ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ευφυής Αυτοματοποιημένη Διαχείριση και Προστασία Συσκευών (Machine to Machine Management)

Του φοιτητή

Γραμμενίδη Νικολάου

Αρ. Μητρώου: 154441

Επιβλέπων καθηγητής

Ηλιούδης Χρήστος

Θεσσαλονίκη 2020

ΠΕΡΙΛΗΨΗ

Στις μέρες μας, τα δίκτυα υπολογιστών είναι πολύ πιο περίπλοκα σε σχέση με όταν δημιουργήθηκε το διαδίκτυο. Με το πέρασμα του χρόνου και την ραγδαία ανάπτυξη της τεχνολογίας, όλο και περισσότερες συσκευές συνδέονται στο διαδίκτυο. Για να είναι ένα δίκτυο λειτουργικό και αποτελεσματικό, βασική προϋπόθεση αποτελεί η εγκατάσταση λογισμικών παρακολούθησης και συντήρησης αυτού, αλλά και η ορθή παραμετροποίησή τους. Τα σύγχρονα επιχειρησιακά δίκτυα, χρησιμοποιούν συσκευές και λογισμικά από διαφορετικούς κατασκευαστές. Αυτό φέρει σαν αποτέλεσμα, να μην υπάρχει συγκεντρωτική ενημέρωση από όλα τα συστατικά του δικτύου, από το λογισμικό παρακολούθησης και συντήρησης αυτού, προς τον διαχειριστή του. Η λύση στο πρόβλημα αυτό, είναι η χρήση του Open Command and Control προτύπου, το οποίο καταφέρνει με τον κατάλληλο προγραμματισμό, την ορθή επικοινωνία και την διαλειτουργικότητα μεταξύ των συστατικών ενός δικτύου.

ABSTRACT

Since the advent of the Internet, there have been tremendous changes in the design and operation of computer networks. The rapid progress of various interconnected technologies has resulted in unprecedented numbers of devices being connected to the internet, as well as other computer networks. If a network is to be considered fully functional and operational, a necessary prerequisite is the installation of maintenance and monitoring software, as well as their proper configuration. Modern business networks utilize a variety of different systems from different vendors. As an immediate result, a network administrator is unable to effectively control and monitor those very disparate systems that have difficulty communicating with each other. The solution to this problem is provided by the Open Command and Control standard, which, when correctly configured, ensures interconnectivity between the various systems.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω εκ βάθους καρδίας τους φίλους μου, για την πολυετή ηθική και υλική στήριξη και συμπαράσταση. Είναι πάντα δίπλα μου ακόμα και όταν κάνω λάθη.

Περιεχόμενα

Ευρετήριο εικόνων

Ευρετήριο Πινάκων

ΚΕΦΑΛΑΙΟ 1

1.1 Περιοχή έρευνας

Στις μέρες μας, το διαδίκτυο είναι ευρέως διαδεδομένο. Οι περισσότεροι άνθρωποι το χρησιμοποιούν είτε για επαγγελματικούς, είτε για προσωπικούς λόγους αρκετές ώρες την ημέρα. Οι περισσότερες επιχειρήσεις στηρίζονται και αναπτύσσονται μέσω αυτού, είτε μέσω διαφημίσεων, είτε έχουν το πληροφοριακό τους σύστημα στο νέφος (cloud). Όλο και περισσότερες συσκευές, εξαιτίας της διαρκούς εξοικείωσης των ανθρώπων με την τεχνολογία, συνδέονται στο διαδίκτυο.

Το διαδίκτυο δεν είναι κάτι άλλο, πέρα από ένα σύνολο δικτύων τα οποία επικοινωνούν μεταξύ τους. Όλα αυτά τα δίκτυα χρησιμοποιούν βασικούς κανόνες για να επικοινωνούν ορθά το ένα με το άλλο. Δεν αρκεί όμως μόνο αυτό, πρέπει να υπολογίζονται και άλλοι παράγοντες. Είναι γεγονός, πως για να υποστηρίζονται όλες αυτές οι συσκευές που συνδέονται στο διαδίκτυο, πρέπει τα δίκτυα που το αποτελούν να είναι έμπιστα, όσον αφορά την προώθηση των πακέτων. Είναι σημαντικό να τηρούνται οι βασικοί κανόνες επικοινωνίας ή όπως αλλιώς ονομάζονται, σουίτα πρωτοκόλλων.

Υπάρχει όμως ένα σημαντικό πρόβλημα. Όσες περισσότερες συσκευές συνδέονται σε ένα δίκτυο, τόσο περισσότερο αυξάνεται η πιθανότητα να υπάρξει συμφόρηση σε αυτό. Για αυτό το λόγο πρέπει να υπάρχει μία στοιχειώδης επιτήρηση στις δικτυακές συσκευές. Είναι απαραίτητο η κατάσταση των κόμβων και των δικτυακών συσκευών, όπως άλλωστε και των μηχανισμών ασφαλείας, να βρίσκονται σε παρακολούθηση πραγματικού χρόνου. Σε περίπτωση που κάποιος δρομολογητής (router) κλείσει ή ακόμα και χαλάσει, να μπορεί να το γνωρίζει ο υπεύθυνος του δικτύου, με σκοπό να αναπληρώσει την βλάβη. Φυσικά, σε ένα επαγγελματικό δικτυακό περιβάλλον υπάρχουν οι κατάλληλες παραμετροποιήσεις και η βλάβη αντικαθίσταται αυτοματοποιημένα, αφού υπάρχουν δρομολογητές σε ετοιμότητα, σε περίπτωση που συμβεί κάποια βλάβη.

Την πρόκληση της συνεχούς παρακολούθησης των συσκευών και των συστατικών ενός δικτύου, έχουν να αντιμετωπίσουν τα συστήματα παρακολούθησης δικτυακών συσκευών. Με τον κατάλληλο προγραμματισμό, η διαχείριση, η επιτήρηση και η αναβάθμιση των συσκευών και των λογισμικών τους επιτυγχάνεται με μεγάλη άνεση από την μεριά του διαχειριστή του δικτύου. Δεν είναι όμως εύκολο να επιτευχθεί αυτός ο στόχος. Δυστυχώς, υπάρχουν αρκετές

ασυμβατότητες μεταξύ των λογισμικών που χρησιμοποιούν οι δικτυακές συσκευές, επομένως είναι δυσκολότερη η επίτευξη του παραπάνω στόχου.

Όλες οι παραπάνω πληροφορίες αναφέρονται στον τρόπο εποπτείας του δικτύου, όσον αφορά τη λειτουργικότητά του και το πόσο αποδοτικό είναι. Από μεριά ασφάλειας, ένα δίκτυο πρέπει να διαθέτει του κατάλληλους μηχανισμούς. Στα περισσότερα δίκτυα, είτε οικιακά είτε επαγγελματικά, συνδέονται όλο και περισσότερες συσκευές. Με την ραγδαία εξέλιξη του διαδικτύου των πραγμάτων (Internet of Things), εισάγονται συσκευές, οι οποίες είναι ευάλωτες σε επιθέσεις κυβερνοασφάλειας. Είναι λογικό, να σκεφτεί κανείς, ότι εφόσον είναι ένας καινούριος τομέας του διαδικτύου, δεν είναι ακόμη σε θέση να διαθέτει τα κατάλληλα συστατικά δικτύου για να αντιμετωπίσει κάποιο είδος επίθεσης του κυβερνοχώρου, επομένως οι μόνοι μηχανισμοί ασφαλείας που μπορούν να χρησιμοποιηθούν, είναι οι ήδη υπάρχοντες.

1.2 Στόχοι που τέθηκαν στη διπλωματική

Η διπλωματική εργασία στοχεύει στην πλήρη κατανόηση και μελέτη των υφιστάμενων προτύπων διαχείρισης και ελέγχου των συσκευών και των υπηρεσιών. Ο κυριότερος στόχος είναι η μελέτη και η ανάπτυξη ενός συστήματος διαχείρισης και προστασίας συσκευών και υπηρεσιών στηριζόμενο στο Open Command and Control (OpenC2) πρότυπο. Θα πραγματοποιηθεί μελέτη των υφισταμένων προτύπων διαχείρισης και ελέγχου συσκευών και υπηρεσιών. Επιπρόσθετα, στοχεύει στην αναλυτική καταγραφή των χαρακτηριστικών του Open Command and Control (OpenC2) προτύπου. Τέλος, πέραν της παρουσίασης των υλοποιήσεων του OpenC2 σε μηχανισμούς ασφαλείας, η διπλωματική στοχεύει και στην πειραματική εφαρμογή ενσωμάτωσης του OpenC2 σε συστήματα self-healing.

1.3 Επιτεύγματα της διπλωματικής

Η διπλωματική εργασία εμβαθύνει σε μεγάλο βαθμό στην κατανόηση και μελέτη των υφιστάμενων προτύπων διαχείρισης και ελέγχου των συσκευών και των υπηρεσιών. Αναλύει και περιγράφει τα υφιστάμενα πρότυπα διαχείρισης και ελέγχου συσκευών και υπηρεσιών. Συγκρίνει τις ανάγκες των δικτύων που χρησιμοποιούνται σήμερα, σε σχέση κατά πόσο καλύπτονται από τα πρότυπα αυτά.

Η διπλωματική πέραν της ανάλυσης των προτύπων διαχείρισης και ελέγχου συσκευών και υπηρεσιών, καταγράφει αναλυτικά τα χαρακτηριστικά του Open Command and Control (OpenC2) προτύπου.

Επιπροσθέτως, πραγματοποιείται συγκριτική παρουσίαση των υλοποιήσεων του OpenC2 σε μηχανισμούς ασφαλείας. Τέλος, εκτελεί μια μεγάλη περίπτωση χρήσης (case study), όπου στήνει έναν μηχανισμό self-healing από την αρχή και τον ενσωματώνει με το OpenC2 πρότυπο, με σκοπό να επιτευχθεί η διαδικασία της ίασης.

1.4 Διάρθρωση της διπλωματικής

Το δεύτερο κεφάλαιο της διπλωματικής μελετά την αρχιτεκτονική και τις βασικές λειτουργίες των συστημάτων διαχείρισης και ελέγχου των συσκευών και των υπηρεσιών. Το τρίτο κεφάλαιο αναλύει τα υπάρχοντα πρότυπα διαχείρισης και ελέγχου των συσκευών και των υπηρεσιών ενός δικτύου. Το τέταρτο κεφάλαιο περιγράφει αναλύει τα χαρακτηριστικά του προτύπου Open Command and Control (OpenC2), καθώς και συγκρίνει τις υλοποιήσεις του σε μηχανισμούς ασφαλείας. Το πέμπτο κεφάλαιο περιλαμβάνει ένα case study, όπου στήνεται ένας μηχανισμός self-healing από την αρχή και ενσωματώνεται με το OpenC2 πρότυπο. Στο έκτο κεφάλαιο αναφέρονται μελλοντικές επεκτάσεις που θα προσέδιδαν ακόμη περισσότερο στην αποδοτικότητα του OpenC2.

ΚΕΦΑΛΑΙΟ 2

Διαχείριση και Έλεγχος Συσκευών και Υπηρεσιών

ΕΙΣΑΓΩΓΗ

Πολλοί πιστεύουν ότι ένα δίκτυο δεν χρήζει από κάποια παρακολούθηση. Εφόσον το δίκτυο είναι λειτουργικό, έχουν λανθασμένα την εντύπωση ότι δεν θα προκύψει κάποιο πρόβλημα. Σχεδόν κανένα όμως δίκτυο στις μέρες μας, δεν είναι στατικό. Όλο και περισσότερο, εισάγεται στη ζωή μας η τεχνολογία και εμείς με τον τρόπο μας φέρνουμε καινούριες συσκευές σε αυτό. Εφόσον το δίκτυο πρέπει να ικανοποιήσει μεγαλύτερες ανάγκες, όλο και πιο συχνά θα εμφανίζονται προβλήματα. Αυτή η εικόνα, ότι ένα δίκτυο είναι και θα παραμείνει σταθερό, διαστρεβλώνεται όταν τα πακέτα αρχίσουν να χάνονται όλο και πιο συχνά ή έως ότου δεχτεί κάποια επίθεση κυβερνοασφάλειας.

Μία επίθεση κυβερνοασφάλειας, μπορεί να ξεκινήσει από ένα άτομο ή περισσότερα. Οι εκάστοτε επιτιθέμενοι, χρησιμοποιούν τουλάχιστον έναν υπολογιστή ή σύστημα για να επιτύχουν το σκοπό τους. Εάν η επίθεση είναι επιτυχής, υπάρχει μεγάλη πιθανότητα να βρίσκεται σε κίνδυνο τόσο το δίκτυο του θύματος, όσο και τα δεδομένα των χρηστών του δικτύου.

2.1 Συστήματα Διαχείρισης Δικτύου

Τα Συστήματα Διαχείρισης Δικτύου έχουν σχεδιαστεί για να παρακολουθούν την κατάσταση των συσκευών σε ένα δίκτυο. Συνήθως αποτελούνται από μία ή περισσότερες εφαρμογές λογισμικού, ακόμα και από συσκευές υλικού. Επιπροσθέτως, στα καθήκοντα ενός τέτοιου συστήματος εμπεριέχεται η παρακολούθηση της κατάστασης του δικτύου. Μπορεί να παρακολουθεί διάφορες λειτουργίες ανά συσκευή π.χ. εάν η θερμοκρασία σε κάποιον επεξεργαστή είναι πολύ υψηλή (ενώ δεν θα έπρεπε), επειδή κάποιος ανεμιστήρας έχει χαλάσει. Επίσης μπορούν να έχουν μια καθολική εικόνα ενός δικτύου. Στην περίπτωση που κάποια καινούργια συσκευή ανιχνευτεί, τότε ειδοποιείται ο διαχειριστής του δικτύου. Μία από τις σημαντικότερες λειτουργίες τους, είναι η ανάλυση της επίδοσης του δικτύου. Σε περίπτωση που υπάρχει μεγάλη απώλεια πακέτων, τα δεδομένα της

ανάλυσης του δικτύου μπορούν να χρησιμοποιηθούν ώστε να βελτιστοποιηθεί το δίκτυο.

Το σύστημα αυτό περιέχει μία διεπαφή για τον διευθυντή του δικτύου. Ο διευθυντής του δικτύου μπορεί να παραμετροποιήσει το λογισμικό ώστε μόλις λαμβάνει χώρα κάποιο συμβάν, τότε να ειδοποιείται. Επίσης μπορεί να αυτοματοποιήσει κάποιες παραμετροποιήσεις για καλύτερη χρήση του δικτύου. Σε περίπτωση που κάποιος από τους μεταγωγείς (switches) κλείσει, τότε μπορεί να στέλνει αυτόματα αυτόματα εντολή να ανοίξει κάποιος άλλος μεταγωγέας για να συνεχίσει το δίκτυο να έχει την ομαλή λειτουργία του.

Είναι σημαντική η χρήση ενός τέτοιου συστήματος για την κεντρικό-ποιημένη παρακολούθηση ενός δικτύου, με σκοπό την διαχείριση, την αναβάθμιση λογισμικών και αντικατάσταση των συσκευών. Ουσιαστικά, το λογισμικό διαχείρισης του δικτύου είναι υπεύθυνο για την τοποθέτηση και λήψη παγίδων από τους ατζέντες. Μία παγίδα δικτύου είναι μία λογική συνθήκη, η οποία μόλις ικανοποιηθεί, τότε πραγματοποιούνται όλα αυτά που έχει καθορίσει το σύστημα διαχείρισης του δικτύου.

2.1.1 Αρχιτεκτονική NMS

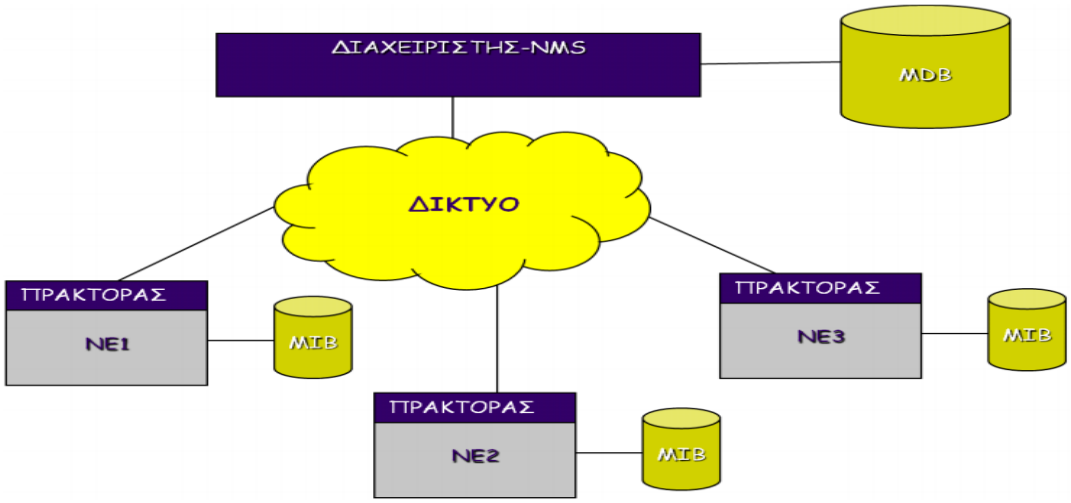
Οι δύο πιο βασικοί παράγοντες που [23] πρέπει να ακολουθήσει κανείς για να έχει την καταλληλότερη αρχιτεκτονική δικτύου, όσον αφορά την λειτουργικότητα του αυτού, είναι η πολυπλοκότητα της διαχείρισής του αλλά και της σύνθεσής του. Όσον αφορά την διαχείριση του δικτύου, οι αρχιτεκτονικές που μπορεί να ακολουθήσει από πάνω προς τα κάτω (top-down) επίπεδα είναι τρεις [25]:

- Κεντρική Αρχιτεκτονική Διαχείρισης (Centralized Network Management Architecture)
- Ιεραρχική Αρχιτεκτονική Διαχείρισης (Hierarchical Network Management Architecture)
- Κατανεμημένη Αρχιτεκτονική Διαχείρισης (Distributed Network Management Architecture)

Οποιαδήποτε από τις παραπάνω αρχιτεκτονικές έχει τα υπέρ και τα κατά της. Θα τις δούμε πιο αναλυτικά παρακάτω.

2.1.1.1 Κεντρική Αρχιτεκτονική Διαχείριση (Centralized Network Management Architecture)

Μία από τις βασικές κατηγορίες αρχιτεκτονικών ομότιμων δικτύων είναι η Κεντρική Αρχιτεκτονική Διαχείριση Δικτύων. Σε αυτήν, υπάρχει ένας κεντρικός σταθμός διαχείρισης του δικτύου. Μπορεί επίσης να υπάρχουν και κάποιοι εφεδρικοί. Η πλατφόρμα διαχείρισης αναλαμβάνει καθολικά τον ρόλο διαχείρισης του δικτύου και την αποθήκευση των πληροφοριών διαχείρισης του δικτύου. Είναι σημαντικό να αναφερθεί, πως οι πληροφορίες δεν είναι απαραίτητο να αποθηκεύονται στον σταθμό διαχείρισης μεμονωμένα. Υπάρχει η δυνατότητα να αποθηκεύονται αποκεντρωμένα σε διάφορα μηχανήματα, όμως ο έλεγχος πρέπει πάντα να γίνεται από τον κεντρικό σταθμό.



