



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

Σχολή Ψηφιακής Τεχνολογίας
Τμήμα Πληροφορικής και Τηλεματικής

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

Τηλεπικοινωνιακά Δίκτυα και συστήματα Τηλεματικής

Διπλωματική Εργασία

Παρακολούθηση του δικτύου της Ομήρου μέσω NMS

Δρούλιας Αντώνιος

Αθήνα, 2020



ΧΑΡΟΚΟΠΕΙΟ ΠΑΝΕΠΙΣΤΗΜΙΟ

Σχολή Ψηφιακής Τεχνολογίας
Τμήμα Πληροφορικής και Τηλεματικής

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

Τηλεπικοινωνιακά Δίκτυα και συστήματα Τηλεματικής

Τριμελής Εξεταστική Επιτροπή

Δαλάκας Βασίλειος
ΕΔΙΠ, Τμήμα Πληροφορικής και Τηλεματικής, Χαροκόπειο Πανεπιστήμιο

Καμαλάκης Θωμάς
Αναπληρωτής καθηγητής στο Τμήμα Πληροφορικής και Τηλεματικής του
Χαροκοπείου Πανεπιστημίου Αθηνών

Μιχαλακέλης Χρήστος
Επίκουρος καθηγητής στο Τμήμα Πληροφορικής και Τηλεματικής του
Χαροκοπείου Πανεπιστημίου Αθηνών

Ο Δρούλιας Αντώνιος δηλώνω υπεύθυνα ότι:

- 1)** Είμαι ο κάτοχος των πνευματικών δικαιωμάτων της πρωτότυπης αυτής εργασίας και από όσο γνωρίζω η εργασία μου δε συκοφαντεί πρόσωπα, ούτε προσβάλει τα πνευματικά δικαιώματα τρίτων.
- 2)** Αποδέχομαι ότι η ΒΚΠ μπορεί, χωρίς να αλλάξει το περιεχόμενο της εργασίας μου, να τη διαθέσει σε ηλεκτρονική μορφή μέσα από τη ψηφιακή Βιβλιοθήκη της, να την αντιγράψει σε οποιοδήποτε μέσο ή/και σε οποιοδήποτε μορφότυπο καθώς και να κρατά περισσότερα από ένα αντίγραφα για λόγους συντήρησης και ασφάλειας.

Στην οικογένεια μου

Πρόλογος

Η παρούσα διπλωματική εργασία υλοποιήθηκε στα πλαίσια του Προγράμματος Μεταπτυχιακών Σπουδών του τμήματος Πληροφορικής και Τηλεματικής στο Χαροκόπειο Πανεπιστήμιο Αθηνών από τον Οκτώβριο του 2018 έως τον Σεπτέμβριο του 2020.

Ειδικότερα θα ήθελα να ευχαριστήσω τον καθηγητή μου Δρ. Βασίλειο Δαλάκα, μέλος ΕΔΙΠ του τμήματος Πληροφορικής και Τηλεματικής του Χαροκόπειου Πανεπιστημίου ο οποίος με εμπιστεύτηκε και με καθοδήγησε καθ' όλη την διάρκεια της προσπάθειας μου.

Έχοντας εργαστεί στον τομέα των δικτύων ως Network Engineer επέλεξα το συγκεκριμένο θέμα διπλωματικής εργασίας με σκοπό να εξελίξω περαιτέρω τις γνώσεις μου τόσο σε θεωρητικό επίπεδο όσο και σε πρακτικό ειδικά στο πεδίο έρευνας των δικτύων και των εφαρμογών τους.

ΠΕΡΙΕΧΟΜΕΝΑ

Πρόλογος.....	5
Περίληψη	8
Summary	10
Κεφάλαιο 1.....	12
1.1 Εισαγωγή.....	12
1.2 Η έννοια του συστήματος παρακολούθησης δικτύου.....	12
1.3 Η λειτουργία του Network Monitoring System (NMS).	12
1.4 Τι είναι το SNMP.....	13
Κεφάλαιο 2.....	16
2.1 Λογισμικό ανοιχτού κώδικα.	16
2.2 Επιλογή λογισμικού.....	19
2.3 Icinga Web 2.	19
2.3 Χρήση Εικονικών μηχανών (Virtual Machines).....	21
2.4 Διαδικασία εγκατάστασης του Icinga Web 2.....	21
Κεφάλαιο 3.....	34
3.1 Αναφορά ως προς τα Plugins των Icinga2 & Nagios.	34
3.2 Τι είναι τα Plugins ή “Επεκτάσεις”.	34
3.3 Πως λειτουργεί ένα Plugin.....	34
3.4 Διαδικασία καταγραφής δικτυακών συσκευών με το icinga2-autod	36
3.5 Διαδικασία επιλογής του Plugin.	38
3.6 Τι είναι το check_snmp_int	39
3.7 check_snmp_int Configuration.....	41
Κεφάλαιο 4.....	44
4.1 Τι είναι το check_iftraffic Plugin.	44
4.2 Εγκατάσταση check_iftraffic στο Icinga2.....	44
4.3 check_iftraffic configuration.....	51
Κεφάλαιο 5.....	53
Συμπεράσματα	57
Βιβλιογραφία	58
Παράρτημα	59

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1. Η αρχιτεκτονική του SNMP.....	174
Εικόνα 2. Το monitoring system του Nagios.....	187
Εικόνα 3. Το monitoring system του Icinga Web2.....	18
Εικόνα 4. Αρχικό περιβάλλον διεπαφής.....	244
Εικόνα 5. Modules.....	254
Εικόνα 6. Οι ρυθμίσεις λογισμικού	265
Εικόνα 7. Οι ρυθμίσεις του συστήματος μετά της αλλαγές.	266
Εικόνα 8. Ορισμός του Authentucation Type.	276
Εικόνα 9. Ορισμός του Database Resource	277
Εικόνα 10. Δημιουργία λογαριασμού root.....	287
Εικόνα 11. Δήλωση ονόματος Database.....	288
Εικόνα 12. Δημιουργία Administrator	298
Εικόνα 13. Ρύθμιση παραμέτρων.....	299
Εικόνα 14. Επιτυχής επιβεβαίωση ρυθμίσεων.....	309
Εικόνα 15. Δήλωση επιλογής δεδομένων	30
Εικόνα 16. Δήλωση της ido mysql database.....	31
Εικόνα 17. Οθόνη Command Transport.	31
Εικόνα 18. Οθόνη Monitoring Security.....	32
Εικόνα 19. Γενική επισκόπηση των ρυθμίσεων και εγκατάσταση	32
Εικόνα 20. Αρχική οθόνη εισαγωγής στη εφαρμογή.....	33
Εικόνα 21. Αρχική οθόνη εισαγωγής στη εφαρμογή.....	35
Εικόνα 22. Διάγραμμα λειτουργίας Nagios plugin.	35
Εικόνα 23. Διάγραμμα λειτουργίας Icinga plugin.....	35
Εικόνα 24. Το Exchange Plugins του Nagios.	36
Εικόνα 25. Το περιεχόμενο του icinga2-autod	Error! Bookmark not defined. 36
Εικόνα 26. Αποτέλεσμα εντολής icinga2-autod	37
Εικόνα 27. Καταγεγραμμένες δικτυακές συσκευές.....	Error! Bookmark not defined. 38
Εικόνα 28. Λίστα των plugins	Error! Bookmark not defined. 39
Εικόνα 29. Output από το TEST_ROUTER.....	Error! Bookmark not defined. 40
Εικόνα 30. Αποτέλεσμα της εντολής check_snmp_int	Error! Bookmark not defined.
Εικόνα 31. Output Router με απενεργοποιημένη την Fa3	41
Εικόνα 32. Output από το TEST_ROUTER.....	46
Εικόνα 33. Εισαγωγική φόρμα στο Icinga Web 2.	53
Εικόνα 34. Τα services που είναι σε λειτουργία στο Icinga Web 2.....	54
Εικόνα 35. Οι Hosts στο Icinga Web 2.	54
Εικόνα 36. Τα services για τον TEST_ROUTER	55
Εικόνα 37. Το Tactical Overview του Icinga Web 2.....	55
Εικόνα 38. Τα incidents του Icinga Web 2.	56
Εικόνα 39. Test_Router Host.....	56

Περίληψη

Αντικείμενο της διπλωματικής εργασίας αυτής είναι η εγκατάσταση και η παραμετροποίηση του NMS (network management system) του Icinga 2 και icingaWeb 2 για έλεγχο της λειτουργίας της Κεντρικής Υπηρεσίας Πιστοποίησης Χρηστών του πανεπιστημίου.

Ο κύριος στόχος της εργασίας αυτής είναι η δημιουργία δυο εφαρμογών Plugin ή “Επεκτάσεων” για το Icinga 2 ώστε να επιτηρούμε και να εντοπίζουμε πρώτον αν κάποια θύρα του δικτυακού μας εξοπλισμού είναι ενεργή ή ανενεργή και στην συνέχεια να μπορούμε να βλέπουμε την κίνηση της εκάστοτε πόρτας την συγκεκριμένη στιγμή.

Με τον τρόπο αυτό εκτιμάται πως κάθε στιγμή ο διαχειριστής του δικτύου θα μπορεί να βλέπει σε πραγματικό χρόνο αν κάποια θύρα απενεργοποιείται ή αν η κίνηση της δημιουργεί πρόβλημα στο σύνολο του δικτύου.

Στο κεφάλαιο 1 γίνεται αναφορά στην έννοια του συστήματος παρακολούθησης και στην λειτουργία του NMS (Network Monitoring System).

Στο κεφάλαιο 2 γίνεται αναφορά στο λογισμικό ανοικτού κώδικα και στην επιλογή λογισμικού και ειδικότερα στην εγκατάσταση του icinga web2

Στο κεφάλαιο 3 γίνεται αναφορά στα Plugins των Nagios και Icinga2 στην λειτουργία τους και στην διαδικασία επιλογής των Plugins. Γίνεται περιγραφή της καταγραφής των δικτυακών συσκευών και περιγραφή του check_snmp_int plugin.

Στο κεφάλαιο 4 γίνεται περιγραφή του check_iftraffic plugin της εγκατάστασης του αλλά και περιγράφονται τα threshold τα οποία θα ορίσουμε στην εκάστοτε θύρα του δικτυακού εξοπλισμού.

Στο κεφάλαιο 5 αναφερόμαστε στο περιβάλλον του Icinga Web 2 με τα Plugins εγκατεστημένα και τα αποτελέσματα που μας δίνουν κατά την διάρκεια λειτουργίας τους.

Κατά την διάρκεια της διπλωματικής μου εργασίας συνάντησα αρκετές προκλήσεις τόσο στο θεωρητικό κομμάτι όσο και στο τεχνικό, κάτι που την έκανε απαιτητική αλλά και ενδιαφέρουσα όλο περισσότερο .

Λέξεις κλειδιά

Icinga2, IcingaWeb2, VirtualBox, Debian 9, check_snmp_int, check_itraffic, Network Monitoring System (NMS)

Summary

The subject of this thesis is the installation and configuration of Icinga 2's NMS (network management system) and icingaweb2 to control the operation of the University's Central User Certification Service.

The main purpose of this work is to create two Plugins or "extensions" for Icinga 2 in order first all to monitor and detect the connectivity of the Ethernet ports of the network devices and secondly to check in real time the bandwidth and utilization of the Ethernet ports.

