu/courses/cs5430/2017sp/l/11-protocols/lec.pdf