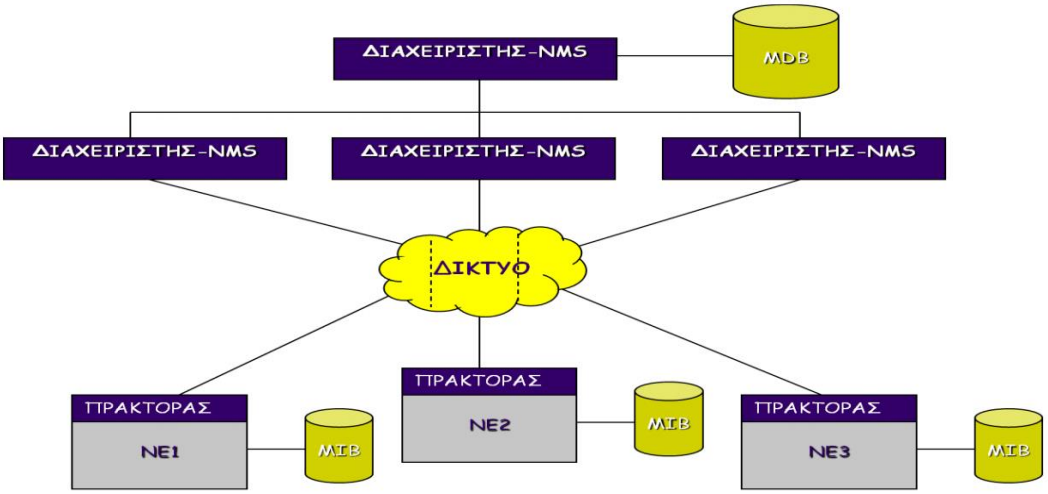
Εικόνα 1 "Διάγραμμα Κεντρικής Αρχιτεκτονικής Διαχείρισης "

Στην παραπάνω εικόνα απεικονίζεται ο Διαχειριστής-NMS , ο οποίος είναι υπεύθυνος για τον έλεγχο των πληροφοριών. Επίσης βλέπουμε τους πράκτορες που τρέχουν πάνω σε στοιχεία δικτύου. Η αποθήκευση των πληροφοριών γίνεται κεντρικό-ποιημένα και εμφανίζεται στην εικόνα ως Management Data Base. Κάθε πράκτορας διαθέτει μοναδικό αναγνωριστικό το οποίο αποθηκεύεται στο Management Information Base.

2.1.1.2 Ιεραρχική Αρχιτεκτονική Διαχείρισης (Hierarchical Network Management Architecture)

Στην Ιεραρχική Αρχιτεκτονική Διαχείρισης μη ομότιμων δικτύων υπάρχει ένας διαχειριστής στον κεντρικό σταθμό του δικτύου, αλλά δεν είναι το μόνο λογισμικό διαχείρισης που εκτελείται. Η κύρια διαφορά είναι ότι βρίσκεται υψηλότερα στην ιεραρχία μεταξύ των υπόλοιπων λογισμικών. Ο κεντρικός σταθμός διαχείρισης είναι υπεύθυνος για την ορθή λειτουργία και για τον συντονισμό των λειτουργιών των επιμέρους διαχειριστών. Οι πληροφορίες του δικτύου βρίσκονται συγκεντρωτικά στο σύστημα διαχείρισης βάσεων δεδομένων του κεντρικού σταθμού.

Ένα σημαντικό μέτρο που θα μπορούσε να παρθεί, με σκοπό την αύξηση της ασφάλειας του δικτύου είναι ο κεντρικός σταθμός να επικοινωνεί με τους επιμέρους σταθμούς διαχείρισης σε ξεχωριστό δίκτυο από αυτό που βρίσκεται το υπόλοιπο δίκτυο. Έτσι, σε περίπτωση παραβίασης του δικτύου ή ακόμα και σε περίπτωση που βρίσκεται κάποιος κακόβουλος χρήστης και παρακολουθεί μη εξουσιοδοτημένα το δίκτυο, να μην μπορεί να παραλάβει αυτές τις σημαντικές πληροφορίες μεταξύ των σταθμών.

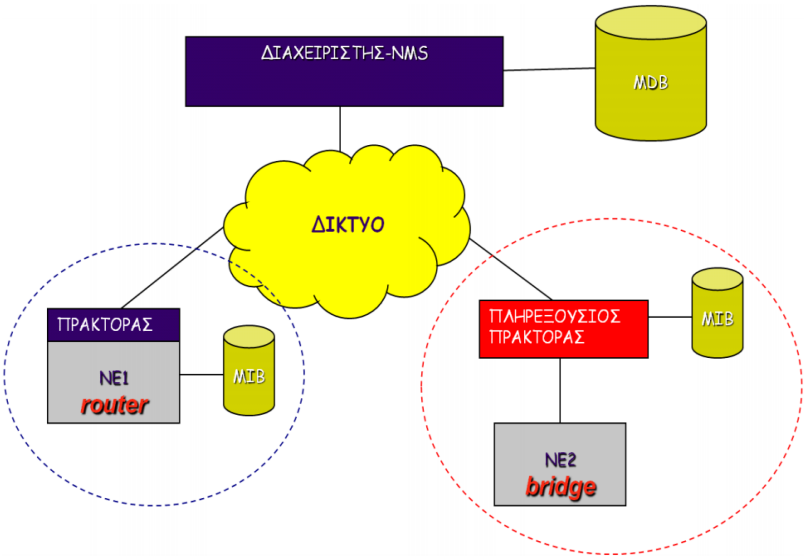


Εικόνα 2 " Ιεραρχική Αρχιτεκτονική Διαχείρισης "

2.1.1.3 Κατανεμημένη Αρχιτεκτονική Διαχείριση (Distributed Network Management Architecture)

Παρατηρείται πως αυτή η αρχιτεκτονική έχει πολλά κοινά χαρακτηριστικά με την λογική που ακολουθούν τα ομότιμα δίκτυα. Πιο αναλυτικά, ένα λογισμικό είναι επικεφαλής για κάποια ομότιμα δίκτυα, με την σημαντική διαφορά ότι τα διαχειριστικά καθήκοντα επιμερίζονται από την επικεφαλή πλατφόρμα διαχείρισης στους τοπικούς διαχειριστές. Επιπροσθέτως, οι πληροφορίες είναι δυνατόν να διαμοιραστούν στο τοπικό σύστημα διαχείρισης βάσεων δεδομένων.

Είναι γεγονός ότι οι σταθμοί διαχείρισης ανταλλάσσουν τις πληροφορίες που έχουν αποθηκεύσει στην τοπική βάση τους, σε περίπτωση που τους ζητηθεί. Εάν κάποιος σταθμός χρειάζεται πληροφορίες για κάποιον πράκτορα που βρίσκεται σε διαφορετικό τοπικό δίκτυο, αρκεί να αιτηθεί τις κατάλληλες πληροφορίες από τον αρμόδιο σταθμό διαχείρισης. Αυτή η κατανομή των αρμοδιοτήτων, επιφέρει και τις ανάλογες αποδόσεις στους σταθμούς. Οι σταθμοί χρειάζονται πολύ λιγότερο υπολογιστική ισχύ από έναν κεντρικό σταθμό σε οποιαδήποτε άλλη αρχιτεκτονική. Συνοψίζοντας, μπορούν να πραγματοποιηθούν πολύ πιο εύκολα αλλαγές στο δίκτυο, να αυξηθεί η αποδοτικότητα αλλά και σε περίπτωση απώλειας κάποιου σταθμού, να αντικατασταθεί πιο εύκολα χωρίς υψηλό κόστος για το δίκτυο.

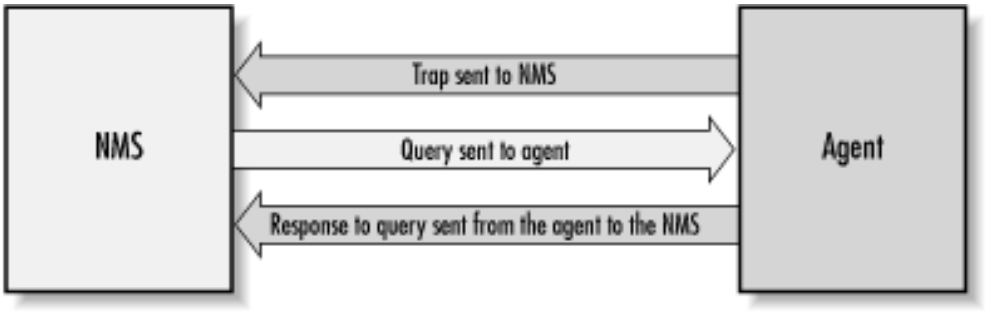


Εικόνα 3 "Κατανεμημένη Αρχιτεκτονική Διαχείρηση"

2.1.2 Βασικές Λειτουργίες

Μία από τις βασικές λειτουργίες των Συστημάτων Διαχείρισης Δικτύων είναι η δυνατότητα ανακάλυψης συσκευών. Τα συστήματα αυτά αναγνωρίζουν τις συσκευές που είναι ενεργές σε ένα δίκτυο. Επίσης παρακολουθούν τις συσκευές και την κατάστασή τους στο δίκτυο. Ουσιαστικά παρακολουθούν την «υγεία» των συσκευών, όπως και την επίδοσή τους στο δίκτυο αλλά και την επίδρασή τους σε αυτό. Είναι αντιληπτό, ότι τα Συστήματα Διαχείρισης Δικτύων παρακολουθούν και αναλύουν και την επίδοση του δικτύου μέσω διαφόρων δεικτών, όπως το bandwidth, την απώλεια πακέτων αλλά και πιο άμεσα, όπως επίσης και παρακολουθώντας τον χρόνο που είναι ενεργός ο δρομολογητής, οι μεταγωγείς αλλά και άλλες συσκευές που έχουν καθοριστεί να αλληλοεπιδρούν.

Παρακάτω θα αναλυθεί η βασική επικοινωνία μεταξύ ενός Συστήματος Διαχείρισης Δικτύου με έναν πράκτορα. Στο σχεδιάγραμμα μπορούμε να δούμε ότι ο πράκτορας στέλνει στο NMS (Network Management System) πληροφορίες. Την πυροδότηση αυτού του γεγονότος την έχει καθορίσει το NMS . Μόλις ικανοποιηθεί το λογικό γεγονός, από το οποίο το NMS έχει καθορίσει, ο πράκτορας στέλνει τις κατάλληλες πληροφορίες που έχουν προκαθοριστεί στο NMS, στη συνέχεια αυτό αποκρίνεται με ένα αίτημα. Τέλος ο Πράκτορας απαντάει στην απόκριση με τις κατάλληλες πληροφορίες.



Εικόνα 4 "Βασική Επικοινωνία NMS - Πράκτορα"

Είναι σημαντικό να επισημανθεί πως το NMS μπορεί ταυτόχρονα να δεχτεί μία εντολή τύπου Παγίδα (Trap) αλλά και να λάβει πληροφορίες για την συσκευή. Η επαναλαμβανόμενη διαδικασία, όπου ο διαχειριστής ζητάει πληροφορίες για την εκάστοτε συσκευή (πράκτορας), ονομάζεται σφυγμομέτρηση (polling).

2.1.3 Πλεονεκτήματα

Στις μέρες μας, ένα δίκτυο πρέπει να ικανοποιεί αποτελεσματικά τις ανάγκες των χρηστών του. Ολοένα και περισσότερο αυξάνονται οι απαιτήσεις των χρηστών. Ένα δίκτυο πρέπει να στέλνει και να δέχεται πακέτα χωρίς απώλειες αλλά και να γίνεται σωστός διαμοιρασμός των πακέτων. Πλέον, βρισκόμαστε σε μία εποχή όπου ένας απλός χρήστης του διαδικτύου, χρησιμοποιεί το δίκτυο για να δει κάποιο βίντεο, να κάνει κλήση μέσω διαδικτύου κτλ. Όλες αυτές οι ενέργειες πρέπει να ικανοποιούνται από το δίκτυο που χρησιμοποιεί. Αρκετές επιχειρήσεις έχουν BYOD (φέρτε την προσωπική σας συσκευή) πολιτική, η οποία κατ' εξακολούθηση επιβαρύνει περισσότερο το δίκτυο.

Σε μικρά δίκτυα, όπως σε μία μικρή επιχείρηση ή όπως σε μία οικεία, ένα Δικτυακό Σύστημα Διαχείρισης δεν είναι υποχρεωτικό. Παρόλα αυτά, είναι απαραίτητο για ένα «μεγάλο» δίκτυο. Σε περίπτωση που η επιχείρηση διαθέτει εξυπηρετητές, data center, server farm, πολλαπλά δίκτυα επικοινωνίας μεταξύ των εργαζομένων ή και των πελατών (internal – external networks), τότε ένα Δικτυακό Σύστημα Διαχείρισης θα διευκόλυνε πάρα πολύ τον προγραμματισμό και την παρακολούθηση του δικτύου.

Η τοποθέτηση ενός κατάλληλου συστήματος διαχείρισης δικτύου μπορεί να βοηθήσει ριζικά τους διαχειριστές του συστήματος με τα απαραίτητα εργαλεία, για τη φροντίδα και τη συντήρηση του εταιρικού δικτύου. Είναι σημαντικό να επισημανθεί, πως όλα τα παραπάνω μπορούν να πραγματοποιηθούν χωρίς υπερβολική χειροκίνητη παρέμβαση. Ο στόχος είναι, η διαχείριση του σύνθετου δικτύου από το κέντρο δεδομένων, έως την άκρη του δικτύου μέσω ενσωματωμένων ενσύρματων και ασύρματων δυνατοτήτων διαχείρισης, αυτόματων ειδοποιήσεων για ζητήματα δικτύου και συμβατότητας με συσκευές σε όλους τους προμηθευτές. Αυτό εξαρτάται ανάλογα με τον τύπο του δικτύου και τις απαιτήσεις του.

Ένα ακόμα βασικό πλεονέκτημα τοποθέτησης συστήματος διαχείρισης δικτύου είναι το κόστος. Γλυτώνοντας αρκετές ώρες δουλειάς από τον διευθυντή του δικτύου και από την ομάδα που το προγραμματίζει. Παράλληλα αυξάνεται το κέρδος της επιχείρησης, έχοντας ένα πιο λειτουργικό δίκτυο, εφόσον μειώνεται ο χρόνος που το δίκτυο δεν είναι λειτουργικό (ή τουλάχιστον αντιμετωπίζεται πιο άμεσα). Συνοψίζοντας, εφόσον καθοριστεί ότι ένα δίκτυο χρειάζεται ένα σύστημα διαχείρισης

δικτύου, ανάλογα με τις ανάγκες των χρηστών του, τότε τα οφέλη υπερτερούν των μειονεκτημάτων.

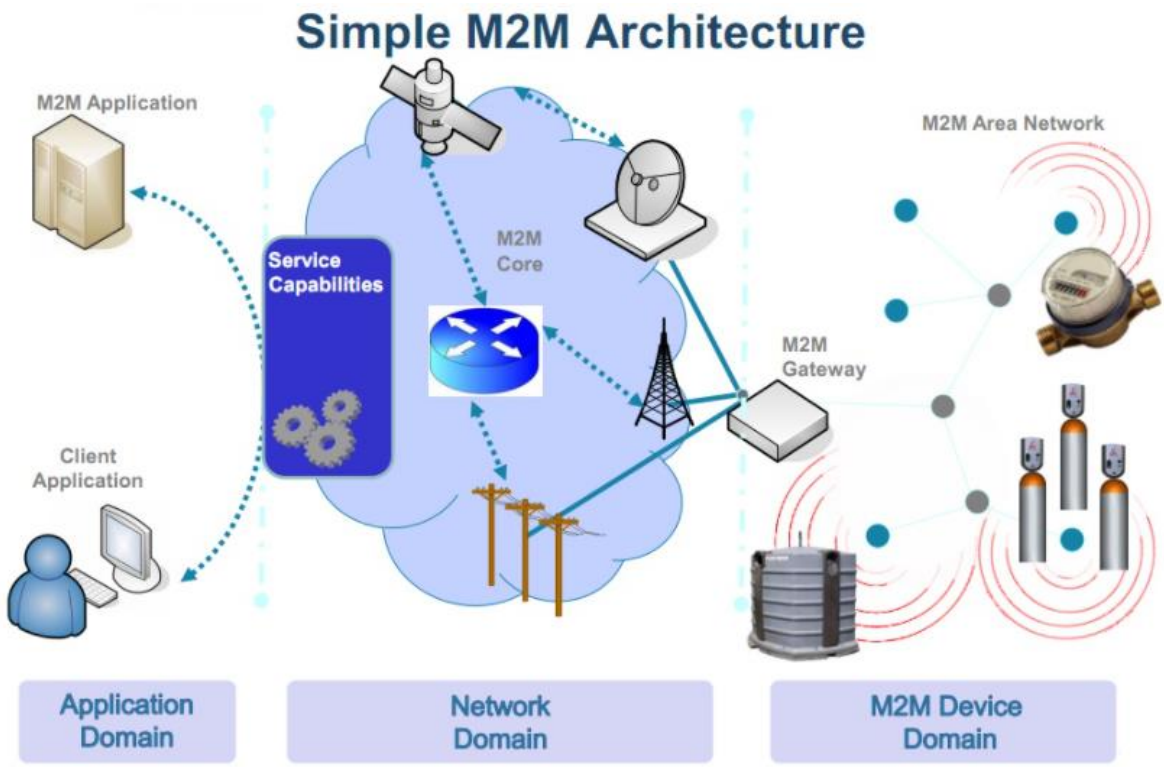
2.2 Machine to Machine Communication (Η επικοινωνία από μηχανή σε μηχανή communication)

Ο αριθμός των δικτυωμένων συσκευών ανά τον κόσμο αυξάνεται με ραγδαίους ρυθμούς. Μάλιστα οι συσκευές αυτές δεν είναι μόνο απλοί προσωπικοί υπολογιστές, κινητά τηλέφωνα, αλλά μπορεί να είναι μια οποιαδήποτε συσκευή. Αισθητήρες, μετρητές, οικιακές συσκευές, βιομηχανικά εργαλεία και μηχανήματα είναι μόνο λίγα από τα παραδείγματα των συσκευών που είτε είναι συνδεδεμένες στο διαδίκτυο είτε θα συνδεθούν μέσα στα επόμενα χρόνια. Η τεχνολογία που εστιάζει στην σύνδεση των καθημερινών μας αντικειμένων με το διαδίκτυο ονομάζεται «διαδίκτυο των πραγμάτων» (internet of things) [17]. Οι απαιτήσεις αυτές διαφέρουν σε αρκετά σημεία από τις απαιτήσεις των ήδη παγιωμένων μορφών επικοινωνίας, όπως είναι η φωνή και τα δεδομένα αρχείων. Γι' αυτόν το λόγο η νέα αυτή μορφή επικοινωνίας με τις νέες απαιτήσεις έχει αποκτήσει και διαφορετικό όνομα: επικοινωνία μηχανής προς μηχανή (Machine to Machine-m2m).

Ένα απλό παράδειγμα επικοινωνίας μηχανής προς μηχανή είναι, από κινητό σε υπολογιστή και από υπολογιστή σε κινητό. Πρόκειται για τη δικτύωση των μηχανών και των συσκευών που διαπερνούν την καθημερινή μας ζωή. Οι επικοινωνίες μηχανής προς μηχανή θα παρέχουν διαφόρων ειδών ευκολίες από μεγάλα υπολογιστικά συστήματα σε καθημερινά προϊόντα (π.χ. οικιακές συσκευές, οχήματα, κτίρια) προκειμένου να απελευθερώσουν νέα επίπεδα «έξυπνων» υπηρεσιών και εμπορίου. Η επικοινωνία μηχανής προς μηχανή έχει τη δυνατότητα να αναδιατάξει βιομηχανικές δομές και να δημιουργήσει ένα απροσδόκητο όφελος για εταιρίες που διαθέτουν την κατάλληλη τεχνογνωσία, χωρίς όμως μεγάλο κεφάλαιο προς επένδυση.