This way it is appreciated that at any time the network administrator will know if an Ethernet port has a problem and if the utilization of the port creates problem in the whole network.

Chapter 1 deals with the concept of monitoring system and the functioning of the NMS (Network Monitoring System).

Chapter 2 deals with open source software and software selection and in particular the installation of icinga web2

Chapter 3 deals with Plugins of Nagios and Icinga2 in their operation and selection process of Plugins. It also describes the procedure of scanning for network devices and the check_snmp_int plugin installation but also describes the errors that occurred until its operation in icinga2.

Chapter 4 describes the check_iftraffic plugin of its installation but describes the errors that occurred until its operation in icinga2.

In Chapter 5 we refer to the Icinga Web 2 environment after Plugins installed and the results, they give us during their operation.

During my thesis I encountered several challenges both in the theoretical and in the technical part which made it demanding but also more challenging in order to complete it.

Key Words

Icinga2, IcingaWeb2, VirtualBox, Debian 9, check_snmp_int, check_itraffic, Network Monitoring System (NMS)

Κεφάλαιο 1

1.1 Εισαγωγή.

Τα συστήματα παρακολούθησης δικτύων (Network Monitoring Systems) διαφέρουν από τα συστήματα ανίχνευσης εισβολών (IDSs) ή τα λεγόμενα συστήματα αποτροπής εισβολών (IPSs). Ενώ τα συστήματα αυτά ανιχνεύουν εισβολές και προλαμβάνουν εχθρικές ή μη εξουσιοδοτημένες δραστηριότητες από μη εξουσιοδοτημένους χρήστες, τα συστήματα παρακολούθησης δικτύων (NMS) μας επιτρέπουν να γνωρίζουμε σε ποια κατάσταση λειτουργίας βρίσκεται το δίκτυο μας κατά τη διάρκεια των συνηθισμένων εργασιών. Η παρακολούθηση του δικτύου επιτυγχάνεται με τη χρήση διαφόρων λογισμικών ή ένα συνδυασμό λύσεων hardware plug-and-play και συσκευών λογισμικού.

1.2 Η έννοια του συστήματος παρακολούθησης δικτύου.

Με τον όρο παρακολούθηση δικτύου (network monitoring) ενός συστήματος πληροφορικής εννοούμε την συνεχή παρακολούθηση ενός δικτύου υπολογιστών σχετικά με το πόσο αργό είναι ή όσον αφορά την έλλειψη ορισμένων στοιχείων. Ένα network monitoring system παρακολουθεί το δίκτυο για ενδεχόμενα προβλήματα που προκαλούνται από υπερφόρτωση, εφαρμογές που έχουν υποστεί κάποια ξαφνική διακοπή, Web servers ή προβλήματα που αφορούν τις συνδέσεις του δικτύου και άλλες συσκευές γενικότερα.

Σχεδόν κάθε είδους δίκτυο μπορεί να παρακολουθείται χωρίς να έχει σημασία αν είναι ασύρματο ή ενσύρματο, εταιρικό LAN ή ακόμα και VPN. Μπορούμε να παρακολουθούμε συσκευές με διαφορετικά λειτουργικά συστήματα, από BlackBerries και κινητά τηλέφωνα, σε servers, routers και switches. Με αυτά τα συστήματα παρακολούθησης μπορούν να εντοπιστούν συγκεκριμένες δραστηριότητες και να παραχθούν τα αποτελέσματα που θα μας επιτρέψουν να βρούμε τον τρόπο αντιμετώπισης τους.

1.3 Η λειτουργία του Network Monitoring System (NMS).

Ο τρόπος λειτουργίας των NMS (Network Monitoring System), αφορά τον έλεγχο σχετικά με την δραστηριότητα και την υγεία των εσωτερικών συστημάτων του δικτύου αποστέλλοντας ένα σήμα, το οποίο ονομάζεται ring, για τις διάφορες θύρες του συστήματος. Το σύστημα ελέγχου χρησιμοποιεί μια μεγάλη ποικιλία από διαστήματα ελέγχου, τα οποία ουσιαστικά είναι ο χρόνος μεταξύ των rings. Τα NMS έχουν τη δυνατότητα να ελέγχουν κάθε είδους πρωτοκόλλων δικτύου και ιδιαίτερα τα πρωτόκολλα διαδικτύου. Για παράδειγμα, οι υπηρεσίες παρακολούθησης ιστοσελίδων μπορούν να ελέγχουν τις σελίδες HTTP, HTTPS, SNMP, FTP, SMTP, POP3, IMAP, DNS, SSH, Telnet, SSL και TCP. Έτσι, όταν πρόκειται για web

servers, ένα πρόγραμμα παρακολούθησης του δικτύου μπορεί να στείλει ένα αίτημα HTTP σε έναν web server για να καθορίσει την κατάστασή του. Αλλά όταν πρόκειται για έναν e-mail server, το λογισμικό παρακολούθησης στέλνει περιοδικά ένα μήνυμα ελέγχου μέσω SMTP, το οποίο ανακτάται από το IMAP ή POP3. Με τον τρόπο αυτό, μπορεί να μιμηθεί τη διαδρομή ενός τυπικού μηνύματος και να ελέγχεται η υγεία του δικτύου και του Server μέσα από το οποίο περνά.

Όταν ένα εργαλείο παρακολούθησης δικτύου εντοπίζει κάποιο πρόβλημα σε ένα σύστημα, μέσω ενός αποτυχημένου αιτήματος κατάστασης για παράδειγμα και δεν υπάρχει η δυνατότητα δημιουργίας σύνδεσης, τότε αυτό οδηγεί στο λεγόμενο time out. Σε περιπτώσεις όπου υπάρχει μια αποτυχία αιτήματος κατάστασης, το σύστημα παρακολούθησης του δικτύου θα παράγει μια ενέργεια. Οι ενέργειες αυτές ποικίλλουν, καθώς μπορούν να ενημερώσουν τον διαχειριστή του δικτύου με μια ειδοποίηση, ένα SMS κειμένου, ένα μήνυμα τηλεειδοποίησης ή ένα email.

1.4 Τι είναι το SNMP

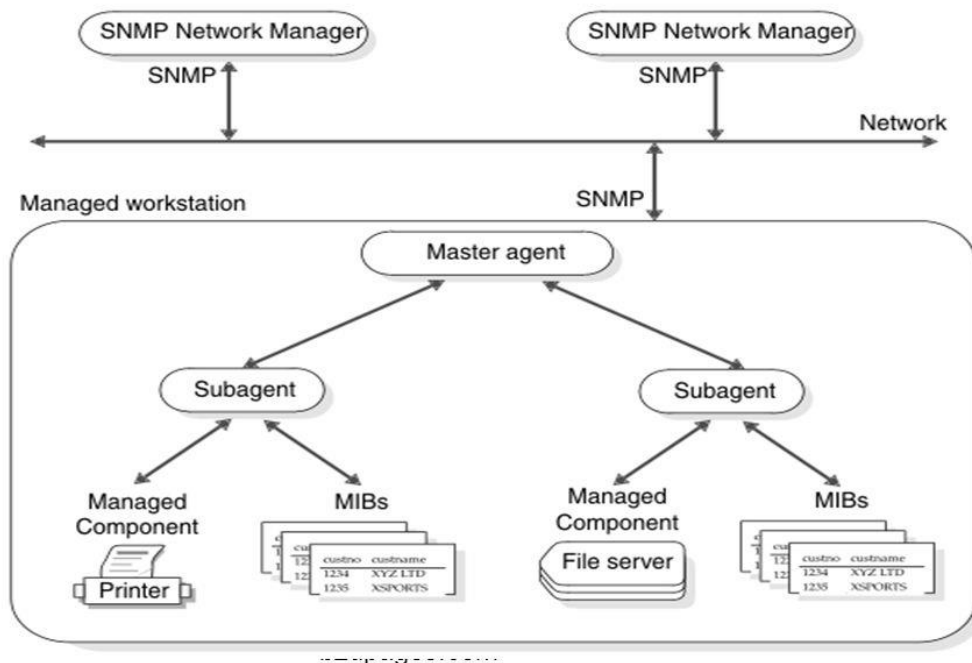
Το SNMP είναι ένα πρωτόκολλο του επιπέδου εφαρμογών του οποίο διευκολύνει την ανταλλαγή πληροφοριών διαχείρισης μεταξύ των συσκευών του δικτύου. Είναι μέρος του TCP/IP και επιτρέπει στους διαχειριστές να παρακολουθούν την απόδοσή του και να επιλύουν τα προβλήματα που εμφανίζονται. Με το SNMP, οι διαχειριστές μπορούν να διαχειρίζονται και τα ρυθμίζουν τις παραμέτρους των υπολογιστών του δικτύου από κάποιον κεντρικό υπολογιστή, χωρίς να χρειάζεται να εκτελούν λογισμικό διαχείρισης δικτύου. Μπορούν επίσης να χρησιμοποιήσουν το SNMP για την εποπτεία της απόδοσης του δικτύου, τον εντοπισμό προβλημάτων στο δίκτυο και για να δουν ποιος χρησιμοποιεί το δίκτυο και με ποιον τρόπο.

Ένα δίκτυο το οποίο διαχειρίζεται με SNMP έχει τρία βασικά στοιχεία: διαχειριζόμενες συσκευές, πράκτορες (agents) και συστήματα διαχείρισης δικτύου (Network Management Systems- NMS). Μια διαχειριζόμενη συσκευή είναι ένας κόμβος του δικτύου ο οποίος περιέχει ένα SNMP πράκτορα και βρίσκεται μέσα στο διαχειριζόμενο δίκτυο. Οι διαχειριζόμενες συσκευές συλλέγουν και αποθηκεύουν πληροφορίες και τις διαθέτουν στο σύστημα διαχείρισης του δικτύου με χρήση του SNMP. Τέτοιες συσκευές είναι οι δρομολογητές, οι γέφυρες, οι διακόπτες. Ένας πράκτορας έχει γνώση των τοπικών πληροφοριών διαχείρισης και τις μετατρέπει σε μορφή που είναι συμβατή με το SNMP. Στη

συνέχεια ένα σύστημα διαχείρισης δικτύου εκτελεί εφαρμογές οι οποίες παρακολουθούν και ελέγχουν τις διαχειριζόμενες συσκευές. Το σύστημα διαχείρισης του δικτύου προσφέρει τον κύριο όγκο των πόρων επεξεργασίας που απαιτούνται για τη διαχείριση

Η αρχιτεκτονική του SNMP φαίνεται στο παρακάτω σχήμα.

SNMP Architecture



Εικόνα 1. Η αρχιτεκτονική του SNMP.

Το πρωτόκολλο που χρησιμοποιείται ευρύτατα για τη διαχείριση σε TCP/IP δίκτυα είναι το Simple Network Management Protocol (SNMP). Πιο εξελιγμένες εκδόσεις του SNMP αποτελούν η SNMPv1 η SNMPv2 και η SNMPv3.

SNMPv1: Πρόκειται για την αρχική εφαρμογή του πρωτοκόλλου SNMP. Λειτουργεί σύμφωνα με τις προδιαγραφές του SMI σε πρωτόκολλα όπως το UDP, IP και χρησιμοποιείται ευρύτατα στο Internet για τη διαχείριση του δικτύου.