Η επικοινωνία μηχανής προς μηχανή, αφορά αλληλεπιδράσεις που μπορούν να πραγματοποιηθούν μεταξύ ατόμων και των προϊόντων τους. Ένα άτομο μπορεί να αποκτήσει δεδομένα και πληροφορίες σχετικά με την κατάσταση μιας συσκευής, την υγεία, τη θέση της, τα επίπεδα υγρών ή αναλώσιμων, τη θερμοκρασία, τα επίπεδα παραγωγικότητας και το ιστορικό συντήρησης. Οι συσκευές μπορούν επίσης να συνδεθούν σε άλλες συσκευές για κοινή χρήση περιεχομένου όπως

μουσική, ειδοποιήσεις και πληροφορίες αλυσίδας εφοδιασμού, δημιουργώντας μια απρόσκοπτη και αυτοματοποιημένη ροή δεδομένων και υπηρεσιών. Επί του παρόντος, υπάρχουν επιτυχημένες υλοποιήσεις για πύργους κυψελών, αγωγούς πετρελαίου και φυσικού αερίου, συστήματα ασφαλείας, οικιακές συσκευές και άλλα. Αυτές οι υλοποιήσεις αποφέρουν σημαντικά αποτελέσματα.



Εικόνα 5 "Βασική Επικοινωνία m2m και τα επίπεδά της "

Υπάρχουν τρία επίπεδα επικοινωνίας μηχανής προς μηχανή (m2m). Το πρώτο είναι το επίπεδο εφαρμογών. Σε αυτό το επίπεδο εφαρμόζεται το λογισμικό για εφαρμογές τεχνολογίας m2m, όπως εφαρμογές εξυπηρετητή και εφαρμογές που θα χρησιμοποιήσει ο τελικός χρήστης. Οι εφαρμογές εξυπηρετητών καθορίζονται ανάλογα με το τι είδους τεχνολογία m2m είναι διαθέσιμη, σύμφωνα με τους αισθητήρες που θα χρησιμοποιηθούν. Αυτό το επίπεδο, για να επικοινωνήσει με τον τελικό χρήστη, πρέπει να συνδεθεί με το επίπεδο Δικτύου. Ο κυριότερός του ρόλος είναι η αποτελεσματική μεταφορά της πληροφορίας μεταξύ των δύο άλλων επιπέδων. Στο επίπεδο συσκευών διακρίνονται δύο ειδών συσκευές. Αυτές οι

οποίες είναι ικανές να συνδεθούν με το δίκτυο και συσκευές οι οποίες χρειάζονται διαδικτυακή πύλη. Όπως παρατηρούμε στην παραπάνω εικόνα, ο χρήστης επικοινωνεί με μία εφαρμογή.

ΚΕΦΑΛΑΙΟ 3

Πρότυπα Διαχείρισης και Ελέγχου Συσκευών και Υπηρεσιών

ΕΙΣΑΓΩΓΗ

Τα εταιρικά δίκτυα διαθέτουν πολλά συστατικά στοιχεία. Σε βάθος χρόνου, κάποιο στοιχείο θα ξεκινήσει να υπολειτουργεί, το οποίο δεν θα παρατηρηθεί αμέσως στο δίκτυο (ή και καθόλου). Σε περίπτωση που συσσωρευτούν πολλά προβλήματα στο δίκτυο, υπάρχει μεγάλη πιθανότητα να μειωθεί δραματικά η απόδοση και η αποτελεσματικότητά του. Τα πρωτόκολλα παρακολούθησης δικτύου είναι λύσεις διαχείρισης δικτύου που υπάρχουν επειδή εμείς, ως άνθρωποι, δεν μπορούμε να παρατηρήσουμε κάθε δραστηριότητα που βρίσκεται σε εξέλιξη εντός ενός δικτύου σε πραγματικό χρόνο.

Για να κατανοήσουμε καλύτερα το πόσο σημαντικό είναι να παρακολουθείται ένα επιχειρησιακό δίκτυο, αρκεί να αντιληφθούμε ότι οι περισσότερες εταιρείες δεν θα μπορούσαν να πραγματοποιήσουν ακόμη και τις πιο βασικές λειτουργίες τους, (εξόφληση λογαριασμών, επικοινωνία με πελάτες) χωρίς ένα λειτουργικό δίκτυο. Εναλλακτικά, θα έπρεπε να δαπανήσουν πολλές περισσότερες ώρες ανθρωπίνων πόρων για να εκτελέσουν βασικές λειτουργίες, οι οποίες στις μέρες μας γίνονται ακόμη και αυτοματοποιημένα.

Τα πρωτόκολλα παρακολούθησης δικτύου είναι πρωτόκολλα που έχουν σχεδιαστεί για να διευκολύνουν την παρακολούθηση και την παροχή αναφορών για δεδομένα και κίνηση που ρέουν προς και από συνδέσμους δικτύου - μεταξύ κεντρικού υπολογιστή και συσκευής πελάτη. Τα δεδομένα που συλλέγονται υποβάλλονται σε επεξεργασία και εμφανίζονται γραφικά μέσω GUI, έτσι ώστε οι διαχειριστές δικτύου να μπορούν να χρησιμοποιούν τις πληροφορίες που παρέχονται για τη διαχείριση της δραστηριότητας του δικτύου.

Τα πρωτόκολλα δικτύου έχουν σχεδιαστεί για να διευκολύνουν τους διαχειριστές αυτών με την συντήρησή τους. Τα πρότυπα αυτά παρέχουν κρίσιμες πληροφορίες και δημιουργούν σημαντικές αναφορές που σχετίζονται με την κίνηση του δικτύου. Τα δεδομένα αυτά αποθηκεύονται και μπορούν να προβληθούν οποιαδήποτε στιγμή το θελήσει ο διαχειριστής. Ανάλογα με το είδος της πληροφορίας που ψάχνει ο διαχειριστής, μπορεί να επιλέξει να προβάλει τα κατάλληλα δεδομένα μέσω ενός περιβάλλοντος γραφικής απεικόνισης.

3.1 SNMP

Το Απλό Πρωτόκολλο Διαχείρισης Δικτύων ονομάστηκε έτσι λόγω του σχεδιασμού του. Αυτό που το χαρακτηρίζει είναι η απλουστευμένη του εκτέλεση και γιατί καταναλώνει ελάχιστους πόρους στο δίκτυο.

Το πρωτόκολλο Simple Network Management (SNMP) [10] χρησιμοποιείται για τον διαμοιρασμό πληροφοριών μεταξύ των συσκευών που ανήκουν σε ένα τοπικό δίκτυο. Επιτρέπει στις συσκευές να επικοινωνούν μεταξύ τους και ανάλογα με τον τύπο των πληροφοριών πραγματοποιούνται και οι αντίστοιχες ενέργειες. Χρησιμοποιείται κυρίως για την διαχείριση συσκευών. Δεν απαιτείται οι συσκευές να είναι ίδιες μεταξύ τους. Χρησιμοποιεί παρόμοια αρχιτεκτονική με αυτήν που χρησιμοποιούν πολλές διαδικτυακές επικοινωνίες σήμερα [18], την εξυπηρετητή - πελάτη . Σε αυτή την περίπτωση όμως, οι εξυπηρετητές ονομάζονται διευθυντές (SNMP Network Managers), ενώ αντίστοιχα οι πελάτες ονομάζονται πράκτορες (Master Agents / Subagents). Επιπροσθέτως υπάρχουν τα αντικείμενα δικτύου που δέχονται ατόφιες εντολές με σκοπό να τις εκτελέσουν (Managed Components).

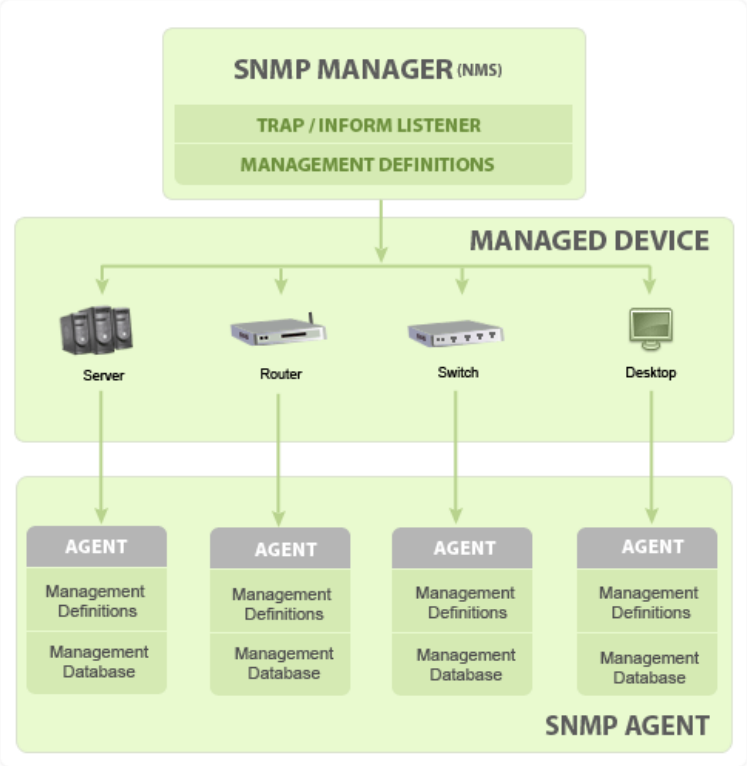
Αρχικά, ο σκοπός του SNMP [13] ήταν η συντήρηση και παρατήρηση του δικτύου. Στην συνέχεια, οι μηχανικοί δικτύου άρχισαν να προγραμματίζουν μέσω αυτού του πρωτοκόλλου και τις συσκευές και τα προγράμματα τα οποία εμπεριέχονται σε αυτό. Ένα από τα βασικά του πλεονεκτήματα είναι, πως η αρχιτεκτονική του συστήματος δεν είναι εμφανής στους υπόλοιπους χρήστες.

Στις μέρες μας, οι περισσότερες συσκευές έχουν εγκατεστημένο το SNMP, ακόμη και από πρώτη στιγμή που αγοράστηκε η συσκευή. Ανεξαρτήτως κατασκευαστή, οι συσκευές που έχουν εγκατεστημένο το SNMP μπορούν να επικοινωνήσουν μεταξύ τους. Ένα από τα βασικά πλεονεκτήματα χρήσης του SNMP είναι ότι δεν χρειάζεται να «τρέχουν» το ίδιο λειτουργικό σύστημα όλες οι συσκευές [19]. Συνοψίζοντας, επιτρέπει να διαχειρίζεσαι όλες τις δικτυακές συσκευές από μία κονσόλα, με πολύ χαμηλό κόστος επεξεργαστικής ισχύος.

3.1.1 Αρχιτεκτονική

Το SNMP βασίζεται πάνω στην αρχιτεκτονική πελάτη/εξυπηρετητή (client/server), ακολουθώντας μία παρόμοια λογική. Όπως και στο παρακάτω

σχήμα, υπάρχουν διάφορα συστατικά στοιχεία σε ένα δίκτυο. Αποτελείται από τον SNMP Διαχειριστή, από τους πράκτορες, από το κοινό μέσο με το οποίο επικοινωνούν και από την MIB [20]. Ένας πράκτορας μπορεί να είναι οποιαδήποτε δικτυακή συσκευή που έχει εγκατεστημένο το SNMP. Ο SNMP Διαχειριστής στέλνει εντολές στους πράκτορες είτε τοποθετεί Traps. Αυτές είναι οι βασικότερες λειτουργίες, τις οποίες θα τις δούμε πιο αναλυτικά παρακάτω.



Εικόνα 6 " Αρχιτεκτονική SNMP"

3.1.1.1 SNMP Managers

Οι διαχειριστές στέλνουν εντολές ώστε να συλλέξουν πληροφορίες και να διατάξουν τους πράκτορες να πραγματοποιήσουν εντολές. Το αποτέλεσμα εξαρτάται από τις πληροφορίες που συλλέγουν. Συνήθως παρακολουθούν την κατάσταση του δικτύου. Διευθυντής μπορεί να είναι ακόμα και κάποιο πρόγραμμα.

Τα μηνύματα που στέλνουν κατά κύριο λόγο είναι GetRequest, GetNextRequest και SetRequest.

3.1.1.2 SNMP Agents

Ο πράκτορας είναι λογισμικό το οποίο ελέγχει την επικοινωνία από και προς οποιαδήποτε συμβατή SNMP συσκευή. Σε μερικές συσκευές υπάρχει προ εγκατεστημένος. Υπάρχουν πράκτορες που επιστρέφουν που είναι προγραμματισμένοι να παρέχουν πολύ βασικές πληροφορίες στον διευθυντή, αλλά και άλλοι που επιστρέφουν πολλές περισσότερες. Είναι καθαρά θέμα λογισμικού, η ποσότητα και η συχνότητα των πληροφοριών που θα αποστείλουν.

Οι πράκτορες έχουν αισθητήρες (οποιασδήποτε μορφής) και συλλέγουν πληροφορίες, ενώ παράλληλα τις αποστέλλουν στους διευθυντές. Είναι σημαντικό να αναφερθεί πως σε μία εκτεταμένη αρχιτεκτονική SNMP υπάρχουν και οι Master Agents οι οποίοι είναι ενός είδους «διαμεσολαβητή» μεταξύ SNMP Network Manager & Subagent. Μπορεί να είναι από κάποιο πρόγραμμα (όπως daemon στο λογισμικό UNIX), κάποιο λογισμικό, όπως στους IOS δρομολογητές (routers).

3.1.1.3 SNMP Management Information Base

Οποιοδήποτε κομμάτι δικτυακής πληροφορίας μπορεί να ληφθεί μέσω του SNMP. Για να μην δημιουργηθεί σύγχυση, όσον αφορά την αποθήκευση των πληροφοριών, αλλά και να μην υπάρξει κάποια σύγκρουση, όπως π.χ. διπλότυπες εγγραφές, πρέπει κάθε πληροφορία να έχει το μοναδικό αναγνωριστικό της. Οι Βάσεις Δεδομένων (Management Information Base) είναι ένα σύνολο από πίνακες που συγκεκριμενοποιούν τις πληροφορίες που στέλνουν οι πράκτορες στον Διαχειριστή. Επίσης, ακόμα και ο κύριος Διαχειριστής μπορεί να ερωτήσει κάποιες πληροφορίες οι οποίες εμπεριέχονται σε αυτές τις βάσεις. Τα δεδομένα που παρέχει ο ατζέντης στον διαχειριστή, είναι πληροφορίες του συστήματος της συσκευής όπως κατάσταση των στοιχείων της συσκευής, υπερφόρτωση συσκευής, σφάλματα και άλλες χρήσιμες πληροφορίες.

Η σχηματική απεικόνιση της δομής των MIB έχει δενδρική μορφή (tree structure), τα φύλλα της οποίας αντιπροσωπεύουν τους ορισμούς των μεταβλητών και ονομάζονται αντικείμενα. Στο κάθε αντικείμενο αντιστοιχίζεται ένας μοναδικός αριθμός, που λέγεται Object Identifier (OID). Το OID αυτό περιγράφει τη διαδρομή που ακολουθείται για το συγκεκριμένο αντικείμενο μέσα στο δέντρο. Ο κάθε κόμβος του δέντρου μπορεί να επεκταθεί με την προσθήκη μικρότερων δέντρων.

Καθολικά, τα αντικείμενα είναι μεταβλητές που καθορίζονται στην Management Information Base (MIB). Υπάρχουν κανόνες που καθορίζουν την MIB. Το συντακτικό μιας MIB καθορίζεται στο Structure of Management Information [11] [12] (SMI), αλλά και στην Abstract Syntax Notation One (ASN.1). Ο τρόπος που κωδικοποιείται μία MIB δίνεται από την Basic Encoding Rules (BER).

Κάθε αντικείμενο αποτελείται από ένα μοναδικό αναγνωριστικό object identifier (OID). Αποτελείται από ένα σύνολο φυσικών αριθμών (π.χ. το OID για την περιγραφή ενός συστήματος είναι 1.3.6.1.2.1.1.1.)

Ο ορισμός των διαχειριζόμενων αντικειμένων μπορεί να χωριστεί σε τρία μέρη:

- Όνομα
- Τύπος
- Συντακτικό και Κωδικοποίηση.

3.1.1.3.1 Όνομα

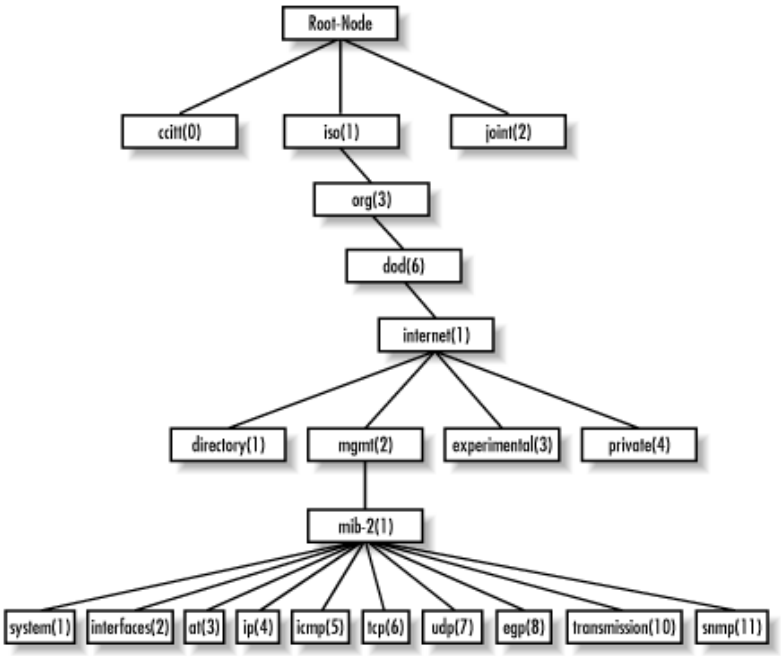
Το όνομα ή αλλιώς object identifier (OID) καθορίζει ένα διαχειριζόμενο αντικείμενο. Ως επί το πλείστων υπάρχουν δύο ειδών μορφές, η μία είναι αριθμητική και η άλλη είναι σε μορφή ευκολοδιάβαστη για τον άνθρωπο. Για παράδειγμα: iso.org.dod.internet.mgmt.mib-2.system.sysDescr είναι το περιγραφικό αντικείμενο (object descriptor) το οποίο αντιστοιχεί στο αναγνωριστικό αντικείμενο (object identifier) 1.3.6.1.2.1.1.1.

Τα ονόματα που αποτελούνται κυρίως από αριθμούς χρησιμοποιούνται κυρίως στους πράκτορες και στην επικοινωνία που έχουν. Από την άλλη μεριά, τα μη αριθμητικά ονόματα χρησιμοποιούνται στο σταθμό διαχείρισης για τη διευκόλυνση των χρηστών.

Ένας ακόμη ρόλος της MIB είναι να μεταφράζει τα μη αλφαριθμητικά ονόματα σε αριθμητικά ονόματα και το αντίστροφο. Εάν προσπαθήσει κάποιος να

ανακαλέσει πληροφορίες για οποιοδήποτε αντικείμενο, χρησιμοποιώντας μη αλφαριθμητικό όνομα (από το σταθμό διαχείρισης), η MIB θα το μεταφράσει σε αριθμητικό και θα το αναζητήσει στην βάση δεδομένων της. Με αυτή τη μέθοδο, μπορεί να εγγυηθεί για την εγκυρότητα και την ακεραιότητα των δεδομένων και του δικτύου.