SNMPv2: Αποτελεί μια εξέλιξη του SNMPv1. Λειτουργεί επίσης με τις προδιαγραφές του SMI και προσφέρει κάποιες βελτιώσεις σε σχέση με το SNMP όπως να επεκτείνει το

πρότυπο ώστε να μπορεί να χρησιμοποιηθεί (εκτός από το TCP/IP) και σε πρωτόκολλα που ακολουθούν το πρότυπο OSI.

Ενώ το SNMPv1 υποστηρίζει μόνο IP διευθύνσεις των 32 bits, το SNMPv2 μπορεί να υποστηρίξει και άλλους τύπους διευθύνσεων. Οι μετρητές είναι μη αρνητικές μεταβλητές οι οποίες αυξάνονται μέχρι να πάρουν μέγιστη τιμή και μετά να ξαναγίνουν μηδέν. Στο SNMPv2 οι μετρητές μπορεί να είναι 32 ή 64 bit.

Στο SNMPv2 ορίζονται επίσης πληροφοριακά υποσύνολα, τα οποία καθορίζουν μια ομάδα από σχετικούς μεταξύ τους ορισμούς. Υπάρχουν τρία τέτοια υποσύνολα: τα MIB υποσύνολα, οι προτάσεις συμβατότητας και οι προτάσεις χωρητικότητας. Τα MIB υποσύνολα προσφέρουν έναν τρόπο για την περιγραφή των διαχειριζόμενων αντικειμένων. Οι προτάσεις συμβατότητας προσφέρουν έναν τρόπο περιγραφής των διαχειριζόμενων αντικειμένων τα οποία πρέπει να δημιουργηθούν έτσι ώστε να υπάρχει συμβιβασμός με τα υπάρχοντα πρότυπα. Και οι προτάσεις χωρητικότητας χρησιμοποιούνται για να υποδείξουν το ακριβές επίπεδο υποστήριξης που έχει ένας πράκτορας για μια MIB ομάδα. Ένα NMS μπορεί να ρυθμίσει τη συμπεριφορά του ως προς τους πράκτορες σύμφωνα με τις προτάσεις αυτές.

Κεφάλαιο 2

2.1 Λογισμικό ανοιχτού κώδικα.

Οι εφαρμογές που παρέχονται μέσω του συστήματος Linux και των λογισμικών ανοιχτού κώδικα είναι πάρα πολλές. Εκτελούν την διαχείριση αιτημάτων των πελατών/χρηστών με μεγάλη αποτελεσματικότητα. Ο χρήστης έχει την δυνατότητα να διαμορφώσει τον πηγαίο κώδικα βάσει των απαιτήσεών του χωρίς να επιβαρύνεται με οποιαδήποτε επιπλέον χρέωση.

Επίσης είναι το ίδιο αξιόπιστα με τα αντίστοιχα εμπορικά πακέτα με επιπλέον ανταγωνιστικά χαρακτηριστικά και είναι ευρέως διαδεδομένα καθώς χρησιμοποιούνται από πολλές επιχειρήσεις ανά τον κόσμο. Στη συνέχεια ακολουθεί μια σύντομη περιγραφή ορισμένων εκ των δημοφιλέστερων λειτουργικών όσον αφορά την παρακολούθηση των συσκευών, των υπηρεσιών, των θυρών, των πρωτοκόλλων και την ανάλυση της κυκλοφορίας του δικτύου.

Εδώ θα αναφέρουμε τα δυο κυριότερα λογισμικά που συνδέονται μεταξύ τους και είναι το Nagios και το Icinga web 2 για το λόγο ότι αποτελούν την κύρια πηγή της μελέτης μας.

Nagios

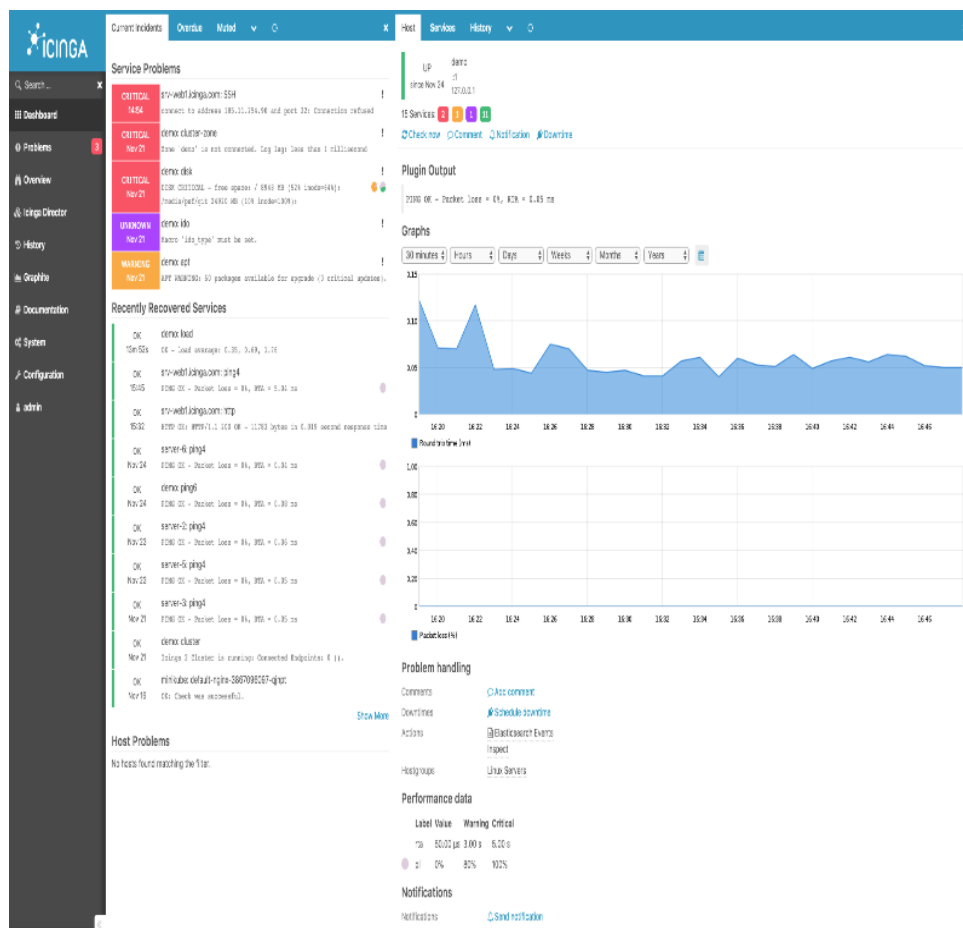
Αποτελεί ισχυρό εργαλείο παρακολούθησης του δικτύου που μας βοηθά στο να εξασφαλιστεί ότι τα κρίσιμα συστήματα, οι εφαρμογές και οι υπηρεσίες μας θα είναι πάντα σε λειτουργία. Παρέχει δυνατότητες, όπως προειδοποιήσεις, χειρισμούς και υποβολή εκθέσεων. Το Nagios Core είναι η καρδιά της εφαρμογής που περιέχει τον πυρήνα παρακολούθησης και ένα βασικό web UI. Στην κορυφή του πυρήνα, μπορούν να εφαρμοστούν τα plugins που θα μας επιτρέψουν να παρακολουθούμε τις υπηρεσίες, τις εφαρμογές και τις μετρήσεις με οπτικοποίηση των δεδομένων όπως γραφήματα, καθώς επίσης υποστηρίζει και βάση δεδομένων MySQL.



Εικόνα 2. Το monitoring system του Nagios.

Icinga web 2

Το Icinga είναι μια Linux based εφαρμογή παρακολούθησης πλήρως ανοικτού κώδικα που ελέγχει τη διαθεσιμότητα των πόρων του δικτύου και ειδοποιεί αμέσως τους χρήστες όταν κάτι δεν λειτουργεί σωστά. Παρέχει δεδομένα επιχειρηματικής ευφυΐας για την σε βάθος ανάλυση και μια ισχυρή διασύνδεση γραμμής εντολών



Εικόνα 3. Το monitoring system του icinga web2.

2.2 Επιλογή λογισμικού.

Για την διπλωματική εργασία επιλέχθηκε για να γίνει χρήση το λογισμικό Icinga web 2. Αυτό που ξεχώρισε το Icinga web 2 από τα υπόλοιπα συστήματα ανοιχτού κώδικα είναι το γεγονός ότι αποτελεί το τελευταίο λειτουργικό που κυκλοφόρησε στην αγορά εργασίας (Φεβρουάριος 2015), καθώς παρουσιάζει τις περισσότερες δυνατότητες σε σύγκριση με τα υπόλοιπα. Το συγκεκριμένο λογισμικό διαθέτει έναν εντελώς νέο σχεδιασμό και πολλές βελτιώσεις οι οποίες φιλικές προς τον χρήστη συγκριτικά με τα υπόλοιπα λειτουργικά συστήματα, γεγονός που το κάνει να υπερτερεί των υπολοίπων. Το Icinga web 2 είναι συμβατό με όλα τα υπάρχοντα plugins, διεπαφές χρήστη (π.χ Classic UI, Icinga Web) και addons. Έχει σχεδιαστεί για να είναι πιο εύκολο στην εγκατάσταση, και είναι αρκετά πιο γρήγορο και ισχυρό. Όπως διαπιστώνουμε παρακάτω παρέχει:

- **Απλή εγκατάσταση:** Σε αντίθεση με τους προκατόχους του, το Icinga 2 έρχεται με IDO, Livestatus, performance data and Graphite writers, σε συμβατότητα με τα χαρακτηριστικά των σχετικών βιβλιοθηκών τους.
- **Εύκολη και αποτελεσματική add-on ενσωμάτωση:** Το Icinga 2 διαθέτει πολλαπλά backends, για οποιοδήποτε add-on και μπορεί να ενσωματωθεί εύκολα. Έτσι, μπορούμε να διαθέτουμε σε πραγματικό χρόνο γραφικές παράστασης παρακολούθησης. Επίσης, αποσκοπεί στη μείωση φορτίου του συστήματος, όπου τα δεδομένα που δεν είναι γραμμένα διαγράφονται στο σκληρό δίσκο πάνω από ένα script όπως συμβαίνει στο Icinga 1 και Nagios.
- **Προσανατολισμένη Απόδοση:** Το Icinga 2 είναι φτιαγμένο για να είναι γρήγορο. Χάρη στον πολυνηματικό σχεδιασμό του, μπορεί να τρέξει χιλιάδες ελέγχους κάθε δευτερόλεπτο χωρίς καμία ένδειξη επιβάρυνσης στην CPU του. Αυτό συμβαίνει ακριβώς για να είμαστε σίγουροι, ότι έχουμε ενσωματωμένα συμπλέγματα ελέγχου που παράγουν δεδομένα απόδοσης για το Icinga2 για διάφορα περιστατικά ή για μία ομάδα χρηστών.

2.3 Icinga Web 2

Το Icinga 2 ξεκίνησε τον Φεβρουάριο του 2015 ως επέκταση του Icinga το οποίο αποτελούσε εφαρμογή παρακολούθησης του συστήματος Nagios. Αποτελεί επέκταση του Icinga η οποία ξαναγράφτηκε για να παρέχει στους χρήστες της ένα σύγχρονο λειτουργικό περιβάλλον εργασίας και υποστήριξης με πολλαπλές βάσεις δεδομένων. Υποστηρίζει γραφήματα, παρέχοντας στους διαχειριστές γραφικά απόδοσης σε πραγματικό χρόνο. Οι διαχειριστές μπορούν να δουν τα φίλτρα, και να κατηγοριοποιήσουν τα προβλήματα, για την παρακολούθηση των οποίων έχουν ήδη ληφθεί μέτρα. Με ένα κανούργιο View

Interface, επιτρέπει στους διαχειριστές να βλέπουν hosts και services σε μία σελίδα. Είναι γραμμένο σε C++ και τρέχει σε Apache Web Server. Οι βάσεις δεδομένων που υποστηρίζει είναι MySQL, PostgreSQL, Oracle, IDODB. Επίσης παρέχει API (Application Programming Interface - διεπαφή προγραμματισμού εφαρμογών) που επιτρέπει στους διαχειριστές να ενσωματώσουν πολλές επεκτάσεις χωρίς πολύπλοκες τροποποίηση του πυρήνα Icinga.

Το Icinga 2 έχει την δυνατότητα να εγκατασταθεί σε Windows, Linux, Mac OS X, και άλλα συστήματα που είναι Unix-like. Κατάλληλα προκατασκευασμένα πακέτα είναι διαθέσιμα για Debian/Ubuntu, Fedora/RHEL/CentOS 6, OpenSUSE/SLES και ArchLinux. Παρέχει επίσης την δυνατότητα χρήσης σε κινητές συσκευές οι οποίες είναι διαθέσιμες σε iOS, Android, BlackBerry Tablet OS και webOS καθώς υπάρχει η εφαρμογή που επιτρέπει την πρόσβαση από εξουσιοδοτημένους χρήστες μέσω κινητών συσκευών τους. Εκδόθηκε σύμφωνα με τους όρους της GNU General Public License έκδοση 2 και αποτελεί ελεύθερο λογισμικό όπως επίσης είναι και ίσως το μόνο που είναι διαθέσιμο σε 21 γλώσσες εκτός των Αγγλικών. Τέλος, άξιο αναφοράς χρίζει το γεγονός ότι παρέχει στο διαδίκτυο πληθώρα πληροφοριών σχετικά με την εγκατάσταση και την παραμετροποίηση του.

Σε αντίθεση με τις προηγούμενες εκδόσεις, η διεπαφή αυτή έχει μια λεπτομερή εικόνα για την υπηρεσία υποδοχής και για ελέγχους που δεν εκτελούνται στο χρόνο. Έχει βελτιωμένη την υπηρεσία ενσωμάτωσης υπηρεσίες καταλόγου Active Directory και σε άλλους διακομητές LDAP. Το Icinga web 2 υποστηρίζει την φόρτωση χρηστών, ομάδες χρηστών καθώς και ομάδες των μελών αυτών όπου έλεγχος φόρτωσης-ταυτότητας γίνεται από το Active Directory. Επιπλέον, σημαντική πρόοδος είναι η δημιουργία των υπηρεσιών hosts και servers. Μια μεγάλη ευκαιρία είναι η δημιουργία host συστημάτων και υπηρεσιών εξυπηρέτησης αντί να προσδιορίζονται μέσω των επιβεβαιωμένων αρχείων του Icinga κάθε χρήστης τώρα μπορεί να τα δημιουργήσει μέσω της διεπαφής του δικτύου και ακόμα να τα κοινοποιήσει σε άλλους. Η ρύθμιση παραμέτρων μας επιτρέπει να προσθέσουμε ενέργειες μόνο σε ορισμένα hosts και servers αλλά υποστηρίζει μακροεντολές για hosts, service name για προσαρμοσμένες μεταβλητές.

Τέλος η νέα δικτυακή διεπαφή παρέχει ένα πολύ βασικό API (Application Programming Interface - Διεπαφή Προγραμματισμού Εφαρμογών) για τον προγραμματισμό αφαίρεσης host και εξυπηρέτηση σε νεκρούς χρόνους.

2.3 Χρήση Εικονικών μηχανών (Virtual Machines).

Όσον αφορά την εγκατάσταση και παραμετροποίηση του icinga web 2, αρχικά έγινε χρήση ενός λογισμικού το οποίο θα μας επέτρεπε την δημιουργία ενός εικονικού περιβάλλοντος (λειτουργικού συστήματος) ίδιου με το περιβάλλον του server του Χαροκοπέιου, δηλαδή μια εικονική μηχανή. Τα δημοφιλέστερα λογισμικά γι' αυτό τον σκοπό είναι το Virtualbox (<https://www.virtualbox.org/>) της Oracle καθώς και το VMware player (<http://www.vmware.com/>) της VMware Inc. Και τα δύο λογισμικά διατίθενται δωρεάν, οι διαφορές μεταξύ τους είναι ελάχιστες και πολλές κριτικές αναφέρουν το VirtualBox ως πιο γρήγορο στην απόδοση του.

Επιλέξαμε να χρησιμοποιήσουμε το Virtualbox v6.1.4 το οποίο μπορείς κανείς να το βρει και να το εγκαταστήσει από την ιστοσελίδα (<https://www.virtualbox.org/wiki/Downloads>). Μέσω αυτού του προγράμματος δημιουργήσαμε ένα εικονικό μηχάνημα (Virtual Machine). Σε αυτό το εικονικό μηχάνημα εγκαταστήσαμε το λειτουργικό σύστημα Debian 9 stretch (64 bit) έκδοση την οποία και κατεβάσαμε από την ιστοσελίδα (<https://www.debian.org/releases/stable/debian-installer/>). Την τελευταία έκδοση του Icinga web 2 μπορεί να την βρει κάποιος στην επίσημη ιστοσελίδα του, (<https://www.icinga.org/download/>). Στη συνέχεια στην ενότητα που ακολουθεί πρόκειται να παρουσιαστούν αναλυτικά τα βήματα που ακολουθήθηκαν και οι εντολές που εκτελέστηκαν με σκοπό την εγκατάσταση της εφαρμογής και των προ-απαιτούμενων της.

2.4 Διαδικασία εγκατάστασης του Icinga Web 2.

Αρχικά εγκαθιστούμε το Icinga 2 το οποίο για να γίνει σωστά θα πρέπει να επιλέξουμε τα επίσημα "package repositories" τα οποία θα ταιριάζουν με το λειτουργικό μας σύστημα και τα οποία τα επιλέγουμε από το (<http://packages.icinga.org/debian/>). Στη συνέχεια, προσθέτουμε το κατάλληλο repository package για το λειτουργικό σύστημα μας το οποίο όπως είπαμε είναι το Debian 9 Stretch. Πριν από αυτό όμως, έχουμε σιγουρευτεί ότι ανοίξαμε το τερματικό του root ή αλλιώς κάναμε su- στο απλό τερματικό του Debian και γράφουμε τις παρακάτω εντολές:

Εντολή εγκατάστασης Debian stretch και πακέτων.

```
wget -O - https://debmon.org/debmon/repo.key 2>/dev/null | apt-key add  
echo 'deb http://debmon.org/debmon debmon-stretch main'  
>/etc/apt/sources.list.d/debmon.list
```

Εντολή ενημέρωσης.

```
apt-get update
```

Εντολή αναβάθμισης.

```
apt-get upgrade
```

Στην συνέχεια κάνουμε εγκατάσταση του icinga 2 με την ακόλουθη εντολή:

```
apt-get install icinga2
```

Για τον έλεγχο των εξωτερικών υπηρεσιών στο icinga 2 χρειάζεται να τρέξουμε ορισμένα plugins τα οποία μπορούν να χρησιμοποιηθούν από το icinga 2, με την ακόλουθη εντολή:

```
apt-get install nagios-plugins
```

Στη συνέχεια γίνεται εκκίνηση της λειτουργίας του icinga 2 με την ακόλουθη εντολή:

```
/etc/init.d/icinga2 start
```

Το επόμενο βήμα αφορά την εγκατάσταση του icinga web 2. Το DB IDO (Database Icinga Data Output) διαμορφώνει την εξαγωγή όλων των ρυθμίσεων και την κατάσταση των πληροφοριών σε μια βάση δεδομένων και υποστηρίζεται μόνο από τη MySQL και τη PostgreSQL. Εμείς επιλέξαμε να χρησιμοποιήσουμε τη MySQL. Για το λόγο αυτό κάνουμε εγκατάσταση τη MySQL με την παρακάτω εντολή (κατά την εγκατάσταση θα μας εμφανίσει κάποια παράθυρα και θα χρειαστεί να βάλουμε ένα κωδικό για να δημιουργήσουμε τον root λογαριασμό):

```
apt-get install mysql-server mysql-client
```

Έπειτα χρειάζεται να εγκαταστήσουμε το IDO module της MySQL οπότε τρέχουμε την εντολή:

```
apt-get install icinga2-ido-mysql
```

Θα εμφανιστούν τέσσερα νέα παράθυρα όπου το πρώτο μας ρωτάει αν θέλουμε να ενεργοποιήσουμε τη λειτουργία Ido-mysql και επιλέγουμε NO, διότι θα ενεργοποιηθεί αμέσως μετά. Στο δεύτερο παράθυρο μας ρωτάει για την επιβεβαίωση της MySQL όπου και επιβεβαιώνουμε επιλέγοντας YES καθώς στο τρίτο παράθυρο μας ζητάει κωδικό πρόσβασης που ορίσαμε για την MySQL και στο τέταρτο παράθυρο την επιβεβαίωση του. Μόλις ολοκληρωθεί η ρύθμιση, μπορούμε να εντάξουμε το αρχείο Ido-mysql.conf. Θα πρέπει να έχουμε σημειώσει τις διαφορετικές ρυθμίσεις διότι θα μας χρησιμεύσει για τη διαμόρφωσή του web interface.