Τα αντικείμενα είναι τα φύλλα στο δέντρο. Για να μην υπάρχει κάποια εμπλοκή ανάμεσα στα αντικείμενα, ονομάζονται βάσει της διαδρομής από την ρίζα του δέντρου μέχρι το φύλλο.



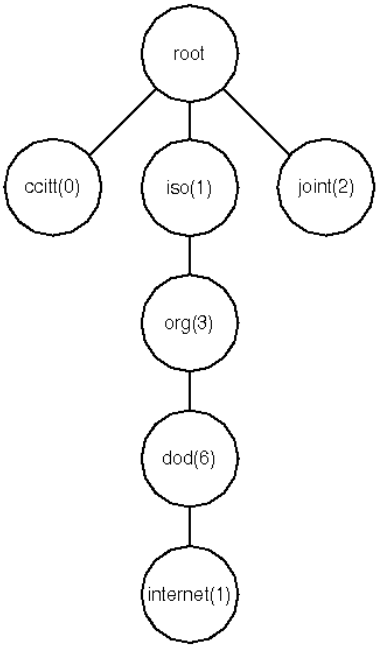
Εικόνα 7 " Δέντρο MIB"

3.1.1.3.2 Τύπος και Συντακτικό

Ο τύπος δεδομένων των αντικειμένων καθορίζεται από ένα υποσύνολο της Συντακτικής Σημειογραφίας Ένα (Abstract Syntax Notation One ASN.1). Είναι ένας τρόπος καθορισμού των δεδομένων που παρουσιάζονται και μεταδίδονται μεταξύ των managers και των πρακτόρων.

3.1.1.3.3 Κωδικοποίηση

Ένα κείμενο κωδικοποιείται σε μία σειρά οκτάδων χρησιμοποιώντας τους βασικούς κανόνες κωδικοποίησης (Basic Encoding Rules BER). Καθορίζεται ο τρόπος κωδικοποίησης και αποκωδικοποίησης των αντικειμένων ώστε να μπορούν να μεταδοθούν μέσω του τοπικού δικτύου Ethernet.



Εικόνα 8 "Δέντρο MIB"

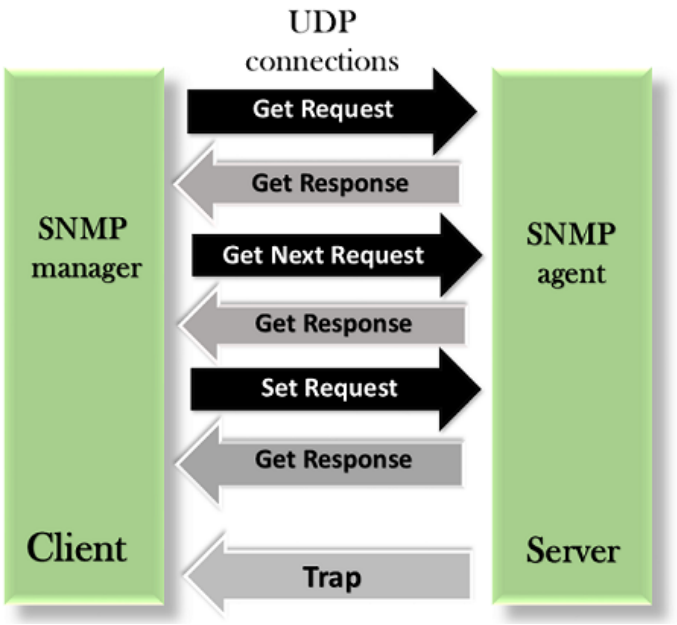
Τα Managed Objects είναι οργανωμένα σε μία ιεραρχία που μοιάζει με δέντρο. Το object ID καθορίζεται από μία σειρά από ακέραιους που βασίζονται σε κάθε κόμβο στο δέντρο, χωρισμένο με τελείες (.) . Στο παραπάνω δέντρο, ο αρχικός κόμβος ονομάζεται ρίζα, οποιοσδήποτε άλλος κόμβος ο οποίος έχει υπό-κόμβους από κάτω του, ονομάζεται υπό-δέντρο. Οποιοσδήποτε άλλος κόμβος ονομάζεται φύλλο.

Το όνομα { iso org dod internet } είναι συμβολική αναπαράσταση για τον ακέραιο 1.3.6.1 . Και οι δύο όροι αναφέρονται στο αναγνωριστικό αντικειμένου του διαδικτύου. Πρακτικά το 1.3.6.1 μπορεί να αναφερθεί και ως «internet». Είναι πολύ

σημαντικό να επισημανθεί πως στο μήνυμα μεταξύ της επικοινωνίας του πράκτορα-διευθυντή, χρησιμοποιούνται μόνο σειρές αριθμητικών.

3.1.1.3.4 SNMP Μηνύματα

Στην παρακάτω εικόνα, αντικατοπτρίζεται η επικοινωνία που έχει ένα ένας αφέντης με έναν πράκτορα. [15] Αναφέρονται τα πιο βασικά μηνύματα που ανταλλάσουν αλλά και η αλληλουχία.



Εικόνα 9 " Αμφίδρομη επικοινωνία SNMP Manager-Agent"

3.1.1.3.4.1 GetRequest

Το SNMP πρότυπο περιέχει διάφορους τύπους μηνυμάτων για να συλλέξει πληροφορίες. Ένα από αυτά είναι το GetRequest. Το στέλνει η διευθύνουσα συσκευή για να ζητήσει δεδομένα από κάποιον ατζέντη. Σε απόκριση αυτού του μηνύματος ο πράκτορας στέλνει με την τιμή του μηνύματος.

3.1.1.3.4.2 GetNextRequest

Αυτό το μήνυμα χρησιμοποιείται για να εξερευνήσει τα δεδομένα τα οποία είναι διαθέσιμα στον Ατζέντη. Ο διαχειριστής μπορεί να ρωτάει συνέχεια μέχρι να μην υπάρχουν άλλα δεδομένα. Με αυτό τον τρόπο ο διαχειριστής αποκτά την γνώση των διαθέσιμων δεδομένων στους ατζέντες. Η απόκριση που λαμβάνει ο αφέντης, είναι η επόμενη λεξικογραφικά δεσμευμένη μεταβλητή στην MIB.

3.1.1.3.4.3 GetBulkRequest

Αυτό το μήνυμα χρησιμοποιείται για να ληφθεί μεγάλος όγκος δεδομένων με μία μόνο απόκριση από τον διαχειριστή. Πρωτοεμφανίστηκε στην έκδοση SNMPv2.

3.1.1.3.4.4 SetRequest

Είναι μία οδηγία που δίνει ο αφέντης στον ατζέντη για την μετατροπή μία τιμής μιας μεταβλητής ή ακόμα και μία λίστα μεταβλητών. Κατά την διαδικασία αλλαγής των μεταβλητών απαγορεύεται να εκτελεστούν άλλες αλλαγές. Ο πράκτορας αποκρίνεται με ένα μήνυμα στο οποίο εμπεριέχονται οι αλλαγές που έλαβαν χώρα.

3.1.1.3.4.5 Response

Είναι ένα μήνυμα το οποίο στέλνεται από τον ατζέντη εφόσον έχει ερωτήσει ο διαχειριστής. Ο σκοπός του είναι να επιβεβαιώσει πως το αίτημα του αφέντη έχει παραληφθεί. Όταν έχει προηγηθεί κάποιο μήνυμα τύπου Get, τότε θα περιέχει τα δεδομένα που του ζητήθηκαν. Όταν έχει προηγηθεί μήνυμα τύπου Set, τότε θα περιέχει την καινούρια μεταβλητή σαν επιβεβαίωση ότι η αξία άλλαξε. Επίσης υπήρχε και στην πρώτη έκδοση του πρωτοκόλλου (SNMPv1) με την ονομασία GetResponse.

3.1.1.3.4.6 Trap

Είναι μία μέθοδος επικοινωνίας μεταξύ του διαχειριστή και του πράκτορα. Η επικοινωνία που έχουν είναι ασύγχρονη. Ουσιαστικά ο διαχειριστής καθορίζει μία συνθήκη. Όταν η συνθήκη ικανοποιηθεί, τότε ο πράκτορας ειδοποιεί τον διαχειριστή. Το μήνυμα που στέλνει ο πράκτορας περιέχει την μεταβλητή sysUpTime, η οποία προσδιορίζει κατά πόσο χρόνο το εκάστοτε μηχάνημα είναι σε λειτουργία.

3.1.1.3.4.7 InformRequest

Πρωτοεμφανίστηκε στο SNMPv2c. Όπως και η Trap είναι μέθοδος ασύγχρονης επικοινωνίας. Χρησιμοποιείται για να αναγνωρίσει εάν ένα μήνυμα είδους Trap έχει ληφθεί από τον διαχειριστή ή όχι. Επιλύει το πρόβλημα επιβεβαίωσης που υπήρχε μεταξύ ενός αφέντη και ενός πράκτορα. Ουσιαστικά οποιοσδήποτε πράκτορας μπορεί να παραμετροποιηθεί να τοποθετεί Τραπ συνέχεια, μέχρι να δεχτεί ένα Inform μήνυμα. Είναι το ίδιο με το μήνυμα Trap, όμως προστίθεται ένα μήνυμα επιβεβαίωσης (ACK). Οι ατζέντες μπορούν να παραμετροποιηθούν να τοποθετούν Traps κατ' εξακολούθηση, έως ότου να λάβουν ένα μήνυμα τύπου Inform.

3.1.2 Πρωτόκολλα

Το SNMP χρησιμοποιεί κυρίως UDP, για την ορθή επικοινωνία μεταξύ των Διαχειριστών και ατζεντών αλλά επίσης μπορεί και να χρησιμοποιήσει το πρωτόκολλο TCP . Χρησιμοποιώντας το πρωτόκολλο UDP , δεν μπορεί να επιτευχθεί σύνδεση από άκρο σε άκρο. Ένα ακόμα πλεονέκτημα του UDP [2] είναι η εξοικονόμηση πόρων στα τερματικά, καθώς δεν χρειάζεται να διατηρούνται οι συνδέσεις. Εφόσον οι κυριότερες λειτουργίες του SNMP [8] στηρίζονται στην άμεση απόκριση και αποστολή πληροφοριών, είναι πιο βολικό να χρησιμοποιούμε UDP.

Το TCP απαιτεί την εγκαθίδρυση σύνδεσης μεταξύ των οντοτήτων, αυτό μεταφράζεται παράλληλα και ως περισσότερα δεδομένα, τα οποία πρέπει να μεταδοθούν. Σε ένα τοπικό δίκτυο, το οποίο υπάρχουν συμφορήσεις, όσα περισσότερα δεδομένα στέλνουμε, τόσο μεγαλώνει η πιθανότητα να αυξηθεί η συμφόρηση στο δίκτυο. Σε περίπτωση που χαθεί κάποιο πακέτο, αποστέλλονται ξανά τα δεδομένα. Όλα αυτά συνεπάγονται με την αύξηση της κίνησης του δικτύου. Παρόλο που το TCP μπορεί να εγγυηθεί για την σειρά των πακέτων που θα σταλούν, δεν αποτελεί αρκετά σημαντικός λόγος για να προτιμάται.

Παράδειγμα επικοινωνίας μεταξύ SNMP Manager – SNMP Agent χρησιμοποιώντας TCP:

```
client sends SYN to server

server sends SYN/ACK to client
```

```
client sends ACK to server - socket is now established

client sends DATA to server

server sends ACK to client

server sends RESPONSE to client

client sends ACK to server

client sends FIN to server

server sends FIN/ACK to client

client sends ACK to server - socket is torn down
```

Παράδειγμα επικοινωνίας μεταξύ SNMP Manager – SNMP Agent χρησιμοποιώντας UDP

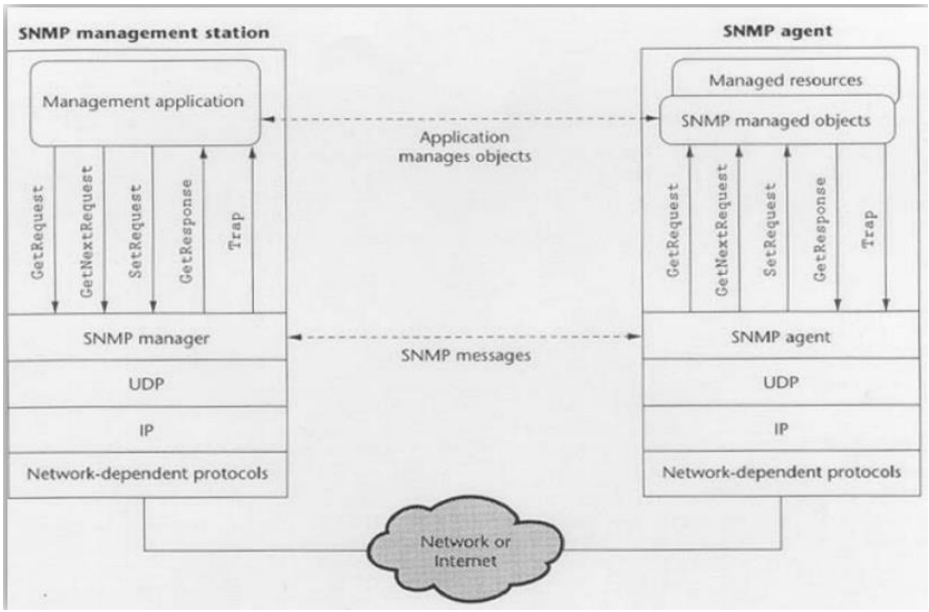
```
client sends request to server

server sends response to client
```

Για τους παραπάνω λόγους, για τις περισσότερες λειτουργίες του SNMP χρησιμοποιείται το πρωτόκολλο UDP.

3.1.3 Λειτουργία

Το SNMP χρησιμοποιεί 3 λειτουργίες:

- Read ο διαχειριστής λαμβάνει την τιμή των αντικειμένων από τον ατζέντη
- Write διαχειριστής καθορίζει τις τιμές από τα αντικείμενα στον ατζέντη
- Trap πράκτορας ειδοποιεί τον διαχειριστή για κάποια προκαθορισμένη αλλαγή στο σύστημα.



Εικόνα 10 "SNMP Operation"

Αρχικά ο Αφέντης στέλνει ένα μήνυμα τύπου UDP και περιμένει για απάντηση [14]. Εάν ο πράκτορας δεν αποκριθεί εγκαίρως, ή και καθόλου, τότε θα θεωρήσει ο Αφέντης πως η αποστολή του μηνύματος απέτυχε. Σε αυτή την περίπτωση, εάν ο αφέντης έχει προγραμματιστεί κατάλληλα, θα αναμεταδώσει το πακέτο. Αυτό το σενάριο ισχύει για τις περισσότερες περιπτώσεις. Σε περίπτωση που ο αφέντης στείλει ένα μήνυμα τύπου παγίδας (Trap), στο οποίο δεν περιμένει κάποια απόκριση από τον ατζέντη, τότε δεν γνωρίζει αν έχει φτάσει ή όχι.

Μία οπτική γωνία χαμηλού επιπέδου. Το SNMP χρησιμοποιεί την πόρτα 161 (UDP) για να στείλει και να λάβει αιτήματα και την πόρτα 162 για να δεχθεί Traps από τις διαχειριζόμενες συσκευές. Σε επίπεδο εφαρμογής, ανάλογα πως έχει καθοριστεί ο αφέντης, στέλνει ένα αίτημα ή μία παγίδα σε μία διαχειριζόμενη συσκευή. Στο επίπεδο μεταφοράς ενθυλακώνεται το μήνυμα και επιτρέπει τους χρήστες να επικοινωνήσουν μεταξύ τους. Το επίπεδο δικτύου προσπαθεί να παραδώσει το μήνυμα στον παραλήπτη, ανάλογα με την λογική διεύθυνση που έχει λάβει. Το επίπεδο ζεύξης είναι υπεύθυνο για την λήψη των πακέτων από το φυσικό μέσο και να τα στείλει στην στοίβα πρωτοκόλλου, με σκοπό να επεξεργαστούν από τον παραλήπτη.

3.1.4 Εξέλιξη των εκδόσεων SNMP

3.1.4.1 SNMPv1

Η πρώτη έκδοση του SNMP κάνει χρήση των πρωτοκόλλων IP, UDP, OSI CLNS, AppleTalk DDP, NovelPX. Εστιάζει κυρίως στο να ορίσει ένα πρωτόκολλο για την ανταλλαγή των διοικητικών πληροφοριών [16]. Επίσης καθορίζει την κωδικοποίηση, το συντακτικό και το πρότυπο ταυτοποίησης των οντοτήτων αλλά και των μηνυμάτων που ανταλλάσσουν αυτές. Ένα από τα μεγαλύτερα αρνητικά της είναι ότι τα αντικείμενα των MIB δεν πρέπει να περιέχουν «ευαίσθητες» πληροφορίες αλλά ούτε και να ελέγχουν «κρίσιμες» λειτουργίες [19]. Εξίσου σημαντικό είναι η έλλειψη κρυπτογράφησης. Ο μοναδικός μηχανισμός ασφάλειας που διαθέτει είναι μία διαδικασία πιστοποίησης αυθεντικότητας, η οποία θεωρείται απαρχαιωμένη στις μέρες μας. Αυτό έχει ως αποτέλεσμα, την εύκολη παρακολούθηση και τροποποίηση των δεδομένων από μη εξουσιοδοτημένους χρήστες.

3.1.4.2 SNMPv2

Όταν δημοσιεύτηκε η έκδοση SNMPv2 [3], αρκετοί χρήστες ήταν ικανοποιημένοι, επειδή συμπλήρωνε πολλά κενά από την πρώτη έκδοση. Όμως αυτό δεν ήταν αρκετό. Οι μηχανισμοί ασφάλειας ήταν πολύπλοκοι. Για αυτό το λόγο κυκλοφόρησαν άλλες δύο εκδόσεις οι οποίες χρησιμοποιούνται περισσότερο, η έκδοση SNMPv2c (Community-Based) αλλά και η SNMPv2u (User-Based). Αυτές οι εκδόσεις χρησιμοποιούνται ακόμη και σήμερα. Επίσης προστέθηκαν κάποιες εντολές για να λαμβάνονται περισσότερα δεδομένα από τον πράκτορα.

3.1.4.3 SNMPv3

Είναι αρκετά θετικό το γεγονός ότι δεν υπάρχουν πολλά προβλήματα λειτουργιών μεταξύ των μηχανημάτων που χρησιμοποιούν διαφορετική έκδοση του

πρωτοκόλλου. Η τελευταία έκδοση του SNMP(v3) προσπαθεί να επιλύσει τα περισσότερα προβλήματα και να καλύψει τα κενά που έχουν οι προηγούμενες εκδόσεις. Το βασικότερο πλεονέκτημά της είναι, πως περιέχει βασικούς μηχανισμούς αυθεντικοποίησης και ακεραιότητας των δεδομένων. Υπάρχει η δυνατότητα πρόβλεψης και εν μέρει επίλυσης του προβλήματος κακόβουλης αναπαραγωγής πακέτων «replay attack" από κακόβουλο χρήστη. Εάν δεχτεί κάποιος πράκτορας μήνυμα, από κάποια συσκευή που δεν έχει παρόμοια παραμετροποίηση με αυτόν, τότε θα απορρίψει το μήνυμα και δεν θα δημιουργήσει απόκριση. Στην δεύτερη έκδοση του πρωτοκόλλου, την SNMPv2, προστέθηκαν ακόμη δύο PDUs, τα GetBulkRequest και InformRequest, τα οποία διατηρήθηκαν και στο SNMPv3.