Εγκατάσταση Ido-MySQL.

```
nano /etc/icinga2/features-available/ido-mysql.conf
```

```
user="icinga"  
password="*****"  
host="localhost"  
database="icinga1"
```

Μετά την ενεργοποίηση της Ido-MySQL κάνουμε επανεκκίνηση το Icinga 2 με την εντολή:

```
service icinga2 restart
```

και συνεχίζουμε με την εγκατάσταση του web server με την εντολή:

```
apt-get install apache2
```

Επειδή web διεπαφές και άλλα add-ons του Icinga είναι σε θέση να στείλουν εντολές στο Icinga 2 μέσω του External Command Pipe εμείς το ενεργοποιούμε με την εντολή:

```
icinga2 feature enable command
```

Στη συνέχεια κάνουμε restart το icinga 2 όπως θα μας ζητηθεί και συνεχίζουμε με την εγκατάσταση του icinga web 2 με την εντολή:

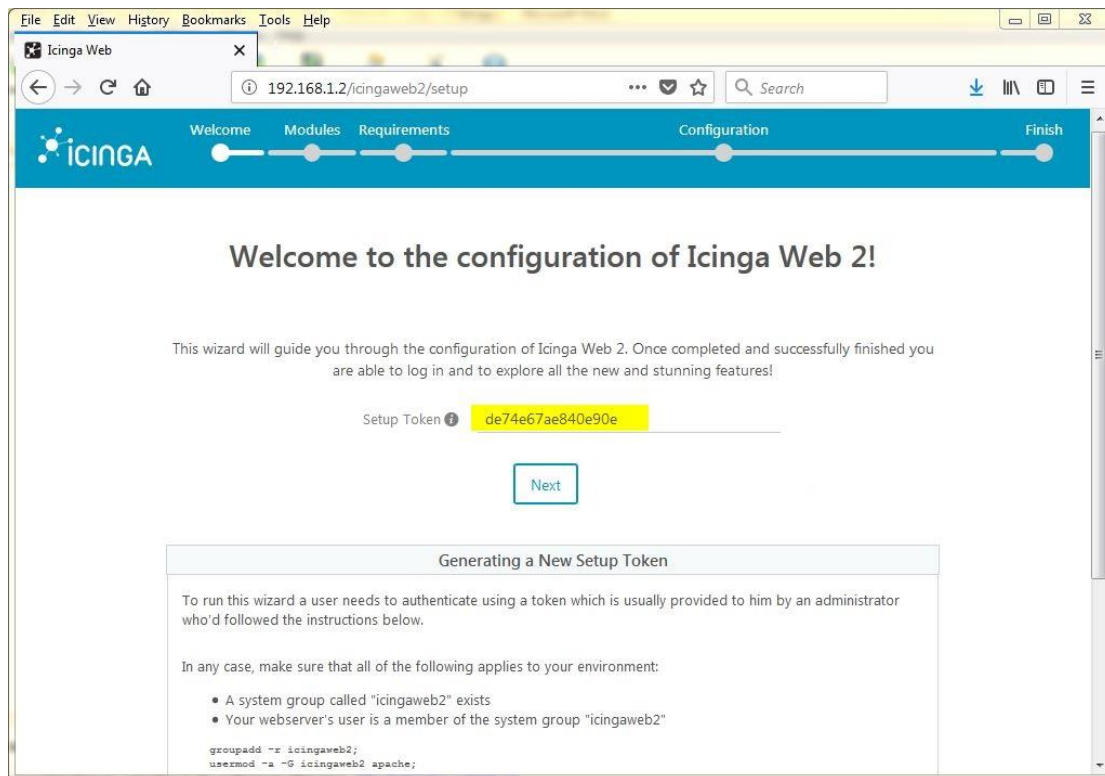
```
apt-get install icingaweb2
```

Σχετικά με την εγκατάσταση του icinga web 2 έχουμε την δυνατότητα να την κάνουμε με δύο τρόπους μέσω web setup και μέσω source στην περίπτωση μας την κάναμε με web setup οπότε τα βήματα που θα ακολουθηθούν είναι για τον τρόπο αυτό.

Ξεκινώντας χρειαζόμαστε ένα token πιστοποίησης το οποίο το παίρνουμε με την εντολή:

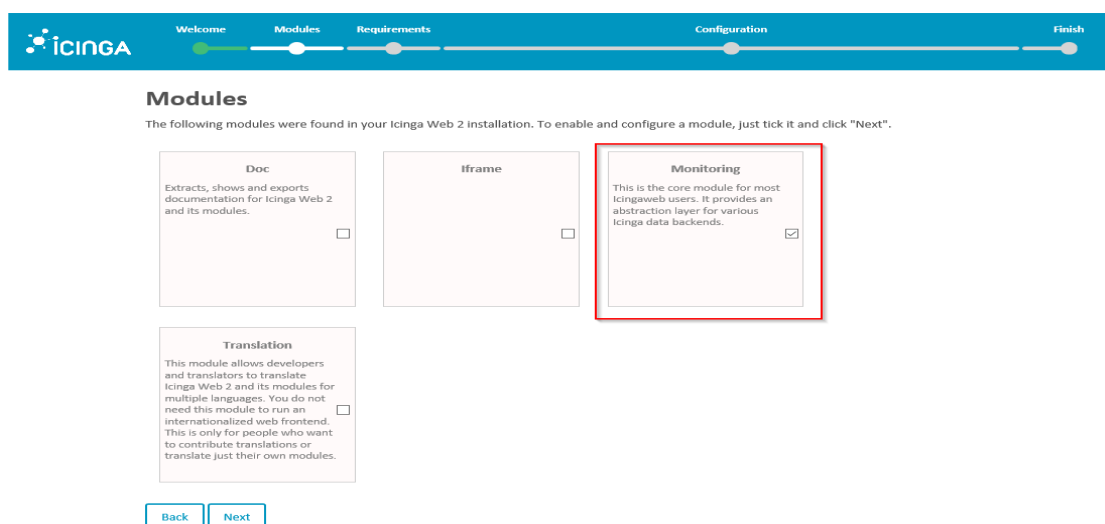
```
icingacli setup token create
```

Στη συνέχεια ανοίγουμε τον browser μας και πηγαίνουμε στη διεύθυνση <http://localhost/icingaweb2/setup>. Πληκτρολογούμε τον κωδικό Token όπως φαίνεται στην εικόνα και πατάμε Next.



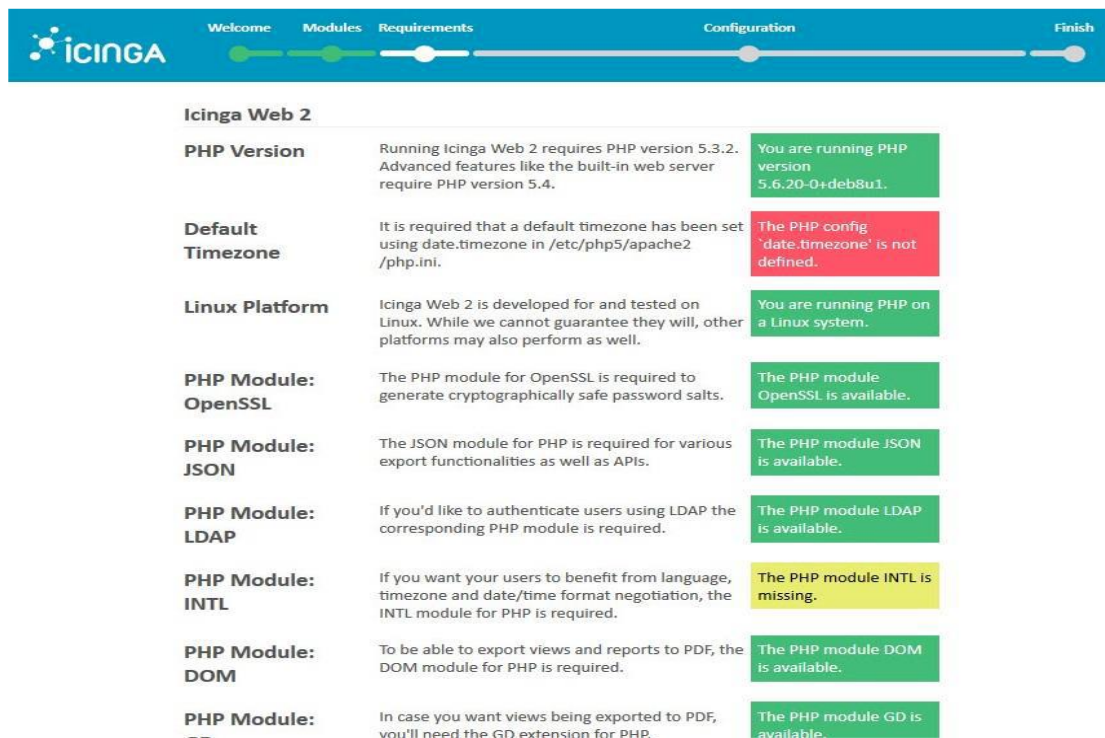
Εικόνα 4. Αρχικό περιβάλλον διαπαφής.

Στην επόμενη σελίδα επιλέγουμε τα modules που θέλουμε στη δική μας περίπτωση το monitoring και πατάμε Next.



Εικόνα 5. Modules.

Τώρα είναι το κύριο τμήμα για να ρυθμίσουμε όλες τις απαιτήσεις πριν από τη μετάβαση στο επόμενο βήμα.

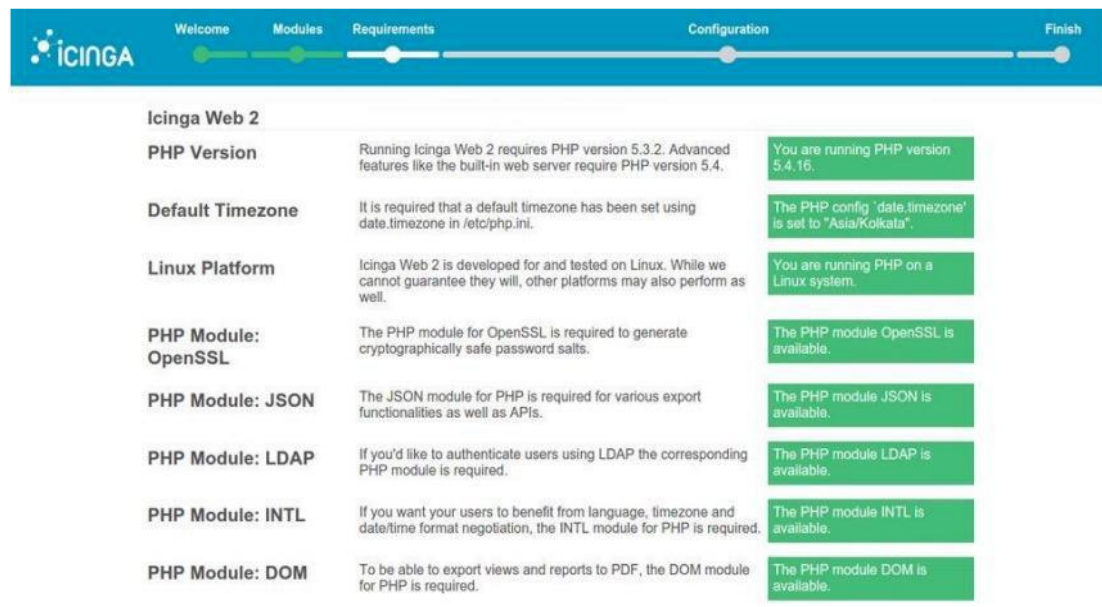


Εικόνα 6. Ρυθμίσεις λογισμικού.

Όσον αφορά την διόρθωση του default timezone error που μας εμφανίζει χρειάζεται να ανοίξουμε το αρχείο που μας υποδεικνύει και να αλλάξουμε την date.timezone σε date.timezone = 'Europe/Athens', αποθηκεύουμε και τρέχουμε την ακόλουθη εντολή ώστε να σβήσουμε τα warnings:

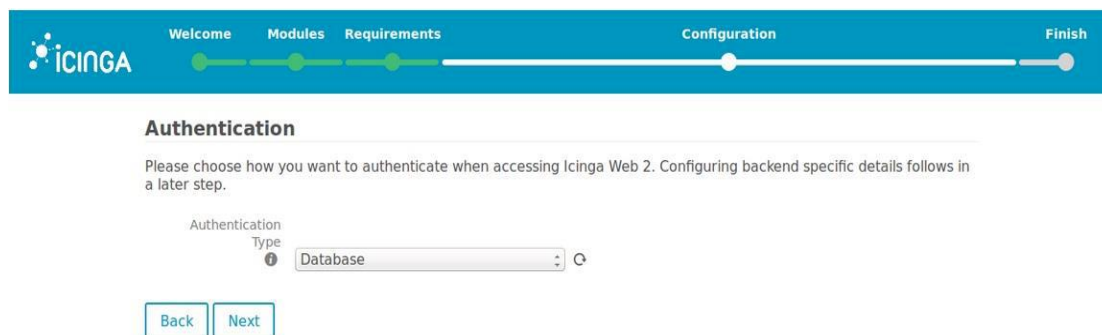
```
apt-get install php5-json php5-gd php5-imagick php5-pgsql php5-intl
```

Κάνουμε refresh την σελίδα μας για να πάρει τις αλλαγές που κάναμε και να μας τις εμφανίσει με πράσινο χρώμα και πατάμε Next.



Εικόνα 7. Οι ρυθμίσεις του συστήματος μετά τις αλλαγές.

Στην επόμενη οθόνη θα πρέπει να επιλέξουμε το Authentication Type ως Database.



Εικόνα 8. Ορισμός του Authentication Type.

```
mysql -u root -p
Enter password:
MariaDB [(none)]> create database icinga2;
Query OK, 1 row affected (0.00 sec)
```

Συνεχίζουμε στο Database Resource όπου και δημιουργούμε μια καινούργια βάση δεδομένων την οποία θα κάνουμε Validate και αφού μας εμφανίσει ότι έγινε, πατάμε Next.

Database Resource

Now please configure the database resource where to store users and user groups.
Note that the database itself does not need to exist at this time as it is going to be created once the wizard is about to be finished.

Resource Name *

Database Type *

Host *

Port *

Database Name *

Username *

Password *

Character Set

Persistent ☐

* Required field

Εικόνα 9. Ορισμός του Database Resource.

Η επόμενη οθόνη database setup μας καλεί να δηλώσουμε το λογαριασμό root για το mysql ώστε να δημιουργηθεί η καινούργια database που ορίσαμε προηγουμένως, πληκτρολογώντας το username και password μας.

Database Setup

It seems that either the database you defined earlier does not yet exist and cannot be created using the provided access credentials, the database does not have the required schema to be operated by Icinga Web 2 or the provided access credentials do not have the sufficient permissions to access the database. Please provide appropriate access credentials to solve this.

Username *

Password *

* Required field

Εικόνα 10. Δημιουργία λογαριασμού root.

Στη συνέχεια στην οθόνη Authentication Backend δηλώνουμε το όνομα της Database μας.

The screenshot shows the 'Authentication Backend' step of the Icinga Configuration Wizard. At the top, a progress bar indicates the current step is 'Configuration', with 'Welcome', 'Modules', 'Requirements', and 'Finish' also visible. The Icinga logo is on the left. The main heading is 'Authentication Backend'. Below it, a message states: 'As you've chosen to use a database for authentication all you need to do now is defining a name for your first authentication backend.' A text input field labeled 'Backend Name' contains the value 'icingaweb2'. At the bottom, there are 'Back' and 'Next' buttons.

Εικόνα 11. Δήλωση ονόματος της Database.

Στην επόμενη οθόνη Administration, δημιουργούμε τον Administrator για το Icinga web 2, δηλώνοντας το username και το password που θα έχει.

The screenshot shows the 'Administration' step of the Icinga Configuration Wizard. The progress bar at the top shows 'Configuration' as the current step. The heading is 'Administration'. A message states: 'Now it's time to configure your first administrative account or group for Icinga Web 2.' There are three text input fields: 'Username' with the value 'icingaadmin', 'Password' with masked characters, and 'Repeat password' also with masked characters. Each field has an information icon on the left and a clear icon on the right. At the bottom, there are 'Back' and 'Next' buttons. Below the buttons, a small note says '* Required field'.

Εικόνα 12. Δημιουργία Administrator.

Στην οθόνη που ακολουθεί θα πρέπει να ρυθμίσουμε τις παραμέτρους της εφαρμογής όπως φαίνονται στην εικόνα.

The screenshot shows the 'Application Configuration' step of the Icinga Web 2 installation wizard. The progress bar at the top indicates the current step is 'Configuration'. Below the header, a note states: 'Note that choosing "Database" as preference storage causes Icinga Web 2 to use the same database as for authentication.' The configuration options are as follows:

- Show Stacktraces: ☒
- User Preference Storage Type *: Database
- Logging Type *: Syslog
- Logging Level *: Error
- Application Prefix *: icingaweb2

Buttons for 'Back' and 'Next' are visible at the bottom of the configuration area. A legend indicates that an asterisk (*) denotes a required field.

Εικόνα 13. Ρύθμιση παραμέτρων.

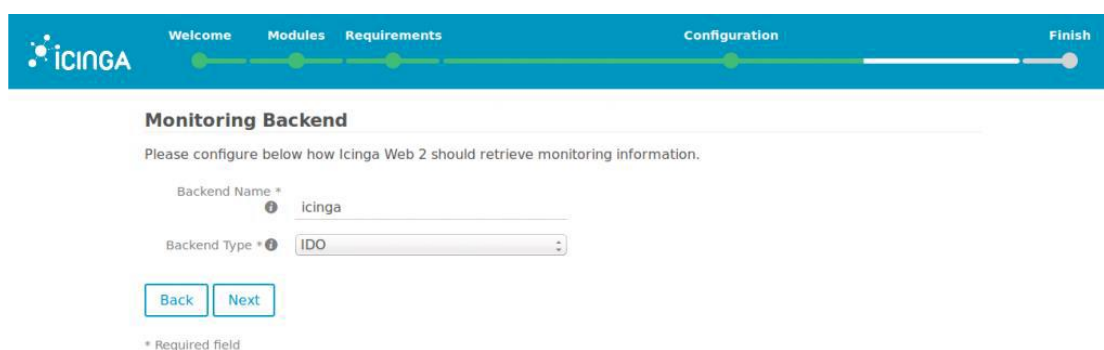
Βεβαιωνόμαστε ότι η ρύθμιση του icinga web 2 έχει γίνει με επιτυχία και επιλέγουμε από που θα παίρνει τα δεδομένα.

The screenshot shows the 'Welcome to the configuration of the monitoring module for Icinga Web 2!' step. The progress bar at the top indicates the current step is 'Configuration'. Below the header, the text reads: 'This is the core module for Icinga Web 2. It offers various status and reporting views with powerful filter capabilities that allow you to keep track of the most important events in your monitoring environment.' Buttons for 'Back' and 'Next' are visible at the bottom of the configuration area.

Εικόνα 14. Επιτυχής επιβεβαίωση ρυθμίσεων.

Θα διαμορφώσουμε τώρα την ενότητα παρακολούθησης του Icinga. Εδώ στις επόμενες δύο οθόνες, θα ρυθμίσουμε το backend παρακολούθησης (το οποίο παραμένει ως προεπιλογή) & τότε θα πρέπει να εισάγουμε ξανά τις πληροφορίες της βάσης δεδομένων, αλλά πριν το κάνουμε αυτό πρέπει να εισάγουμε το σχήμα IDO.

```
mysql -u root -p
MariaDB [(none)]> CREATE DATABASE icinga;
MariaDB [(none)]> GRANT SELECT, INSERT, UPDATE, DELETE, DROP, CREATE VIEW,
INDEX, EXECUTE ON icinga.* TO 'icinga'@'localhost' IDENTIFIED BY 'icinga';
MariaDB [(none)]> FLUSH PRIVILEGES;
MariaDB [(none)]> EXIT;
```



The screenshot shows the Icinga Web 2 Configuration interface. At the top, a blue header bar contains the Icinga logo and a progress bar with five steps: Welcome, Modules, Requirements, Configuration (highlighted), and Finish. Below the header, the 'Monitoring Backend' section is active. It contains a message: 'Please configure below how Icinga Web 2 should retrieve monitoring information.' There are two input fields: 'Backend Name *' with the value 'icinga' and 'Backend Type *' with a dropdown menu showing 'IDO'. Below these fields are 'Back' and 'Next' buttons. A small asterisk indicates that the fields are required.

Εικόνα 15. Δήλωση επιλογής δεδομένων.

Εδώ δημιουργήσαμε μια άλλη βάση δεδομένων για το Icinga IDO και θα εισάγουμε τώρα το σχήμα IDO σε αυτό.

```
mysql -u root -p icinga < /usr/share/icinga2-ido-mysql/schema/mysql.sql
```

Επίσης, κάντε τις αλλαγές ως βάση δεδομένων σας στο ακόλουθο αρχείο.

```
nano /etc/icinga2/features-available/ido-mysql.conf
/**
 * The db_ido_mysql library implements IDO functionality
 * for MySQL.
 */
library "db_ido_mysql"
object IdoMysqlConnection "ido-mysql" {
    user = "icinga"
    password = "icinga"
    host = "localhost"
    database = "icinga"
}
```

Στη συνέχεια δηλώνουμε την ido mysql database που φτιάξαμε προηγουμένως και κάνουμε validate

Monitoring IDO Resource

Please fill out the connection details below to access the IDO database of your monitoring environment.

Resource Name *

Database Type *

Host *

Port *

Database Name *

Username *

Password *

Character Set

Persistent ☒

* Required field

Εικόνα 16. Δήλωση της ido mysql database.

Στην επόμενη οθόνη που μας εμφανίζει, αφήνουμε ίδιο το Command Transport,

Command Transport

Please define below how you want to send commands to your monitoring instance.

Transport Name *

Transport Type *

Command File *

* Required field

Εικόνα 17. Οθόνη Command Transport.

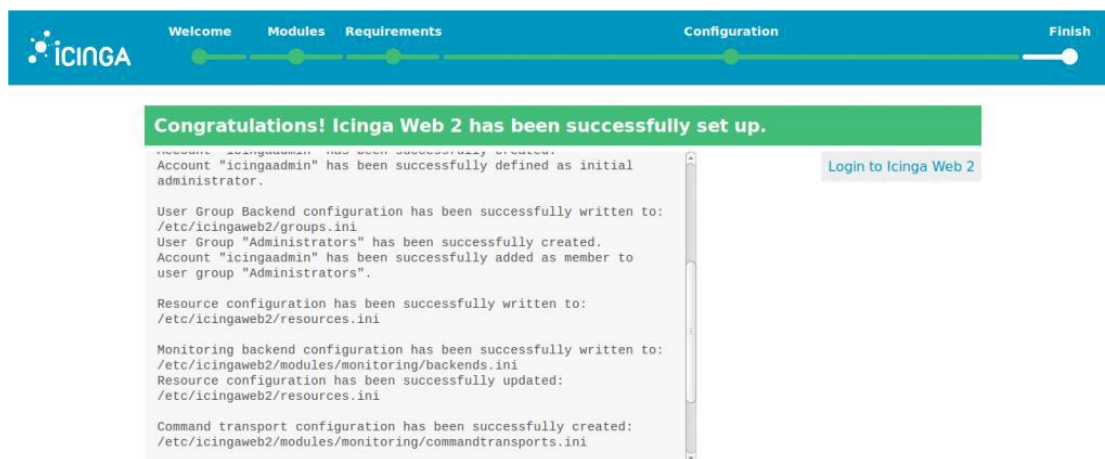
Όπως επίσης το ίδιο κάνουμε και στην οθόνη Monitoring Security.