3.2 ICMP

Το ICMP είναι ένα πρωτόκολλο επιπέδου διαδικτύου. Χρησιμοποιείται κυρίως από δικτυακές συσκευές όπως δρομολογητές. Η πιο συχνή λειτουργία του είναι για να δημιουργήσει μηνύματα λανθασμένης επικοινωνίας μεταξύ δρομολογητών. Οποιαδήποτε συσκευή χρησιμοποιεί το πρωτόκολλο IP [1] μπορεί να στείλει, δεχτεί και να δημιουργήσει μηνύματα του ICMP.

Το ICMP συμπληρώνει το IP πρωτόκολλο. Το πρωτόκολλο δικτύου IP εστιάζει κυρίως στο να αποστείλει με τον καλύτερο δυνατό τρόπο τα δεδομένα. Οι βασικές του λειτουργίες είναι να έχουν επικοινωνία οι συσκευές, να δρομολογηθούν τα πακέτα και να ληφθούν. Δεν περιέχει λειτουργίες αναμετάδοσης και ελέγχους εάν τα πακέτα έχουν ληφθεί με την σωστή σειρά. Εφόσον δεν είναι αξιόπιστο το IP, τότε το ICMP μπορεί να επιβεβαιώσει τυχόν πιθανά λάθη στην ακολουθία ενός δικτύου. Εφόσον δεν υπάρχει πουθενά στην δομή του IP η προώθηση και ο έλεγχος μηνυμάτων, το ICMP μπορεί να προσθέσει λειτουργίες και περισσότερες εφαρμογές π.χ. Όταν ένα δεδομένο-διάγραμμα δεν μπορεί να φτάσει στον προορισμό του, τότε το ICMP ειδοποιεί τον αποστολέα.

Ο σκοπός του ICMP είναι να παρέχει ανατροφοδότηση σχετικά με τα πακέτα του αποστολέα. Άλλος ένας πολύ σημαντικός στόχος που πετυχαίνουν τα μηνύματα ICMP είναι να ειδοποιούν τον αποστολέα ενός μηνύματος IP σε περίπτωση που υπήρχε κάποιο πρόβλημα με την μετάδοση του πακέτου.

3.2.1 Λειτουργίες

Απαιτείται υλοποίηση του IP πρωτοκόλλου για να υποστηριχθεί το ICMP. Θεωρείται ένα μέρος του IP, παρόλα αυτά, ανήκει στο επίπεδο μεταφοράς. Οι βασικές του λειτουργίες είναι η αναφορά λαθών, ο έλεγχος της κίνησης και η ανακατεύθυνση πρώτων διεξόδων (first-hop gateway). Το ICMP πακέτο διαθέτει όλο το IP πακέτο μέσα του επομένως ξέρει ποιο πακέτο δεν παραδόθηκε σωστά. Επιπροσθέτως, το ICMP στέλνει αιτήματα και πληροφορίες για λειτουργίες του δικτύου.

Το ICMP προσφέρει ανατροφοδότηση και πληροφορίες σχετικά με σφάλματα, μηνύματα ελέγχου και ερωτήματα διαχείρισης. Το πρώτο πεδίο κώδικα στο μπλοκ ICMP καταφέρνει μόνο του να μεταφέρει πολλές πληροφορίες.

```
0: Echo Reply.

3: Destination is unreachable.

4: Source quench.

5: Redirect.

8: Echo Request.

9: Router advertisement reply.

10: Router solicitation.

11: Time Exceeded.
```

Το πρώτο και το όγδοο μήνυμα χρησιμοποιούνται στην εντολή Ping κυρίως. Το τέταρτο μήνυμα δηλώνει τη χρήση άλλου δρομολογητή. Το ενδέκατο μήνυμα χρησιμοποιείται στην εντολή traceroute.

3.2.1.1 Ping

Η εντολή Ping χρησιμοποιεί το πρωτόκολλο ICMP. Η εντολή ping ελέγχει αν ένας προορισμός είναι προσπελάσιμος. Το ping κάνει χρήση των μμηνυμάτων του ICMP echo request και echo reply. Για να γίνει έλεγχος αν είναι προσπελάσιμος ο προορισμός, το ping στέλνει το echo request και περιδένει απόκριση από τον παραλήπτη για ένα συγκεκριμένο χρόνο. Όταν περάσει ο χρόνος αυτός,

επαναλαμβάνει την διαδικασία, δηλαδή στέλνει echo request και αν δεν λάβει πάλι κάποια απόκριση, τότε ο προορισμός θεωρείται πως είναι μη προσπελάσιμος. Εάν η επικοινωνία είναι αμφίδρομη και επιτυχής, τότε ο παραλήπτης στέλνει το μήνυμα echo reply του ICMP. Μπορεί να χρησιμοποιηθεί η εντολή αυτή για την αποσφαλμάτωση της επικοινωνίας μεταξύ των συσκευών αλλά και για την παρακολούθηση της κίνησης του δικτύου. Ο χρόνος που χρειάζεται για να σταλεί ένα πακέτο και να λάβει απόκριση από τον παραλήπτη, δηλαδή να «επιστρέψει» στον αποστολέα, ονομάζεται round-trip time. Τέλος, εμφανίζεται η τιμή Time-to-live (TTL), η οποία έχει μία τιμή από το πακέτο πρωτοκόλλου IP. Η μεταβλητή αυτή καθορίζει εάν ένα πακέτο μπορεί να προωθηθεί από έναν δρομολογητή ή όχι. Ο σκοπός της είναι να μην προωθείται ένα πακέτο στο διαδίκτυο επ' αόριστον. Κάθε φορά που το πακέτο περνάει από έναν δρομολογητή, τότε αφαιρείται κατά ένα η τιμή που έχει. Εάν φτάσει στο 0, τότε το πακέτο απορρίπτεται.

Παράδειγμα εντολής ping:

```
Pinging 1.1.1.1 with 32 bytes of data:
Reply from 1.1.1.1: bytes=32 time=36ms TTL=56
Reply from 1.1.1.1: bytes=32 time=28ms TTL=56
Reply from 1.1.1.1: bytes=32 time=37ms TTL=56
Reply from 1.1.1.1: bytes=32 time=30ms TTL=56

Ping statistics for 1.1.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 28ms, Maximum = 37ms, Average = 32ms
```

Εικόνα 11 " Παράδειγμα εντολής Ping"

Όπως παρατηρούμε στην εικόνα, ο αποστολέας έστειλε τέσσερα πακέτα echo request. Ο παραλήπτης αποκρίθηκε με τέσσερα πακέτα επίσης. Επίσης μπορούμε να δούμε τον όγκο των δεδομένων που στάλθηκαν (bytes=32), αλλά και σε πόσο χρόνο χρειάστηκε για να πάει και να επιστρέψει το πακέτο (time=36ms). Η τιμή TTL έχει την τιμή 56. Μπορούμε να διακρίνουμε και τα στατιστικά που βγαίνουνε από αυτή την εντολή (Μέσος χρόνος απόκρισης 32ms).

3.2.1.2 Traceroute

Μία ακόμα εξίσου σημαντική εντολή είναι η traceroute (ή και tracert). Καθορίζει τη δρομολόγηση σε έναν προορισμό, αποστέλλοντας πακέτα ICMP echo. Σε αυτά τα πακέτα, η εντολή χρησιμοποιεί διάφορες τιμές στο πεδίο Time-To-Live (TTL) του πρωτοκόλλου IP. Επειδή κάθε δρομολογητής κατά μήκος της διαδρομής, ελαττώνει την τιμή TTL ενός πακέτου κατά 1, πριν από την προώθηση του πακέτου, η τιμή TTL είναι αποτελεσματικά μετρητής μεταπηδήσεων. Όταν η τιμή TTL ενός πακέτου φτάσει στο μηδέν (0), ο εκάστοτε δρομολογητής στέλνει μια απόκριση τύπου ICMP " υπέρβαση χρόνου" μηνυμάτων στον αποστολέα. Η εντολή traceroute αποστέλλει το πρώτο πακέτο ηχούς με μια τιμή TTL ίση με 1 και αυξάνει το TTL κατά 1 σε κάθε επόμενη μετάδοση, έως ότου αποκριθεί ο προορισμός ή έως ότου επιτευχθεί η μέγιστη τιμή TTL. Τα ICMP "υπέρβαση χρόνου" μηνύματα των ενδιάμεσων δρομολογητών εμφανίζουν την αποστολή της διαδρομής. Είναι σημαντικό να επισημανθεί, ότι ορισμένοι δρομολογητές είναι προγραμματισμένοι να μην αποκρίνονται όταν η τιμή TTL έχει την τιμή 0. Αυτοί οι δρομολογητές δεν είναι ορατοί στην εντολή traceroute. Η traceroute εκτυπώνει μια ταξινομημένη λίστα με τους ενδιάμεσους δρομολογητές που επιστρέφουν ICMP "Υπέρβαση χρόνου" μηνύματα.

3.2.2 Ζητήματα Ασφάλειας

Για έναν κακόβουλο χρήστη, είναι αρκετό να γνωρίζει πως ένα σύστημα βρίσκεται συνδεδεμένο στο διαδίκτυο. Μία ανάλυση των αποκρίσεων της εντολής ping μπορεί να «μαρτυρήσει» εάν το σύστημα είναι επηρεάζεται με κάποιου είδους επίθεση DDoS. Ένας κακόβουλος χρήστης του διαδικτύου, μπορεί να καταλάβει τι είδους λογισμικό τρέχει στο σύστημα, ανάλογα με τον τρόπο που αποκρίνεται.

Τα περισσότερα εργαλεία διείσδυσης, προσπαθούν να λάβουν όσες περισσότερες πληροφορίες μπορούν. Συνήθως εκμεταλλεύονται την αφέλεια των διαχειριστών συστημάτων και κάνουν ping σε όλο το διαθέσιμο εύρος θυρών σε ένα σύστημα. Με αυτό τον τρόπο μπορούν να λάβουν όσες το δυνατόν πληροφορίες μπορούν για ένα σύστημα. Έτσι οι κακόβουλοι χρήστες μπορούν να εκμεταλλευτούν τις πληροφορίες που έλαβαν και να ενεργήσουν αναλόγως.

3.2.2.1 Ping Flood

Αυτή η επίθεση είναι αρκετά συχνή. Είναι αρκετά εύκολο να υλοποιηθεί επειδή δεν χρειάζεται ο επιτιθέμενος να διαθέτει ισχυρό υπολογιστικό σύστημα. Βασική προϋπόθεση για να επιτευχθεί είναι, ο επιτιθέμενος να μπορεί να στείλει περισσότερα πακέτα, από ότι μπορεί να λάβει ο παραλήπτης. Είναι πολύ περισσότερο αποτελεσματική όταν ο κακόβουλος χρήστης έχει μεγαλύτερη (σε όγκο δεδομένων) σύνδεση δικτύου από τον παραλήπτη.

Ουσιαστικά ο επιτιθέμενος στέλνει ICMP πακέτα στον αποστολέα, τύπου “echo request” . Το θύμα θα αποκριθεί με ICMP πακέτα τύπου “echo reply” , καταναλώνοντας έτσι και το εύρος εσωτερικής και εξωτερικής ζώνης. Εάν το σύστημα που δέχεται την επίθεση είναι αργό, είναι πολύ πιθανό να καταναλώσει πολλούς υπολογιστικούς πόρους ώστε να αντιμετωπίσει την επίθεση, με αποτέλεσμα να επιβραδυνθεί. Ως εκ τούτου, εάν ο επιτιθέμενος δεν το πετύχει αυτό, σίγουρα θα έχει πετύχει την απώλεια πολλών πακέτων του θύματος.

3.2.2.2 Smurf Attack

Είναι μια κατανεμημένη Denial Of Service (DOS) επίθεση με μεγάλο αριθμό ICMP πακέτων [21]. Ουσιαστικά η IP του θύματος μεταδίδεται σε ένα γνωστό δίκτυο υπολογιστών χρησιμοποιώντας μία broadcast IP διεύθυνση. Ο κακόβουλος χρήστης στέλνει το πακέτο, προβάλλει την IP του θύματος στα άλλα συστήματα. Όταν το πακέτο φτάσει στα υπόλοιπα συστήματα, θα απαντήσουν σε αυτό το πακέτο, με αποτέλεσμα, εάν υπάρχουν πολλοί συνδεδεμένοι χρήστες, τότε θα προκληθεί το φαινόμενο flood . Αυτή η διαδικασία, θα έχει ως αποτέλεσμα να επιβραδύνει τον υπολογιστή που δέχεται την επίθεση και ως εκ τούτου να μην μπορεί να λειτουργήσει.

Θα μπορούσε να λειτουργήσει αποτελεσματικά το να διαμορφώσουμε του χρήστες του δικτύου με το να μην αποκρίνονται σε ερωτήματα ICMP τα οποία είναι broadcast. Επίσης, πρέπει να προγραμματίσουμε τους δρομολογητές, να μην προωθούν πακέτα τέτοιου τύπου. Μία άλλη λύση είναι να φιλτράρονται τα πακέτα, με σκοπό να απορρίπτονται με βάση την διεύθυνση προέλευσης.

Η συγκεκριμένη επίθεση έχει κάποιες λύσεις, οι οποίες συνίσταται ανάλογα με το πως λειτουργεί και έχει προγραμματιστεί το δίκτυο. Δυστυχώς δεν είναι εφικτό το να υπάρχει ένα άριστο δίκτυο, χωρίς δυσλειτουργίες ή ακόμα και χωρίς τρωτά

σημεία. Με το απενεργοποιούμε κάποιες λειτουργίες του δικτύου, σίγουρα θα υπάρχουν άλλα στοιχεία στο δίκτυο που θα υπολειτουργούν.

ΚΕΦΑΛΑΙΟ 4

OpenC2

ΕΙΣΑΓΩΓΗ

Στις μέρες μας, οι περισσότερες εταιρείες για να καταφέρουν να έχουν μία ασφαλή αρχιτεκτονική, προσπαθούν να διαθέτουν μηχανισμούς ασφαλείας από πολλούς κατασκευαστές. Εάν για παράδειγμα κάποια εταιρία διαθέτει μηχανισμούς ασφαλείας, από έναν και μόνον ένα κατασκευαστή, σε περίπτωση που βρεθεί κάποια καινούργια ευπάθεια στο λογισμικό που χρησιμοποιείται από τους μηχανισμούς, τότε θα είναι «ευάλωτο» σε επιθέσεις zero-day [24] το δίκτυο της εταιρείας. Η μεγαλύτερη πρόκληση που πρέπει να αντιμετωπίσει κάποιος οργανισμός που χρησιμοποιεί μηχανισμούς ασφαλείας από διαφορετικούς κατασκευαστές είναι η συντήρηση τους. Η διαδικασία ανανέωσης και αναβάθμισης των λογισμικών είναι χρονοβόρα. Ένα άλλο μειονέκτημα, το να διαθέτει κανείς μηχανισμούς ασφαλείας από διαφορετικούς κατασκευαστές είναι ότι η επικοινωνία των μεταξύ τους συστημάτων είναι προβληματική.

Τις περισσότερες φορές, ο διευθυντής του δικτύου ελέγχει ένα προς ένα τα συστήματα και τα επιμέρους συστατικά τους και αν κρίνει απαραίτητη κάποια αλλαγή (όπως αναβάθμιση λογισμικού), τότε την εφαρμόζει. Σε περίπτωση που δεχθεί επίθεση το δίκτυο της εταιρίας, τότε δεν υπάρχει ολοκληρωμένη επικοινωνία μεταξύ των επιμέρους συστημάτων. Οι περισσότεροι κατασκευαστές προσπαθούν να πουλήσουν μία ολοκληρωμένη σουίτα μηχανισμών ασφαλείας, η οποία όμως, δεν σημαίνει απαραίτητα ότι θα είναι πιο αποτελεσματική από μία ενοποιημένη λύση μεταξύ διαφόρων κατασκευαστών.

Το Open Command and Control πρότυπο (OpenC2) μπορεί να λύσει το πρόβλημα της επικοινωνίας μεταξύ των μηχανισμών ασφαλείας διαφορετικών κατασκευαστών. Διαθέτει μία λειτουργία η οποία καθορίζει ένα μοτίβο συμπεριφοράς από μία συσκευή. Έτσι δεν χρειάζεται να διαθέτει κάποιο συγκεκριμένο πρόγραμμα εγκατεστημένο ο μηχανισμός ασφαλείας, παρά μόνο το OpenC2.

Με αυτό τον τρόπο μπορεί να προγραμματίσει τον μηχανισμό ασφαλείας με σκοπό να αμυνθεί, εφόσον έχει λάβει τις κατάλληλες πληροφορίες. Συνοψίζοντας, το πρότυπο OpenC2 απλοποιεί την ενδοεπικοινωνία των συστατικών και την κάνει πολύ πιο αποτελεσματική.

4.1 Σκοπός

Η ανίχνευση σε πραγματικό χρόνο και η μετρίαση των απειλών σε οποιοδήποτε επίπεδο αλλά και σύστημα, απαιτεί την ενσωμάτωση, τον συγχρονισμό και την αυτοματοποίηση των μηχανισμών ασφαλείας. Απαιτεί από τους μηχανισμούς να είναι σε θέση να «ανιχνεύσουν» μία επίθεση, να μπορούν σε πραγματικό χρόνο να αποφασίσουν, να είναι σε θέση να ανταπεξέλθουν για να την περιορίσουν, ώστε να μην παρατηρηθεί καμία απώλεια στο πληροφοριακό σύστημα. Είναι γεγονός πως δεν μπορεί κανένα σύστημα να αποφύγει όλες τις πιθανές επιθέσεις, επομένων πρέπει να είναι σε θέση να «αμυνθεί».

Μία βασική προϋπόθεση για να γνωρίζει ένα πληροφοριακό σύστημα την κατάστασή του, είναι να ενημερώνεται συνεχώς από διαφόρων ειδών αισθητήρες, να επεξεργάζεται τις πληροφορίες αυτές, με σκοπό να ενημερώνεται σε πραγματικό χρόνο. Ανάλογα με τα δεδομένα που θα λαμβάνει από τα συστατικά του δικτύου, θα μπορεί να «αντιληφθεί» εάν βρίσκεται σε κίνδυνο ή όχι. Εφόσον το κρίνει κατάλληλο, θα πρέπει να είναι σε θέση αποφασίσει εάν θα χρειαστεί να πραγματοποιήσει κάποιου είδους απόκριση.

Όλα τα παραπάνω για να πραγματοποιηθούν και να τα μπορούμε να βασιστούμε πάνω στους μηχανισμούς ασφαλείας, απαιτείται μία εγγυημένη υποδομή επικοινωνιών για τη διασφάλιση ενός τυπικού μέσου επικοινωνίας για όλες τις τεχνολογίες που εμπλέκονται. Επίσης θα πρέπει να εκτελούνται οι εντολές εγκαίρως από επικυρωμένους και εξουσιοδοτημένους φορείς.

Το OpenC2 πραγματεύεται στο τμήμα απόκρισης (δηλαδή στο τμήμα που ενεργεί) στον κυβερνοχώρο. Η αλληλουχία ενεργειών που δημιουργεί, αναφέρεται τόσο σε μέτρα που μπορούν να ληφθούν για την πρόληψη όσο και για την αντιμετώπιση των επιθέσεων. Η καθορισμένη γλώσσα (defined language of OpenC2) επιτρέπει τη σαφή επικοινωνία μεταξύ δύο μηχανών και δεν προϋποθέτει κάποιο συγκεκριμένο πρωτόκολλο μεταφοράς. όσον αφορά την ολοκληρωμένη διαδικασία της συντονισμένης απόκρισης, το στάδιο της ανίχνευσης και της λογικής απόφασης, είναι εκτός του πεδίου εφαρμογής του OpenC2.