The screenshot shows the Icinga web interface during the 'Configuration' step of the setup process. The progress bar at the top indicates the current step. The main heading is 'Monitoring Security'. Below it, a message states: 'To protect your monitoring environment against prying eyes please fill out the settings below.' There are two tabs: 'Protected' and 'Custom Variables'. The 'Custom Variables' tab is active, showing a text input field with the value '*pw*,*pass*,community'. Below the input field are 'Back' and 'Next' buttons.

Εικόνα 18. Οθόνη Monitoring Security.

Τέλος, στην οθόνη που ακολουθεί μπορούμε να δούμε μια γενική επισκόπηση των ρυθμίσεων που έχουμε κάνει, ολοκληρώνεται η εγκατάσταση του icinga web 2 και συνδεόμαστε στην εφαρμογή.



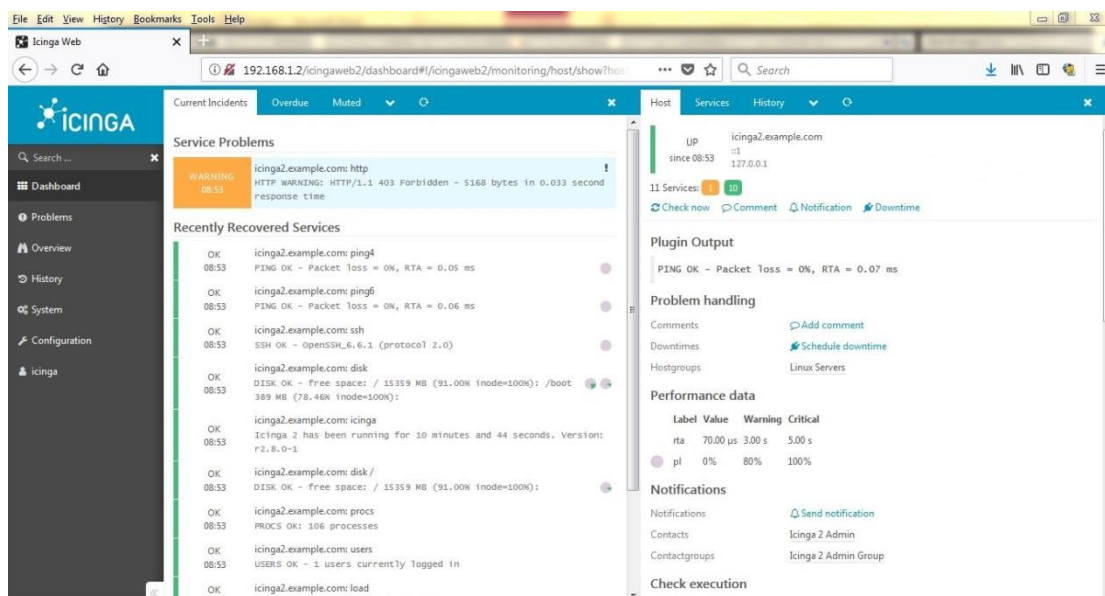
The screenshot shows the Icinga web interface after successful installation. The progress bar at the top indicates the 'Configuration' step is complete. A green banner at the top reads 'Congratulations! Icinga Web 2 has been successfully set up.' Below the banner is a scrollable text area containing the following messages: 'Account "icingaadmin" has been successfully defined as initial administrator.', 'User Group Backend configuration has been successfully written to: /etc/icingaweb2/groups.ini', 'User Group "Administrators" has been successfully created.', 'Account "icingaadmin" has been successfully added as member to user group "Administrators".', 'Resource configuration has been successfully written to: /etc/icingaweb2/resources.ini', 'Monitoring backend configuration has been successfully written to: /etc/icingaweb2/modules/monitoring/backends.ini', 'Resource configuration has been successfully updated: /etc/icingaweb2/resources.ini', and 'Command transport configuration has been successfully created: /etc/icingaweb2/modules/monitoring/commandtransports.ini'. A 'Login to Icinga Web 2' button is visible in the top right corner.

Εικόνα 19. Γενική επισκόπηση των ρυθμίσεων και εγκατάσταση.



Εικόνα 20. Αρχική οθόνη εισαγωγής στη εφαρμογή.

Εδώ έχουμε την δυνατότητα να δούμε τις ειδοποιήσεις της υπηρεσίας του κύριου Icinga server μας ή μπορούμε απλά να περιηγηθούμε στο URL `http://IP/icingaweb2/` ώστε να έχουμε πρόσβαση στο web interface. Μπορούμε να προσθέσουμε οποιοδήποτε αριθμό κόμβων σε αυτό το σύστημα για την παρακολούθηση.



Εικόνα 21. Αρχική οθόνη εισαγωγής στη εφαρμογή.

Κεφάλαιο 3

3.1 Αναφορά ως προς τα Plugins των Icinga2 & Nagios.

Το Icinga2 είναι ένα αντίγραφο πηγαίου κώδικα του Nagios διατηρώντας έτσι την συμβατότητα με το Nagios και διευκολύνοντας τη μετάβαση μεταξύ των δύο λογισμικών παρακολούθησης.

Σε αντίθεση με πολλά άλλα εργαλεία παρακολούθησης, το Nagios δεν περιλαμβάνει εσωτερικούς μηχανισμούς για τον έλεγχο της κατάστασης των κεντρικών υπολογιστών και των υπηρεσιών στο δίκτυο. Αντίθετα, το Nagios βασίζεται σε εξωτερικά προγράμματα (που ονομάζονται plugins - “επεκτάσεις”) για να κάνει όλη την διαδικασία.

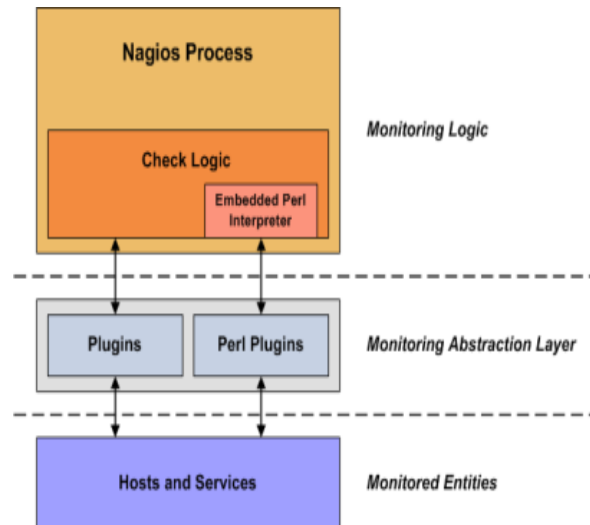
3.2 Τι είναι τα Plugins ή “Επεκτάσεις”.

Τα Plugins ή “επεκτάσεις” είναι μεταγλωττισμένα εκτελέσιμα αρχεία ή δέσμες ενεργειών (δέσμες ενεργειών (compiled binaries) Perl scripts, shell scripts, Python, PHP, Ruby κ.λπ.) που μπορούν να εκτελεστούν από μια γραμμή εντολών για να ελέγξουν την κατάσταση ή έναν κεντρικό υπολογιστή ή υπηρεσία. Το Nagios Core στην προκειμένη περίπτωση το Icinga2 χρησιμοποιεί τα αποτελέσματα από τα plugins για να καθορίσει την τρέχουσα κατάσταση των κεντρικών υπολογιστών και των υπηρεσιών στο δίκτυό σας. Το Nagios Core ή το Icinga2 θα επεξεργαστεί τα αποτελέσματα που λαμβάνει από το plugin και θα λάβει όλες τις απαραίτητες ενέργειες (διαχειριστές συμβάντων εκτέλεσης, αποστολή ειδοποιήσεων κ.λπ.).

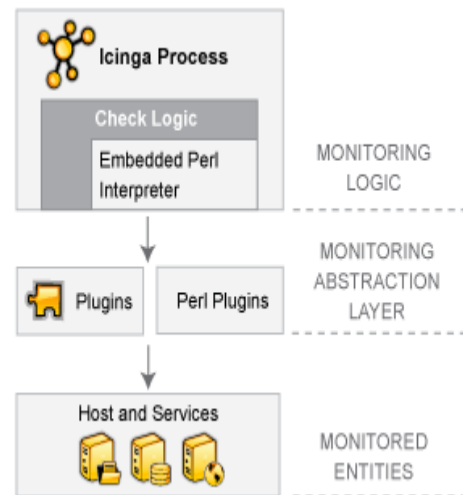
3.3 Πως λειτουργεί ένα Plugin.

Οι επεκτάσεις λειτουργούν ως επίπεδο αφαίρεσης μεταξύ της λογικής παρακολούθησης που υπάρχει στον Nagios Core ή του Icinga2 και των πραγματικών υπηρεσιών και κεντρικών υπολογιστών που παρακολουθούνται. Το προτέρημα αυτού του τύπου αρχιτεκτονικής plugin είναι ότι μπορούμε να παρακολουθούμε με ακρίβεια οτιδήποτε μπορούμε να σκεφτούμε. Εάν μπορούμε να αυτοματοποιήσουμε τη διαδικασία ελέγχου, μπορούμε να το παρακολουθούμε με το Nagios Core. Υπάρχουν ήδη πολλά πρόσθετα (plugins) που έχουν δημιουργηθεί για να παρακολουθούμε βασικούς πόρους, όπως φόρτο επεξεργαστή, χρήση δίσκου, ρυθμούς ring κλπ. Το μειονέκτημα αυτού του τύπου plugin αρχιτεκτονικής είναι το γεγονός ότι το Nagios Core δεν έχει καμία απολύτως ιδέα τι είναι που παρακολουθείτε. Θα μπορούσαμε να παρακολουθούμε στατιστικά στοιχεία κίνησης δικτύου, ρυθμούς σφαλμάτων δεδομένων, εύκρατο χώρο, τάση CPU, ταχύτητα ανεμιστήρα, φορτίο επεξεργαστή, χώρο στο δίσκο. Ο Nagios Core δεν κατανοεί τις ιδιαιτερότητες του τι

παρακολουθείται - παρακολουθεί μόνο τις αλλαγές στην κατάσταση των πόρων αυτών. Μόνο τα Plugins καθαυτά γνωρίζουν το τι ακριβώς παρακολουθούν και πώς να πραγματοποιήσουν τους πραγματικούς ελέγχους.