Ουσιαστικά δεν υπολογίζει από μόνο του το πότε και με ποιον τρόπο πρέπει να δράσει. Τα παραπάνω είναι καθορισμένα από άλλα συστατικά ή διαφορετικούς μηχανισμούς ασφαλείας του πληροφοριακού συστήματος, που θα αναλυθούν παρακάτω στο 5.1.1. Είναι αναγκαίο όμως για την αρχικοποίηση της συντονισμένης απόκρισης να έχουν καθοριστεί αλλά και να λειτουργούν ορθά σε πραγματικό

χρόνο. Έτσι, το OpenC2 υποθέτει ότι το υπόλοιπο πληροφοριακό σύστημα είναι ενεργό και λειτουργικό.

4.2 Ανάλυση OpenC2

4.2.1 Ορολογία

4.2.1.1 Δράση (Action)

Μια μεμονωμένη εργασία που έχει καθοριστεί από τον παραγωγό OpenC2 να εκτελεστεί. Πιο συγκεκριμένα είναι μια εντολή από έναν παραγωγό σε έναν καταναλωτή και εκτελείται από έναν ενεργοποιητή [22].

Ονομασία Δράσης	**Περιγραφή**
Allow	επιτρέπει την πρόσβαση ή την εκτέλεση από έναν στόχο
Cancel	ακυρώνει μια ενέργεια που είχε εκδοθεί προηγουμένως
Contain	απομονώνει ένα αρχείο, μία διεργασία ή μια οντότητα με σκοπό να μην μπορεί να προσπελαστεί ή να είναι αναγνώσιμο από διεργασίες κ.α.
Delete	διαγράφει μία οντότητα (π.χ. δεδομένα, αρχεία)
Deny	αποτρέπει την ολοκλήρωση ενός συγκεκριμένου συμβάντος ή ενέργειας
detonate	εκτέλεση και παρατήρηση μίας συμπεριφοράς ενός στόχου σε ένα απομονωμένο περιβάλλον
investigate	αναθέτει στον παραλήπτη να συγκεντρώσει συγκεκριμένες πληροφορίες που σχετίζονται με κάποιο γεγονός ή συμβάν ασφάλειας
Locate	εντοπίζει ένα αντικείμενο με φυσικό, λογικό ή λειτουργικό τρόπο
Query	πραγματοποιείται ένα αίτημα για πληροφορίες
Restart	σταματά και ξεκινά ένα σύστημα ή μία δραστηριότητα
Scan	πραγματοποιείται συστηματική εξέταση κάποιας πτυχής της οντότητας
Start	εκκίνηση διεργασίας, εφαρμογής, συστήματος ή δραστηριότητας
Stop	σταματά ένα σύστημα ή τερματίζεται μία διεργασία

Πίνακας 1 "OpenC2 Actions"

Στον παραπάνω πίνακα φαίνονται οι δράσεις που μπορούν να σταλούν.

4.2.1.2 Στόχος (Target)

Το αντικείμενο της δράσης. Μια ενέργεια εκτελείται σε έναν στόχο. Ένας Στόχος μπορεί να είναι μία διεύθυνση IP , ένα αρχείο, μία διεργασία ή ακόμα και μία συσκευή.

OpenC2 Targets (Defined in the Language Specification)		
Name	Type	Description
artifact	Artifact	an array of bytes representing a file-like object or a link to that object
command	String	a reference to a previously issued Command
device	Device	the properties of a hardware device
domain_name	Domain-Name	a network domain name
email_addr	Email-Addr	a single email address
features	Features	a set of items used with the query action to determine an actuator's capabilities
file	File	properties of a file
ipv4_connection	IPv4-Connection	a 5-tuple of source and destination IPv4 address ranges, source and destination ports, and protocol
ipv6_connection	IPv6-Connection	a 5-tuple of source and destination IPv6 address ranges, source and destination ports, and protocol
mac_addr	MAC-Addr	a Media Access Control (MAC) address - EUI-48 or EUI-64
process	Process	common properties of an instance of a computer program as executed on an operating system

Πίνακας 2 "OpenC2 ονοματολογία και ορισμοί των Στόχων"

4.2.1.3 Κατηγόρημα (Argument)

Μια ιδιότητα που προσδίδει μία επιπλέον πληροφορία σχετικά με τον τρόπο, το πότε, και τέλος το πού να εκτελεστεί μια εντολή. Τα κατηγορήματα εξαρτώνται από το περιεχόμενο.

4.2.1.4 Προσδιοριστής (Specifier)

Μια ιδιότητα ή πεδίο που προσδιορίζει έναν στόχο ή έναν ενεργοποιητή σε κάποιο επίπεδο ακρίβειας.

4.2.1.5 Ενεργοποιητής (Actuator)

Η συνάρτηση που εκτελείται από τον καταναλωτή που εκτελεί την εντολή. Μία τέτοια διαδικασία θα μπορούσε να ήταν π.χ. Stateless ή state-full packet filtering.

4.2.1.6 Προφίλ ενεργοποιητή (Actuator Profile)

Ένα υποσύνολο της γλώσσας OpenC2 που σχετίζεται με μία συγκεκριμένη συνάρτηση του κυβερνοχώρου.

4.2.1.7 Εντολή (Command)

Είναι ένα μήνυμα που ορίζεται από ένα ζεύγος στόχου – δράσης. Ενδεχομένως να προστεθούν κάποια ορίσματα και οι αντίστοιχοι προσδιοριστές. Αποστέλλεται από έναν παραγωγό, λαμβάνεται από έναν καταναλωτή και εκτελείται από έναν ενεργοποιητή.

```
{
  "action": <action type>,
  "target": {
    <target type>: {
      <target specifiers as key:value
         pairs>}},
  "args": {
    <general command arguments as key:
       value pairs>,
    <actuator type>: {
      <actuator-relevant command
         arguments as key:value pairs>}},
  "actuator": {
    <actuator type>: {
      <actuator specifiers as key:value
         pairs>}}
}
```

Εικόνα 12 "Παράδειγμα Εντολής OpenC2"

Στην παραπάνω εικόνα παρατηρούμε το συντακτικό που χρησιμοποιείται σε μορφή JSON. Απεικονίζεται η αφαιρετική δομή μιας εντολής.

4.2.1.8 Απόκριση (Response)

Ένα μήνυμα από έναν καταναλωτή σε έναν παραγωγό που αναγνωρίζει μια εντολή ή επιστρέφει τους πόρους ή την κατάσταση που ζητήθηκε βάσει μιας εντολής που είχε ληφθεί προηγουμένως.

```
{
    "status": 200,
    "results": {
        "versions": ["1.0"]
    }
}
```

4.2.1.9 Μήνυμα (Message)

Ένα μήνυμα είναι ένα σύνολο στοιχείων τα οποία μεταφέρονται μεταξύ παραγωγών και καταναλωτών. Για να διασφαλιστεί η διαλειτουργικότητα μεταξύ των συσκευών, όλες οι προδιαγραφές μεταφοράς πρέπει να ορίζουν σαφώς τον τρόπο με τον οποίο τα στοιχεία μηνυμάτων παρουσιάζονται στο πρωτόκολλο ασφαλούς μεταφοράς. Μπορούμε να δούμε αναλυτικά τα στοιχεία των μηνυμάτων στον παρακάτω πίνακα.

Name	Type	Description
content		Message body as specified by content_type and msg_type.
content_type	String	Media Type that identifies the format of the content, including major version. Incompatible content formats must have different content_types. Content_type **application/openc2** identifies content defined by OpenC2 language specification versions 1.x, i.e., all versions that are compatible with version 1.0.
msg_type	Message-Type	The type of OpenC2 Message.
status	Status-Code	Populated with a numeric status code in Responses.
request_id	String	A unique identifier created by the Producer and copied by Consumer into all Responses, in order to support reference to a particular Command, transaction, or event chain.
created	Date-Time	Creation date/time of the content.
from	String	Authenticated identifier of the creator of or authority for execution of a message.
to	ArrayOf(String)	Authenticated identifier(s) of the authorized recipient(s) of a message.

Πίνακας 3 "Τα συχνότερα στοιχεία ενός Μηνύματος"

Τα πεδία Content, content_type, and msg_type είναι υποχρεωτικά για όλα τα μηνύματα. Η εσωτερική αναπαράσταση ενός μηνύματος δεν επηρεάζει τη διαλειτουργικότητα και επομένως είναι εκτός του πεδίου εφαρμογής του OpenC2.

4.2.1.10 Παραγωγός (Producer)

Ένας παραγωγός είναι μια οντότητα η οποία δημιουργεί και στέλνει εντολές. Αυτή η οντότητα μπορεί να έχει δυνατότητες παραγωγού αλλά και καταναλωτή. Ένας παραγωγός μπορεί να ζητήσει πληροφορίες οι οποίες σχετίζονται με τις εκδόσεις της γλώσσας του OpenC2, τα προφίλ ενεργοποιητή και τα ζευγάρια action-target τα οποία υποστηρίζονται από ένα συγκεκριμένο προφίλ.

```
{
    "action": "query",
    "target": {
        "features": ["versions", "profiles", "rate_limit"]
    }
}
```

Εικόνα 13 "Παράδειγμα Ερώτησης OpenC2"

Στην παραπάνω εικόνα φαίνεται ένα ερώτημα το οποίο παράγεται από κάποιον παραγωγό.

```
{
   "status":200,
   "results":{
      "versions":["1.0"],
      "profiles":["slpf-1.0"],
      "pairs":{
         "allow":["ipv6_net"],
         "deny":["ipv6_net"],
         "query":["features"],
         "delete":["slpf:rule_number"],
         "update":["file"]}
   }
}
```

Εικόνα 14 "Παράδειγμα Απόκρισης OpenC2"

Στην παραπάνω εικόνα φαίνεται ένα κομμάτι του μηνύματος, στο οποίο ο καταναλωτής αποκρίνεται στον παραγωγό. Όπως βλέπουμε του απαντάει με τον κωδικό μηνύματος “status” : 200, το οποίο σημαίνει πως το ερώτημα είχε την κατάλληλη δομή και η επικοινωνία ήταν επιτυχής. Επίσης τα δεδομένα που του αποστέλλει είναι η έκδοση η οποία τρέχει, τι είδους προφίλ ενεργοποιητή τρέχει και τι ζευγάρια στόχου-δράσης υποστηρίζονται.

4.2.1.11 Καταναλωτής (Consumer)

Μια οντότητα που λαμβάνει και πιθανώς ενεργεί βάσει κάποιον εντολών που του έχουν αποσταλεί. Όπως προαναφέρθηκε και παραπάνω, είναι σημαντικό να σημειωθεί ότι μια οντότητα μπορεί να έχει καθήκοντα καταναλωτή αλλά και παραγωγού ταυτόχρονα. Αυτό επιτυγχάνεται με το να έχει πολλαπλά προφίλ ενεργοποιητή. Είναι πιθανό για έναν καταναλωτή να υποστηρίζει ένα περιορισμένο εύρος ζευγών στόχου-δράσης ενός προφίλ, αλλά εξακολουθεί να αξιώνει τη συμμόρφωση με το προφίλ ενεργοποιητή.

4.2.2 Επικοινωνία

Η καθορισμένη γλώσσα επιτρέπει τη σαφή επικοινωνία machine-to-machine και είναι αβέβαιο για οποιοδήποτε συγκεκριμένο πρωτόκολλο μεταφοράς ή μεταφοράς, και εφαρμογή διασφάλισης πληροφοριών.

4.2.2.1 Προδιαγραφή Γλώσσας (language specification)

Παρέχει τη σημασιολογία για τα βασικά στοιχεία της γλώσσας, τη δομή εντολών και απαντήσεων και καθορίζει τις κατάλληλες συνθέσεις και τύπους δεδομένων για τα στοιχεία γλώσσας που αντιπροσωπεύουν την εντολή και την απόκριση. Η γλώσσα OpenC2 [7] καθορίζει ένα σύνολο αφαιρετικών και ατομικών δράσεων στην κυβερνοασφάλεια, επιτρέποντας τη διαλειτουργικότητα μεταξύ των συστημάτων άμυνας στον κυβερνοχώρο ανεξάρτητα από οποιεσδήποτε άλλες πτυχές των υποκείμενων εφαρμογών.

4.2.2.2 Προφίλ Ενεργοποιητή (actuator profile)

Είναι ένα από τα σημαντικότερα συστατικά του OpenC2. Είναι ο πιο κρίσιμος παράγοντας για να συνεχίσει το πρωτόκολλο να χρησιμοποιείται έως ότου διαφοροποιηθεί τελείως η αρχιτεκτονική των δικτύων. Η επέκταση και ο καθορισμός υποομάδων της γλώσσας OpenC2 είναι τα βασικά καθήκοντα του προφίλ

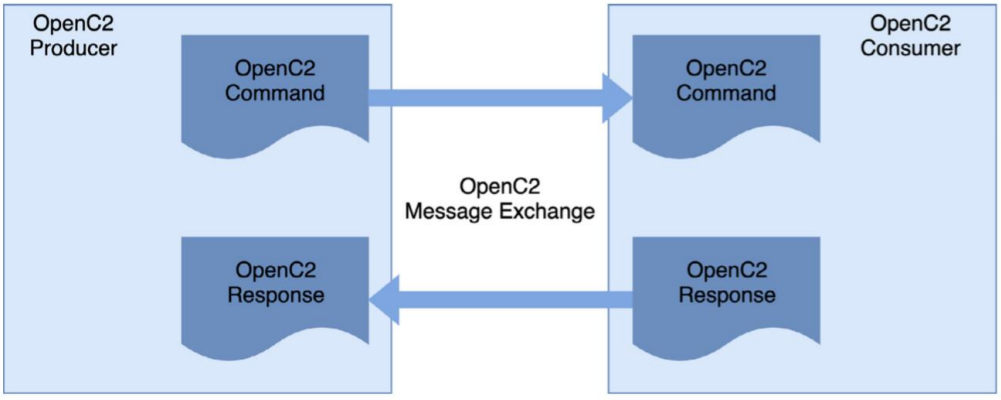
ενεργοποιητή που σχετίζονται με συγκεκριμένες λειτουργίες άμυνας στον κυβερνοχώρο. Ορίζει τις κατάλληλες απαιτήσεις συμμόρφωσης και τις συστάσεις, για την ενεργοποίηση της διαλειτουργικότητας μεταξύ διαφορετικών τεχνολογιών. Η μεγαλύτερη πρόκληση σε αυτό το κομμάτι, είναι ότι ανεξαρτήτως κατασκευαστή, πρέπει να φτιαχτεί ένα προφίλ, το οποίο θα λειτουργεί αφαιρετικά.

4.2.2.3 Προδιαγραφή Μεταφοράς (transfer specification)

Χρησιμοποιεί τα υπάρχοντα πρωτόκολλα και πρότυπα για την κωδικοποίηση και την επικοινωνία των μηνυμάτων OpenC2 με ασφάλεια. Καθορίζει την κωδικοποίηση με την οποία πρέπει να επικοινωνούν τα συστατικά.

4.2.2.4 Παράδειγματα Επικοινωνίας

Ένα πολύ απλοποιημένο παράδειγμα επικοινωνίας του πρωτοκόλλου OpenC2 συμπεριλαμβάνει ένα μήνυμα, όπως αυτό ορίζεται στο στην ορολογία του πρωτοκόλλου, μεταξύ ενός παραγωγού και ενός καταναλωτή. Αυτή η βασική επικοινωνία που έχουν μεταξύ τους μπορεί να οριστεί με την αποστολή μιας εντολής από τον παραγωγό και με την απόκριση του καταναλωτή προς τον παραγωγό.

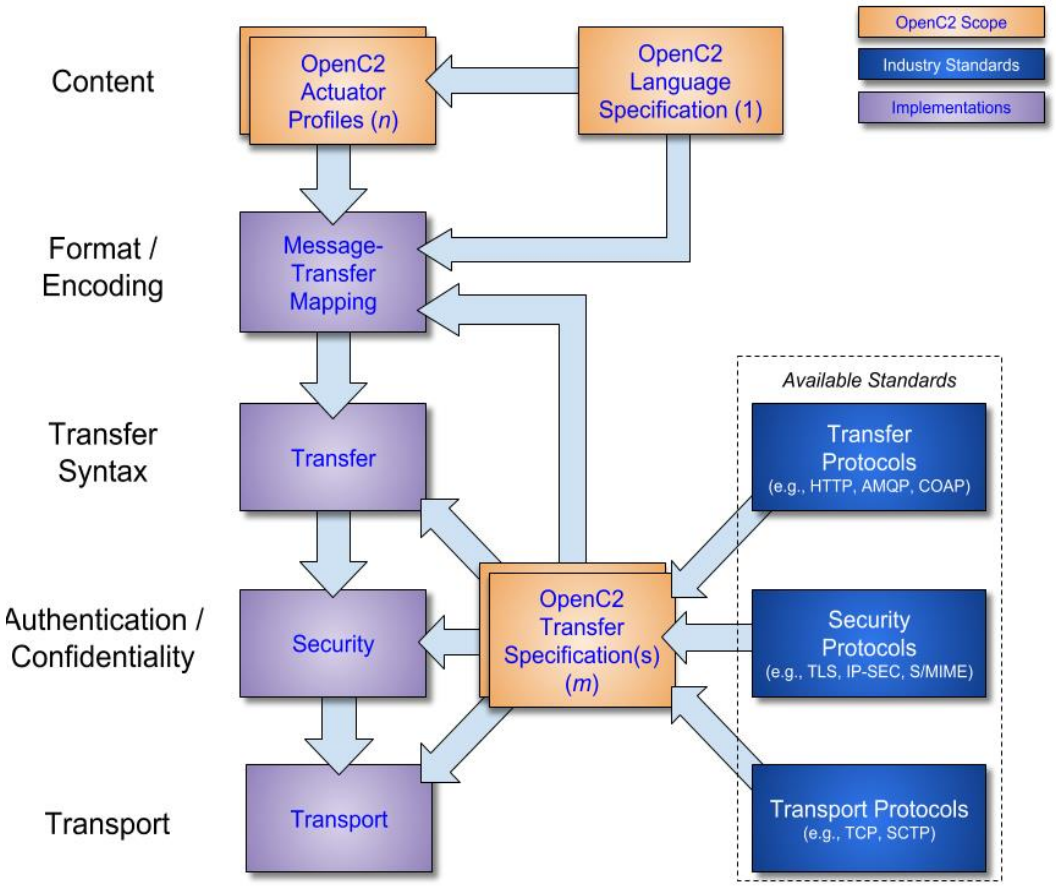


Εικόνα 15 "Παράδειγμα Επικοινωνίας OpenC2"

Για την αποσαφήνιση των ορισμών από’ το προηγούμενο κεφάλαιο, τα συστατικά μιας εντολής είναι μία δράση, η οποία καθορίζει τι πρέπει να γίνει. Ένας στόχος συγκεκριμενοποιεί το που ενεργεί η εντολή. Εφόσον υπάρχει ενεργοποιητής, είναι αυτός που πραγματοποιεί την εντολή. Μία ενέργεια σε συνδυασμό με έναν στόχο περιγράφουν με ακρίβεια μία πλήρη εντολή. Τα στοιχεία μιας απόκρισης είναι ένας αριθμητικός κωδικός κατάστασης, μια προαιρετική συμβολοσειρά κειμένου κατάστασης και κάποια αποτελέσματα. Η μορφή των αποτελεσμάτων, εάν περιλαμβάνεται, εξαρτάται από τον τύπο της απόκρισης που μεταφέρεται.

4.2.3 Αρχιτεκτονική

Οι υλοποιήσεις του OpenC2 ενσωματώνουν τις σχετικές προδιαγραφές OpenC2 με βιομηχανικές προδιαγραφές, πρωτόκολλα και πρότυπα. Το Σχήμα παρακάτω απεικονίζει τις σχέσεις μεταξύ των προδιαγραφών OpenC2 και τις σχέσεις τους με άλλα βιομηχανικά πρότυπα και συγκεκριμένες για το σύστημα υλοποιήσεις του OpenC2. Είναι σημαντικό να σημειωθεί ότι η στρώση του επιπέδου εφαρμογής στο διάγραμμα είναι πλασματική και δεν προορίζεται να αποκλείσει οποιαδήποτε συγκεκριμένη προσέγγιση για την εφαρμογή της απαιτούμενης λειτουργικότητας π.χ. η χρήση μιας λειτουργίας υπογραφής μηνύματος επιπέδου εφαρμογής για να παρέχει τον έλεγχο της ταυτότητας και την ακεραιότητα της πηγής του μηνύματος.



Εικόνα 16 "Αρχιτεκτονική Σουίτας που χρησιμοποιείται και το OpenC2"

Το επίπεδο ασφαλούς μεταφοράς παρέχει μια διαδρομή επικοινωνίας μεταξύ του παραγωγού και του καταναλωτή. Το OpenC2 μπορεί να επικαλυφθεί σε οποιοδήποτε πρότυπο πρωτοκόλλου μεταφοράς.

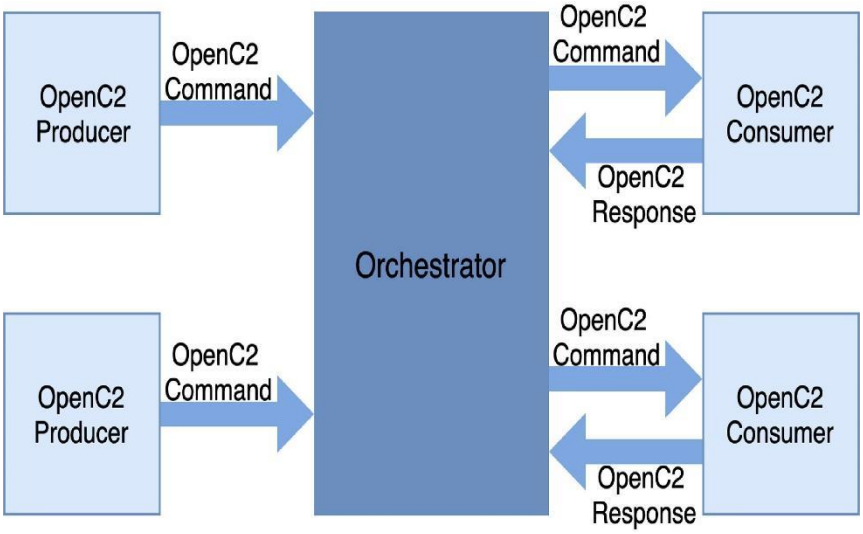
Το επίπεδο μηνυμάτων παρέχει έναν μηχανισμό μεταφοράς. Μια προδιαγραφή μεταφοράς χαρτογραφεί στοιχεία πρωτοκόλλου σχετικά με την μεταφορά σε ένα σύνολο ανεξάρτητων από μεταφορά στοιχείων μηνυμάτων που αποτελούνται από περιεχόμενο και σχετικά metadata.

Το επίπεδο κοινού περιεχομένου καθορίζει τη δομή των εντολών και των απαντήσεων και ένα σύνολο στοιχείων κοινής γλώσσας που χρησιμοποιούνται για την κατασκευή τους. Στο επίπεδο αυτό χρησιμοποιείται η προδιαγραφή γλώσσας.

Το επίπεδο περιεχομένου για συγκεκριμένες λειτουργίες ορίζει τα γλωσσικά στοιχεία που χρησιμοποιούνται για την υποστήριξη μιας συγκεκριμένης λειτουργίας υπεράσπισης στον κυβερνοχώρο. Ένα προφίλ ενεργοποιητή καθορίζει τις απαιτήσεις συμμόρφωσης εφαρμογής για αυτήν τη λειτουργία. Οι παραγωγοί και οι καταναλωτές θα υποστηρίξουν ένα ή περισσότερα προφίλ. Στο επίπεδο αυτό πρωταγωνιστικό ρόλο έχουν οι ενεργοποιητές προφίλ.

4.2.3.1 Αρχιτεκτονική χωρίς διαμεσολαβητή

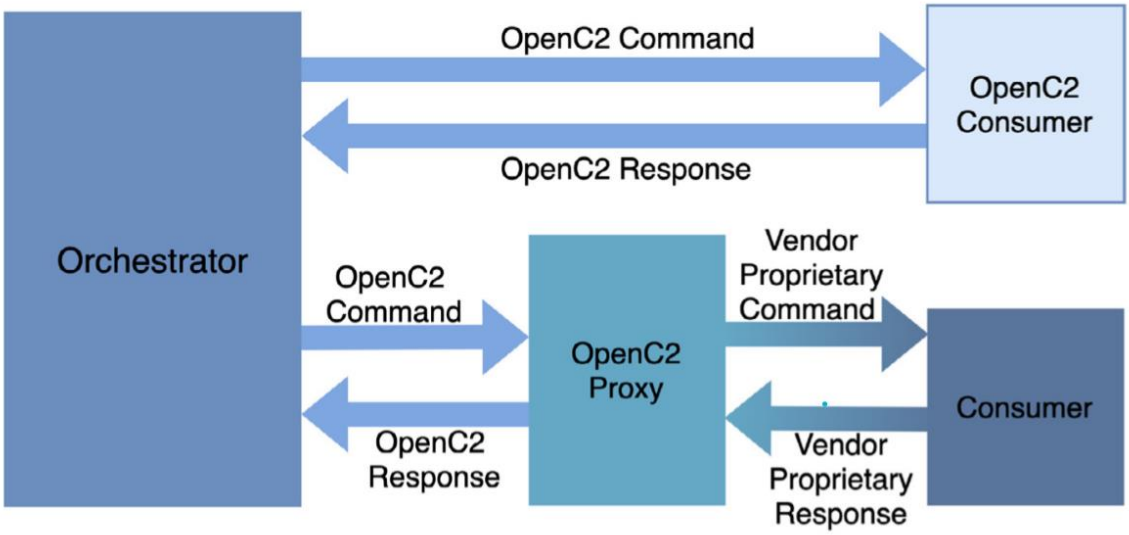
Όπως παρατηρείται στο παρακάτω διάγραμμα, το συστατικό ενορχηστρωτής (orchestrator) έχει οργανωτικά καθήκοντα σχετικά με τις αποστολές που έχει να διεκπεραιώσει το OpenC2. Ο ενορχηστρωτής διαδίδει ή προωθεί τις OpenC2 εντολές σε ενεργοποιητές προφίλ. Παρόλα αυτά, ένας ενορχηστρωτής πρέπει να είναι ικανός να στείλει, να λάβει και να κρατήσει ιστορικό τις εντολές που εκπέμπει αλλά και τις αποκρίσεις αυτών. Επιπροσθέτως πρέπει να είναι σε θέση αυθεντικοποιεί συσκευές και λειτουργίες.



Εικόνα 17 "Παράδειγμα Αρχιτεκτονικής χωρίς διαμεσολαβητή"

4.3.2.2 Αρχιτεκτονική με διαμεσολαβητή

Σε αυτή την περίπτωση αρχιτεκτονικής, ο ενορχηστρωτής επικοινωνεί με τον καταναλωτή μέσω διαμεσολαβητή. Αυτή η είδους αρχιτεκτονική είναι πιο ασφαλής από την προηγούμενη. Επίσης υπάρχει πιο ομαλή κίνηση των πακέτων στο δίκτυο. Ουσιαστικά, ο ενορχηστρωτής δεν γνωρίζει ακριβώς ποια εντολή πρέπει να στείλει στον καταναλωτή. Όμως του στέλνει ποια ενέργεια θα ήθελε να διεκπεραιώσει. Αυτό το αφαιρετικό αίτημα, το ικανοποιεί ο διαμεσολαβητής μέσω της μετάφρασης του αιτήματος. Ύστερα στέλνει στον καταναλωτή την ακριβή εντολή.



Εικόνα 18 "Παράδειγμα Αρχιτεκτονικής με διαμεσολαβητή"

4.3 Σφαιρική Εικόνα

Η γλώσσα καθορίζει δύο βασικές δομές ωφέλιμου φορτίου. Η πρώτη είναι η πράξη (command) και η δεύτερη είναι ο έλεγχος (control). Η πράξη ορίζεται ως την οδηγία που δίνεται από ένα σύστημα, το οποίο σύστημα ονομάζεται ως παραγωγός (producer). Η πράξη κατευθύνεται προς ένα ή περισσότερα συστήματα, τα οποία ονομάζονται ως καταναλωτές (consumers).

Η εντολή τύπου «πράξη» (command) αποτελείται από τέσσερα συστατικά, εκ των οποίων τα δύο μόνο είναι υποχρεωτικά να καθοριστούν, τα άλλα δύο είναι προαιρετικά. Τα δύο πρώτα συστατικά είναι αλληλένδετα, τα οποία ονομάζονται

δράση-στόχος (action-target). Ουσιαστικά καθορίζουν το τι ενέργεια λαμβάνει χώρα εκείνη τη στιγμή και σε ποιον στόχο. Τα άλλα δύο συστατικά είναι τα ορίσματα εντολών και οι προσδιοριστές ενεργοποιητή. Τα ορίσματα εντολών μπορούν να χρησιμοποιηθούν για να μεταδώσουν την ανάγκη για επιβεβαίωση ή πρόσθετες πληροφορίες για την κατάσταση μιας εντολής, σχετικά με την εκτέλεσή της.

Οι προσδιοριστές ενεργοποιητή προσδιορίζουν περαιτέρω έναν ενεργοποιητή σε κάποιο επίπεδο ακριβείας, όπως έναν συγκεκριμένο ενεργοποιητή (specific actuator) ή μια ομάδα ενεργοποιητών. Μια εντολή μπορεί επίσης να περιέχει ένα προαιρετικό αναγνωριστικό εντολών για παρακολούθηση και αναφορά σχετικών εντολών και απαντήσεων. Η απόκριση είναι ένα μήνυμα που αποστέλλεται από τον παραλήπτη μιας εντολής. Τα μηνύματα απόκρισης παρέχουν επιβεβαίωση, κατάσταση, αποτελέσματα ερωτήματος ή άλλες ζητούμενες πληροφορίες. Το λιγότερο που περιέχει μία απόκριση, είναι ένας κωδικός κατάστασης, για να δείξει το αποτέλεσμα της εκτέλεσης μιας εντολής.

Η γλώσσα OpenC2 καθορίζει ένα σύνολο αμερόληπτων αφαιρετικών και ατομικών δράσεων στον κυβερνοάμυνα, επιτρέποντας τη διαλειτουργικότητα μεταξύ των συστημάτων άμυνας στον κυβερνοχώρο ανεξάρτητα από οποιεσδήποτε άλλες πτυχές των υποκείμενων εφαρμογών.

Το δεύτερο χαρακτηριστικό της γλώσσας OpenC2 είναι ότι χαρακτηρίζεται από την επανάληψη και τη σταδιακή αλλαγή των σύντομων φράσεων, εστιάζοντας μόνο στις βασικές πληροφορίες που απαιτούνται για τη λήψη στοχευμένων δράσεων υπεράσπισης στον κυβερνοχώρο. Η γλώσσα έχει σχεδιαστεί για να παρέχει ελάχιστη επιβάρυνση στην επικοινωνία των μηνυμάτων OpenC2 και είναι κατάλληλη για περιορισμένο αριθμό περιβαλλόντων δικτύου.

Το πιο σημαντικό χαρακτηριστικό της γλώσσας είναι ο αφαιρετικός τρόπος λειτουργίας της. Οι εντολές και οι απαντήσεις του Openc2 ορίζονται αφηρημένα και μπορούν να κωδικοποιηθούν και να μεταφερθούν μέσω πολλαπλών σχημάτων όπως υπαγορεύονται από τις ανάγκες διαφορετικών περιβαλλόντων.

Η γλώσσα OpenC2 θα πρέπει να εξελιχθεί παράλληλα με τις τεχνολογίες άμυνας στον κυβερνοχώρο. Υποστηριζόμενη από τις προαναφερθείσες αρχές σχεδιασμού, το OpenC2 μπορεί να επεκταθεί για την εισαγωγή νέας λειτουργικότητας στον χώρο της κυβερνοάμυνας.

4.4 Συγκριτική Παρουσίαση Υλοποιήσεων

4.4.1 Stateless Packet Filter (SLPF)

Το «Stateless Packet Filter» (SLPF) [6] είναι ένας μηχανισμός επιβολής πολιτικής που περιορίζει ή επιτρέπει την κυκλοφορία με βάση στατικές τιμές, όπως διεύθυνση προέλευσης, διεύθυνση προορισμού ή / και αριθμούς θύρας. Ένα SLPF, δεν λαμβάνει υπόψη μοτίβα κυκλοφορίας, κατάσταση σύνδεσης, ροές δεδομένων, εφαρμογές ή πληροφορίες ωφέλιμου φορτίου. Το εύρος αυτού του προφίλ περιορίζεται στο φιλτράρισμα πακέτων χωρίς κατάσταση που αναφέρεται εδώ ως SLPF.

Αυτό το προφίλ ενεργοποιητή καθορίζει το σύνολο ενεργειών, στόχων, προσδιοριστών και κατηγορημάτων (arguments) που ενσωματώνει τη λειτουργικότητα SLPF με το σύνολο εντολών Open Command and Control (OpenC2). Μέσω αυτού του συνόλου εντολών, οι ενορχηστρωτές ασφάλειας στον κυβερνοχώρο μπορούν να αποκτήσουν ορατότητα και να παρέχουν τον έλεγχο της λειτουργικότητας SLPF ανεξάρτητα από την παρουσία της λειτουργίας SLPF.

Όλα τα εξαρτήματα, συσκευές και συστήματα που παρέχουν λειτουργικότητα SLPF θα εφαρμόσουν τις ενέργειες OpenC2, τους στόχους, τους προσδιοριστές και τα επιχειρήματα που προσδιορίζονται ως απαιτούνται σε αυτό το έγγραφο. Οι ενέργειες που είναι εφαρμόσιμες, αλλά δεν απαιτούνται απαραίτητα, για το SLPF θα αναγνωρίζονται ως προαιρετικές. Συμμορφώνεται σύμφωνα με την έκδοση γλώσσας 1.0 OpenC2.

Τα συστήματα άμυνας στον κυβερνοχώρο που χρησιμοποιούν το OpenC2 ενδέχεται να απαιτούν τα ακόλουθα στοιχεία για την εφαρμογή του προφίλ SLPF:

- Παραγωγοί OpenC2: Συσκευές που στέλνουν εντολές, λαμβάνουν απαντήσεις και διαχειρίζονται την εκτέλεση εντολών που περιλαμβάνουν έναν ή περισσότερους SLPF ή άλλους ενεργοποιητές με δυνατότητα SLPF. Ο OpenC2 Producer χρειάζεται μια εκ των προτέρων γνώση σχετικά με τις εντολές που ο Ενεργοποιητής μπορεί να επεξεργαστεί και να εκτελέσει, επομένως πρέπει να κατανοήσει τα προφίλ για οποιαδήποτε συσκευή που σκοπεύει να δώσει εντολή

- OpenC2 Consumers: Συσκευές ή παρουσίες που παρέχουν λειτουργίες φιλτραρίσματος πακέτων χωρίς κατάσταση. Συνήθως αυτοί είναι ενεργοποιητές που εκτελούν τη λειτουργία υπεράσπισης στον κυβερνοχώρο, αλλά θα μπορούσαν να είναι ενορχηστρωτές (δηλαδή, μια συσκευή ή μια παρουσία που προωθεί τις εντολές στον ενεργοποιητή)
- Αν και τα στοιχεία, οι συσκευές, τα συστήματα ή / και οι παρουσίες στον κυβερνοάμυνα μπορούν να εφαρμόσουν πολλά προφίλ ενεργοποιητή, ένα συγκεκριμένο μήνυμα OpenC2 ενδέχεται να αναφέρεται το πολύ σε ένα μόνο προφίλ ενεργοποιητή. Το πεδίο εφαρμογής αυτού του εγγράφου περιορίζεται στο SLPF.

4.4.2 OpenC2 Messages μέσω HTTPS

Αυτή η υλοποίηση καθορίζει τη χρήση του Hypertext Transfer Protocol (HTTP) μαζί με Transport Layer Security (TLS) ως μηχανισμού μεταφοράς για τα μηνύματα OpenC2. Αυτή η στρώση HTTP / TLS αναφέρεται συνήθως ως HTTPS. Όπως περιγράφεται στο [5], το HTTP έχει γίνει ένα κοινό "υπόστρωμα" για μεταφορά πληροφοριών για άλλα πρωτόκολλα σε επίπεδο εφαρμογής. Η ευρεία διαθεσιμότητα του HTTP το καθιστά μια χρήσιμη επιλογή για μεταφορά μηνυμάτων OpenC2 για υποστήριξη πρωτοτύπων, δοκιμές διαλειτουργικότητας μεταξύ μηχανισμών διαφορετικών κατασκευαστών και για λειτουργική χρήση σε περιβάλλοντα όπου μπορούν να παρέχονται κατάλληλες προστασίες ασφαλείας. Παρομοίως, το TLS είναι ένα ώριμο και ευρέως χρησιμοποιούμενο πρωτόκολλο για την εξασφάλιση μεταφοράς πληροφοριών σε περιβάλλοντα δικτύου TCP / IP.

Αυτή η προδιαγραφή μεταφοράς του OpenC2 over HTTPS είναι κατάλληλη για λειτουργικά περιβάλλοντα όπου:

A. Η συνδεσιμότητα μεταξύ OpenC2 παραγωγών και OpenC2 καταναλωτών είναι:
    1. Πολύ διαθέσιμο, με σπάνιες διακοπές δικτύου.
    2. Υπάρχει επαρκές εύρος ζώνης στο οποίο δεν παρατηρείται σημαντική καθυστέρηση μηνυμάτων ή πτώση πακέτων.

B. Η διαπραγμάτευση μιας ζώνης για σύνδεση που ξεκινάει είτε από τον Παραγωγό είτε από τον Καταναλωτή είναι δυνατή χωρίς να απαιτείται δίκτυο σηματοδότησης εκτός ζώνης.

C. Η επιβάρυνση του HTTPS είναι αποδεκτή (π.χ. πολλαπλές ανταλλαγές εντολών / απόκρισης OpenC2 μπορούν να περάσουν μέσω μιας σύνδεσης HTTPS).

4.4.3 Σύγκριση Υλοποιήσεων σε μηχανισμούς ασφαλείας

Τα firewall που είναι στατικά και συμπεριφέρονται κυρίως με μία πηγή δεδομένων που έχουν, είναι αρκετά περιορισμένα ως προς την λειτουργία τους. Το μεγαλύτερο θετικό πλεονέκτημα είναι, πως εάν ανιχνεύσουν έναν κίνδυνο, τότε δεν του επιτρέπουν να επικοινωνήσει με το εσωτερικό δίκτυο. Τα firewall τα οποία διαθέτουν μηχανισμούς κρυπτογράφησης και αποκρυπτογράφησης, μπορούν να παρακολουθήσουν ολόκληρη την διαδρομή του πακέτου. Γι' αυτό το λόγο τα μηνύματα OpenC2 μέσω HTTPS θα μπορούν εύκολα να εξελιχθούν έναντι των στατικών firewall.

ΚΕΦΑΛΑΙΟ 5

Μελέτη περίπτωσης χρήσης (Case Study)

ΕΙΣΑΓΩΓΗ

Για την περίπτωση χρήσης και την πειραματική εφαρμογή ενσωμάτωσης του OpenC2 σε self-healing συστήματα, επέλεξα να δείξω ένα πολύ βασικό σενάριο χρήσης. Σε αυτό το σενάριο καλείται μία πολύ βασική μορφή ενός συστήματος self-healing, να εξετάσει εάν υπάρχει κάποιος κίνδυνος για το σύστημα. Παρακάτω θα εξηγήσω πιο αναλυτικά, τον ορισμό ενός self-healing συστήματος. Ο OpenC2 μεσολαβητής θα δεχτεί εντολή αυτό-ίασης ώστε να προστατέψει την αρχιτεκτονική. Με αυτό το μοτίβο μπορούν να αναπτυχθούν και να χρησιμοποιηθούν σε πολλές τεχνολογίες.

Το μηχάνημα που χρησιμοποιήθηκε για να επιτευχθούν οι παραπάνω προϋποθέσεις στήθηκε σε Linux Ubuntu 20.04 LTS, είχε την ισχύ ενός πυρήνα επεξεργαστή Intel i5 και μνήμη RAM 4 GB.

5.1 Εγκατάσταση των υποσυστημάτων

5.1.1 OpenC2

Χρειάζεται να έχουμε έκδοση Python πάνω από 3.7.0 . Επίσης πρέπει να εγκαταστήσουμε το εργαλείο pip3. [4]

```
$ pip3 install virtualenv
$ mkdir test_yuuki
$ cd test_yuuki
$ virtualenv venv
$ source venv/bin/activate
$ git clone https://github.com/oasis-open/openc2-yuuki.git
$ pip3 install ./openc2-yuuki
```

Σε αυτό το στάδιο, εάν η εγκατάσταση έχει γίνει όπως έχει καθοριστεί, τότε θα πρέπει να μας δείξει αυτό το μήνυμα :

Εικόνα Σεναρίου Χρήσης 1 "Μήνυμα επιτυχούς εγκατάστασης"

Εφόσον η εγκατάσταση πέτυχε, δεν μας ενδιαφέρει το μήνυμα προσοχής.

Για την εκκίνηση του διαμεσολαβητή OpenC2:

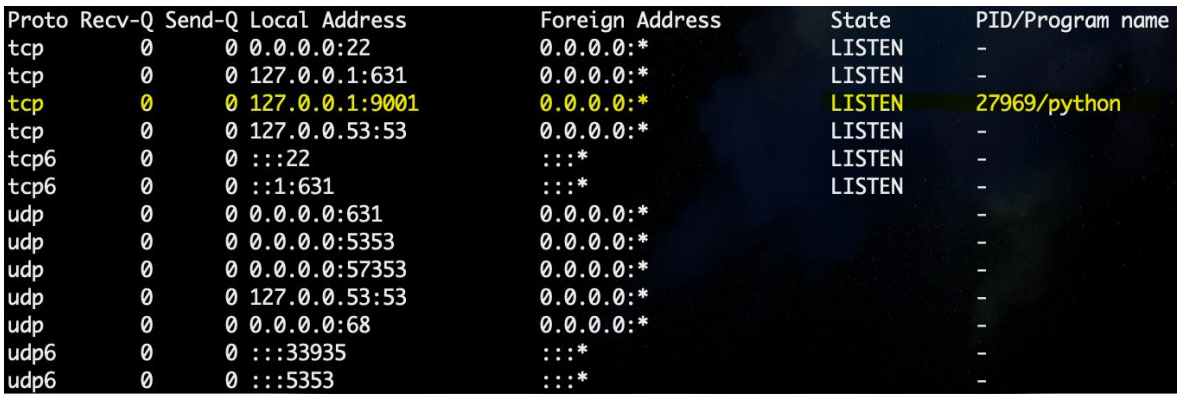
```
yuuki.consumer > /dev/null 2>&1 &
```

Θα ενεργοποιηθεί με τις τιμές που είναι προκαθορισμένες, εφόσον δεν βάζουμε κάποια μεταβλητή.

Για να επαληθεύσουμε ότι ξεκίνησε, πρέπει να είναι ενεργοποιημένη μια λειτουργία python στην θύρα 9001. Αυτό μπορούμε να το ελέγξουμε πολύ εύκολα πατώντας την εντολή:

```
netstat -tulpn
```

Η έξοδος της εντολής θα μοιάζει κάπως έτσι. Στην στήλη Local Address, στο αριστερό μέρος από την [:] βρίσκεται η IP διεύθυνση. Αυτό που μας ενδιαφέρει όμως είναι το δεξί μέρος. Π.χ. 127.0.0.1:9001 Αυτό που ψάχνουμε είναι μία ip που περιμένει μία σύνδεση στην θύρα 9001.



```
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:9001          0.0.0.0:*               LISTEN      27969/python
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      -
tcp6       0      0 :::22                   :::*                    LISTEN      -
tcp6       0      0 ::1:631                 :::*                    LISTEN      -
udp        0      0 0.0.0.0:631             0.0.0.0:*                           -
udp        0      0 0.0.0.0:5353            0.0.0.0:*                           -
udp        0      0 0.0.0.0:57353           0.0.0.0:*                           -
udp        0      0 127.0.0.53:53           0.0.0.0:*                           -
udp        0      0 0.0.0.0:68              0.0.0.0:*                           -
udp6       0      0 :::33935                :::*                                -
udp6       0      0 :::5353                 :::*                                -
```

Εικόνα Σεναρίου Χρήσης 2 «Έλεγχος Ανοιχτής Πόρτας Διαμεσολαβητή»

5.2 Αλληλουχία ενεργειών για τα σενάρια χρήσης

```
1   server_needs_maintenance_restart:restart
2   network_ddos:firewall_rule
3   malware:disinfect
4   suspicious_file:system_scan
5   disk_failure:defrag
6   overheat:poweroff
7   ssh_bruteforce:blockuser
8   trojan:netclean
9   badblocks_ondisk:instant_notify
```

Εικόνα Σεναρίου Χρήσης 3 "Το αρχείο που χρησιμοποιείται ως βάση δεδομένων"

5.2.1 Self-Healing

Ο όρος self-healing ορίζεται στα πληροφοριακά συστήματα, ως η διεργασία των συστημάτων η οποία αναγνωρίζει ότι υπάρχει κάποιο σφάλμα στην λειτουργία του συστήματος και το επαναφέρει σε αρχική θέση.

Ως self-healing ορίζεται μία διεργασία, η οποία όταν τρέχει προσπαθεί να αποφασίσει αν χρήζει για ίαση κάποιο υπολογιστικό σύστημα. Αποτελείται από τέσσερα μέρη:

- Δέχεται είσοδο δεδομένων
- Παίρνει την απόφαση εάν χρειάζεται να δράσει
- Πραγματοποίηση των κατάλληλων εντολών
- Καταγράφει σε ένα αρχείο τις εντολές που εκτελεί και το αποτέλεσμά τους

```
#!bin/bash

if test $# -ne 1 ; then
    echo "`date` Prepei na eisagetai mono 1 orisma me tin ektelesi tou script. Exit status 1" >> self_healing.log
    exit 1
elif test ! -r $1 ; then
    echo "`date` Den diavastike to arxeio $1. Eite den uparxei eite den exei ta katallila. Exit status 2" >> self_healing.log
fi

#pragmatopoieitai extract to victim_ip kai to incident_description apo to json arxeio
victim_ip=$(cat $1 | egrep -o 'victim_ip":"[^\,]+"' | cut -d':' -f2 | sed 's/"//g')
description=$(cat $1 | egrep -o 'description":"[^\,]+"' | cut -d':' -f2 | sed 's/"//g')

#pragmatopoieitai extract to action pou prepei na ginei. Analoga me to incident-description poy pairnei to action apo to database
action=$(egrep $description ./database | cut -d':' -f2)

#edo kaleitai to openc2 {pithana actions}
#restart
#firewall_rule
#disinfect
#system_scan
#defrag
#poweroff
#blockuser
#netclean
#instant_notify

#Ean to flow einai fusiologiko, grafetai sto logfile.
echo "`date` Epitixis leitourgia tou script. {$victim_ip , $description , $action}. Exit status 0" >> self_healing.log
exit 0

```

Εικόνα Σεναρίου Χρήσης 4 "bash script selfhealing.sh"

Στον παραπάνω κώδικα πραγματοποιείται ο έλεγχος των ορισμάτων που δίνονται κατά την εκτέλεση του κώδικα. Θα πρέπει να δίνεται μόνο ένα όρισμα και αυτό είναι το αρχείο που περιέχει την διεύθυνση του θύματος και την περιγραφή του γεγονότος. Στην συνέχεια καλείται η εντολή του OpenC2. Τέλος καταγράφεται στο αρχείο το αποτέλεσμα της πράξης.

5.3 Σενάριο ίασης - εξαναγκασμένης επανεκκίνησης

Σε αυτή την περίπτωση, ο στόχος πρέπει να επανεκκινηθεί. Έχει κριθεί από το self-healing σύστημα, πως ο στόχος με την IP διεύθυνση 192.168.1.14 και την θύρα 9001, είναι ευάλωτος και χρήζει επανεκκίνηση λόγω συντήρησης της υποδομής. Με την επανεκκίνηση θα αλλάξει το αρχείο παραμετροποίησης και δεν θα είναι πλέον ευάλωτος.

Η ροή ξεκινάει με την εισαγωγή δεδομένων στο self-healing σύστημα. Εισάγεται ένα JSON αρχείο το οποίο περιέχει 2 μεταβλητές.

Εικόνα Σεναρίου Χρήσης 5 “Αρχείο Format JSON“

Στο πεδίο victim_ip καθορίζεται ο στόχος που πρέπει να «ιατρευθεί».

Στο πεδίο description υπάρχει ο λόγος που κρίνεται απαραίτητο για να «ιατρευθεί».

Το self-healing σύστημα αναζητεί στην βάση δεδομένων του, στο συγκεκριμένο σενάριο είναι το αρχείο database, προσπαθεί να αντιστοιχίσει την περιγραφή με εντολή OpenC2. Έπειτα εκτελεί την εντολή η οποία έχει καθοριστεί και παράγεται το ανάλογο αποτέλεσμα. Το αποτέλεσμα της εντολής καταγράφεται στο αρχείο self_healing.log .

Παράδειγμα ροής:

```
$ bash self_healing.sh incident.json
```

Για να σιγουρευτούμε ότι πραγματοποιήθηκε η επανεκκίνηση στον εξυπηρετητή, αρκεί να δούμε το αρχείο που κρατάει τις αποκρίσεις από τον εξυπηρετητή.

Εικόνα Σεναρίου Χρήσης 6 "Αποτέλεσμα Αρχείου Εγγραφής”

Στο πρώτο μέρος αναγράφεται η ημερομηνία και ένα μήνυμα επιτυχίας της εκτέλεσης του κώδικα και η επιτυχής έξοδος από το σενάριο.

ΚΕΦΑΛΑΙΟ 6

ΕΙΣΑΓΩΓΗ

Σε αυτό το κεφάλαιο αναφέρονται τα συμπεράσματα που προέκυψαν από το πρακτικό κομμάτι της διπλωματικής. Αναλύονται περισσότερο κάποιο όροι που δεν έγινε αποσαφήνιση νωρίτερα. Επίσης, συνοψίζοντας σε αυτό το κεφάλαιο θα γίνει μια σύντομη αναφορά των μελλοντικών επεκτάσεων που μπορούν να υλοποιηθούν γύρω από τη μελέτη περίπτωσης.

Συμπεράσματα

Με την ραγδαία αύξηση των συσκευών που συνδέονται στο διαδίκτυο, δημιουργείται η ανάγκη για μία ασφαλέστερη και ολοκληρωμένη υποδομή δικτύου. Για να αυξηθεί η αποδοτικότητα και η αποτελεσματικότητα του δικτύου, το δίκτυο πρέπει να παρακολουθείται, να ενημερώνονται εγκαίρως τα λογισμικά των συσκευών του. Είναι σημαντικό επίσης να πραγματοποιείται αναβάθμιση των συσκευών που συμβάλλουν στην ασφάλεια. Η άμεση απόκριση σε περιστατικό «επίθεσης», η ορθή αντιμετώπιση και η συντήρηση της αρχιτεκτονικής του δικτύου, παίζουν καθοριστικό ρόλο για την ομαλή λειτουργία του δικτύου. Η χρήση του OpenC2 θα καταφέρει να λύσει το χάσμα της επικοινωνίας των προϊόντων διαφορετικών κατασκευαστών. Μπορεί να προάγει ένα λειτουργικό, έμπιστο και αποτελεσματικό δίκτυο.

6.1 Μελλοντικές επεκτάσεις

6.1.1 Self-Healing

Ένα σύστημα self-healing [9] εμπορικής χρήσης, διαθέτει πολλά περισσότερα συστατικά μέρη. Είναι πολύ πιο περίπλοκο από αυτό του σεναρίου. Συνήθως διαθέτουν προσωπική βάση, η οποία ανανεώνεται τακτικά με τις πιο καινούργιες ευπάθειες που δημιουργούνται και ανακαλύπτονται ανά τον κόσμο. Θα μπορούσε να διαθέτει μία ολοκληρωμένη βάση δεδομένων τύπου mysql. Θα

υπήρχαν πίνακες που θα καθόριζαν τον τύπο του προβλήματος, την μέθοδο αντιμετώπισης, δηλαδή την εντολή που θα πρέπει να ενεργοποιηθεί για να εκτελεστεί το OpenC2. Επίσης θα μπορούσε να συνδεθεί με ένα σύστημα το οποίο θα ενημερώνεται αυτόματα με τις καινούριες ευπάθειες που ανακαλύπτονται ανά τον κόσμο. Έτσι θα είναι ενήμερο, όσον αφορά τους κινδύνους που μπορεί να χρειάζεται να γνωρίζει και να αντιμετωπίζει το πληροφοριακό σύστημα. Ωστόσο όλα αυτά αποτελούν ολοκληρωμένες λύσεις συστημάτων εντοπισμού ευπαθειών και αντιμετώπισής τους.

6.1.2 Καινοτόμες χρήσεις του OpenC2

Η επέκταση του OpenC2 καθορίζεται από τους προγραμματιστές του. Εφόσον θέλουν και μπορούν να συνεργαστούν οι κατασκευαστές των μηχανισμών ασφαλείας, μπορούν να προγραμματίσουν με την κατάλληλη καθοδήγηση οποιονδήποτε τρόπο αντιμετώπισης. Μεγάλοι κατασκευαστές firewall και antivirus έχουν προγραμματίσει λειτουργίες και τρόπους αντιμετώπισης σεναρίων επιθέσεων και κινδύνων. Το μόνο που χρειάζεται είναι να μπορεί να προσδιοριστεί ο τρόπος λειτουργίας των συστατικών του δικτύου. Είναι σημαντικό να διαθέτουν συγκεκριμένο τρόπο λειτουργίας, με στόχο να προγραμματιστεί ανάλογα και στην γλώσσα του OpenC2.

Αν και το OpenC2 έχει λίγα χρόνια που δημιουργήθηκε, παρατηρείται μία ραγδαία άνοδος της χρήσης του. Η μεγαλύτερη πρόκληση που έχει να αντιμετωπίσει είναι η ορθή ενημέρωση των εταιριών. Το πρότυπο αυτό αναφέρεται σε οποιαδήποτε κατηγορία επιχειρήσεων, η οποία διαθέτει πληροφοριακό σύστημα, ένα δίκτυο το οποίο χρήζει παρακολούθηση και τους απαραίτητους μηχανισμούς ασφαλείας. Είναι ένα πολύ δυνατό εργαλείο για την κυβερνό-ασφάλεια και πρέπει οπωσδήποτε να χρησιμοποιηθεί και να επεκταθεί.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Internet Control Message Protocol. Retrieved September 18, 2020, from https://tools.ietf.org/html/rfc792
2. User Datagram Protocol. Retrieved September 18, 2020, from https://tools.ietf.org/html/rfc768
3. Introduction to Community-based SNMPv2. Retrieved September 18, 2020, from https://tools.ietf.org/html/rfc1901
4. OpenC2. Retrieved September 18, 2020, from https://github.com/OpenC2-org
5. Specification for Transfer of OpenC2 Messages via HTTPS Version 1.0. Retrieved September 18, 2020, from https://docs.oasis-open.org/openc2/open-impl-https/v1.0/cs01/open-impl-https-v1.0-cs01.html
6. OpenC2-org. OpenC2-org/openc2-apsc-stateless-packet-filter. Retrieved September 18, 2020, from https://github.com/OpenC2-org/openc2-apsc-stateless-packet-filter
7. OpenC2 Specifications. Retrieved September 18, 2020, from https://openc2.org/specifications
8. Hunt, C. (2002). *TCP/IP network administration*. Sebastopol, Calif: O'Reilly.
9. Spyros, A., Rantos, K., Papanikolaou, A., & Ilioudis, C. (2020). An Innovative Self-Healing Approach with STIX Data Utilisation. *Proceedings of the 17th International Joint Conference on E-Business and Telecommunications*. doi:10.5220/0009893306450651

10. A Simple Network Management Protocol (SNMP). Retrieved September 18, 2020, from https://www.ietf.org/rfc/rfc1157.txt
11. Textual Conventions for SMIv2. Retrieved September 18, 2020, from https://tools.ietf.org/html/rfc2579
12. Conformance Statements for SMIv2. Retrieved September 18, 2020, from https://tools.ietf.org/html/rfc2580
13. Networks, A. (2019, April 23). Network Basics: What Is SNMP and How Does It Work? Retrieved September 18, 2020, from https://www.auvik.com/franklyit/blog/network-basics-what-is-snmp/
14. Sloan, J. D. (2001). *Network troubleshooting tools*. Sebastopol (California): O'Reilly.
15. Mauro, D. R., & Schmidt, K. J. (2005). *Essential SNMP*. Beijing: O'Reilly.
16. Amirthalingam, K., & Moorhead, R. SNMP-an overview of its merits and demerits. *Proceedings of the Twenty-Seventh Southeastern Symposium on System Theory*. doi:10.1109/ssst.1995.390588
17. Antón-Haro, C., & Dohler, M. (2015). *Machine-to-machine (M2M) communications architecture, performance and applications*. Amsterdam: Elsevier.
18. Doc: RFC 1098: Simple Network Management Protocol (SNMP). Retrieved September 18, 2020, from https://www.hjp.at/doc/rfc/rfc1098.html
19. Stallings, W. (2009). *SNMP, SNMPv. 2, SNMPv. 3 and RMON 1 and 2*. Boston, MA: Addison-Wesley.

20. Doc: RFC 2571: An Architecture for Describing SNMP Management Frameworks. Retrieved September 18, 2020, from https://www.hjp.at/doc/rfc/rfc2571.html
21. Kumar, S. (2007). Smurf-based Distributed Denial of Service (DDoS) Attack Amplification in Internet. *Second International Conference on Internet Monitoring and Protection (ICIMP 2007)*. doi:10.1109/icimp.2007.42
22. Mavroeidis, V., & Brule, J. (2020). A nonproprietary language for the command and control of cyber defenses – OpenC2. *Computers & Security, 97*, 101999. doi:10.1016/j.cose.2020.101999
23. Structure of Management Information Version 2 (SMIv2). Retrieved September 22, 2020, from https://tools.ietf.org/search/rfc2578
24. Vaisla, K. S. Analyzing of Zero Day Attack and its Identification Techniques. Retrieved September 22, 2020, from https://www.researchgate.net/publication/260489192_Analyzing_of_Zero_Day_Attack_and_its_Identification_Techniques
25. Φαρμάκης, Κ. SNMP protocol for communication and management of network devices. Retrieved from http://digilib.teiemt.gr/jspui/bitstream/123456789/2048/1/012014049.pdf