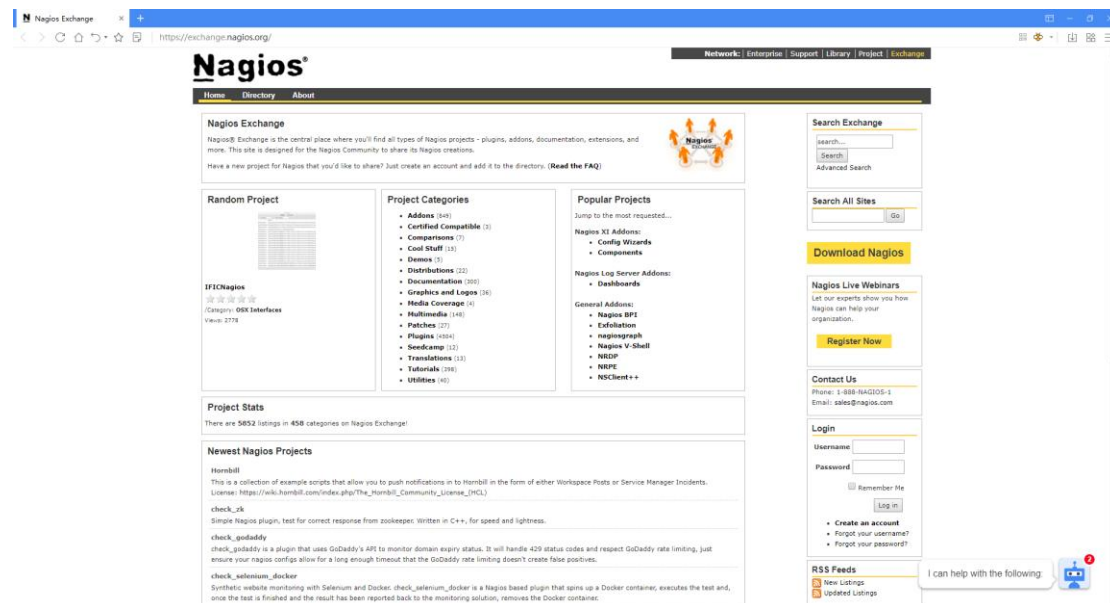
Εικόνα 22. Διάγραμμα λειτουργίας Nagios plugin.



Εικόνα 23. Διάγραμμα λειτουργίας Icinga plugin.

Υπάρχουν πάνω από 4.000 διαθέσιμα Plugins στο Nagios Exchange για την παρακολούθηση πολλών διαφορετικών ειδών συσκευών και υπηρεσιών, όπως:

- HTTP, POP3, IMAP, FTP, SSH, DHCP
- CPU Load, Disk Usage, Memory Usage, Current Users
- Unix/Linux, Windows, and Netware Servers
- Routers and Switches



Εικόνα 24. To Exchange Plugins του Nagios.

Ως διπλωματική εργασία θα ασχοληθούμε με την μελέτη και την διαμόρφωση δύο συγκεκριμένων Plugins : το check_snmp_int Plugin και το check_iftraffic Plugin. Τέλος για να μπορέσουμε να συγκεντρώσουμε όλες τις δικτυακές συσκευές θα χρησιμοποιήσουμε ένα script, το icinaga2-autod, με την χρήση του snmp.

3.4 Διαδικασία καταγραφής δικτυακών συσκευών με το icinaga2-autod

Οι δοκιμές μας θα γίνουν σε έναν test router που έχουμε εγκαταστήσει σε μια VDSL γραμμή με χαρακτηριστικά 50Mbps download και 5Mbps upload και έχει και κάποια τερματικά συνδεδεμένα μέσω των Ethernet θυρών του. Η παραμετροποίηση του ρούτερ βρίσκεται στο **Παράρτημα Α σελ. 58**.

Στην γραμμή εντολών του Debian VM που έχουμε στήσει δίνουμε την παρακάτω εντολή:

```
git clone https://github.com/hobbsh/icinga2-autod.git
```

Αφού ολοκληρωθεί η διαδικασία λήψης και εγκατάστασης του script δίνουμε της παρακάτω εντολές έτσι ώστε να δούμε τα περιεχόμενα του φακέλου όπου έχει γίνει η εγκατάσταση.

```
root@debian9:~# cd icinga2-autod/

root@debian9:~# cd icinga2-autod/
root@debian9:~/icinga2-autod# ls
hosts_home.conf  iana_numbers.json  iciautd.py  icinga-autod.py  LICENSE  README.md  util
```

Εικόνα 25. Το περιεχόμενο του icinga2-autod.

Η εντολή που θα χρησιμοποιηθεί για να γίνει εκτέλεση του script προκειμένου να ξεκινήσει η καταγραφή των δικτυακών συσκευών του δικτύου θα είναι σύμφωνα με την παρακάτω σύνταξη:

```
./icinga-autod.py [-h] -n NETWORK [-L LOCATION] [-c COMMUNITIES]
```

Όπου NETWORK είναι το δίκτυο το οποίο θέλουμε να καταγράψουμε. Όπου LOCATION είναι η τοποθεσία όπου βρίσκεται το δίκτυο μας. Τέλος COMMUNITIES είναι το community string το οποίο χρησιμοποιεί το δίκτυο μας. Επομένως στην δική μας περίπτωση η παραπάνω εντολή θα συνταχθεί με τον παρακάτω τρόπο:

```
./icinga-autod.py -n 192.168.2.0/24 -L TEST_HUA -c Icinga
```

Με το που τελειώσει η διαδικασία το scrip μας ενημερώνει για το πόσες δικτυακές συσκευές έχει βρει και το ότι έχει δημιουργήσει ένα καινούργιο αρχείο με όνομα hosts_test_hua.conf όπως φαίνεται και στο παρακάτω.

```
Starting scan for 192.168.2.0/24
Scan took 11.9320881367 seconds
Found 11 hosts - gathering more info (can take up to 2 minutes)
```

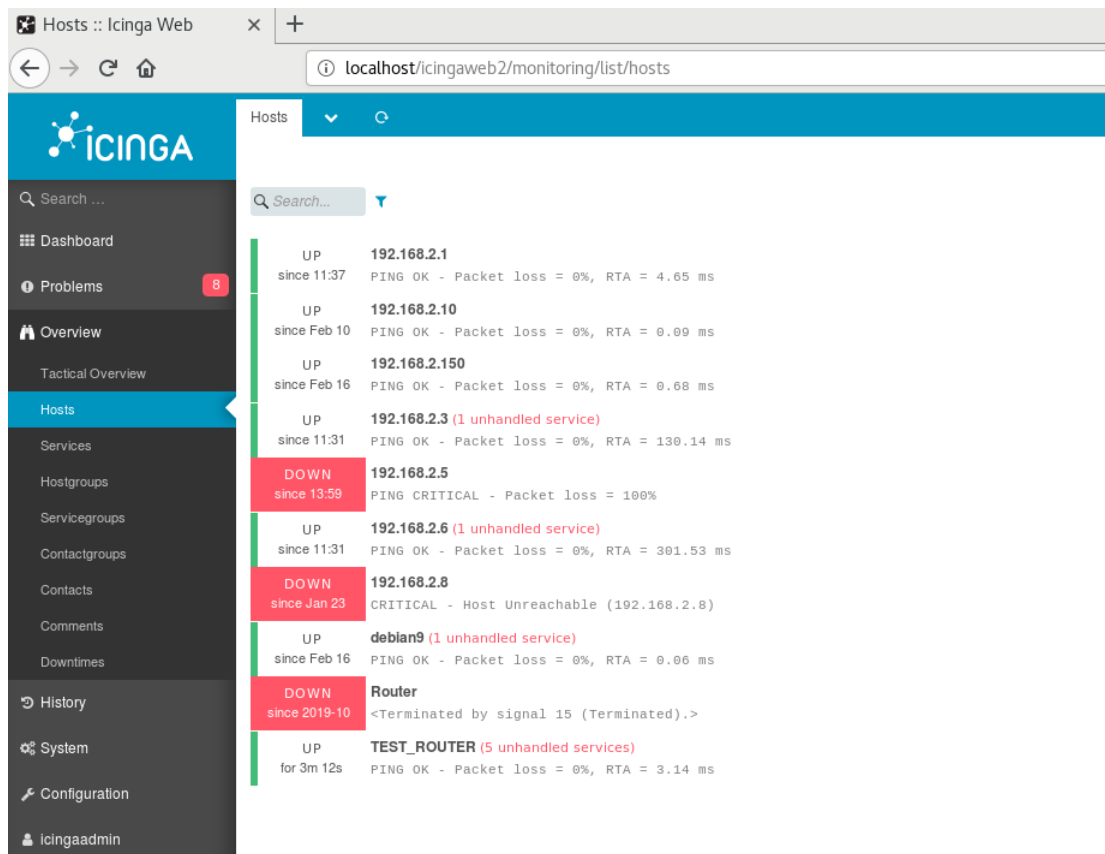
```
Discovery took 33.1995248795 seconds
Writing data to config file. Please wait
Wrote data to hosts_test_hua.conf
```

Εικόνα 26. Αποτέλεσμα εντολής icinga2-autod

Εν συνεχεία το συγκεκριμένο αρχείο το αντιγράφουμε στο directory /etc/icinga2/conf.d/ έτσι ώστε το Icinga να μπορέσει ξεκινήσει τον έλεγχο των συσκευών των οποίων συγκέντρωσε. Για να μπορέσει όμως να μας εμφανίσει τους hosts θα πρέπει να γίνει και ένα restart το service του Icinga με την παρακάτω εντολή:

```
root@debian9:~# service icinga2 restart
```

Έπειτα πηγαίνοντας στο Web Interface του Icinga και στο tab όπου είναι συγκεντρωμένοι οι host παίρνουμε το αποτέλεσμα που βρίσκεται στην παρακάτω εικόνα.



Εικόνα 27. Καταγεγραμμένες δικτυακές συσκευές

3.5 Διαδικασία επιλογής του Plugin.

Επιλέγοντας να δημιουργήσουμε τον τρόπο ελέγχου της μέτρησης του bandwidth των interfaces των δικτυακών συσκευών του Χαροκόπειου Πανεπιστημίου μέσω της χρήσης του icingaweb2 χρειάζεται να επιλέξουμε από το directory του Nagios το plugin που θα είναι το κατάλληλο για την υλοποίηση αυτής της διαδικασίας ή να βρούμε κάποιο πιο κατάλληλο από το exchange plugins του Nagios.

Η επιλογή του κατάλληλου Plugin για την λειτουργία της επιτήρησης της υπηρεσίας είναι ένα σύνολο από συνδυασμούς διαφορετικών plugins, ιδεών και ελέγχων τους (test) ώστε να αποκλείσουμε τα πιο αδύναμα και να καταλήξουμε στο βέλτιστο δυνατό plugin που θα καλύψει το στόχο μας.

Ξεκινώντας λοιπόν με βάση αυτά που είναι ήδη εγκατεστημένα στο directory list του Icinga2:

Εντολή για εμφάνισης λίστας plugins.

```
# root@debian9:~# cd /usr/lib/nagios/plugins
# root@debian9:/usr/lib/nagios/plugins# ls
```

```

root@debian9:/usr/lib/nagios/plugins# ls
check_apt      check_dig      check_fping    check_ide_smart  check_ldap      check_mysql
check_ntp      check_ping     check_sensors  check_ssmtp      check_users
check_breeze   check_disk     check_ftp      check_ifoperstatus  check_ldaps     check_mysql_query
check_ntp_peer check_pop       check_simap     check_swap       check_wave
check_by_ssh   check_disk_smb check_game      check_ifstatus    check_load      check_nagios
check_ntp_time check_procs    check_smtp      check_tcp        negate
check_clamd    check_dns      check_host      check_iftraffic.pl check_log        check_nttp
check_nwstat   check_radius   check_snmp      check_time       urlize
check_cluster check_dummy     check_hpjd      check_imap        check_mailq      check_nntp
check_oracle   check_real      check_snmp_int.pl check_udp         utils.pm
check_dbi      check_file_age  check_http      check_ircd        check_mrtg       check_nscp_api
check_overcr   check_rpc       check_spop      check_udpport     utils.sh
check_dhcp     check_flexlm    check_icmp      check_jabber      check_mrtgtraf   check_nt
check_pgsql    check_rta_multi check_ssh        check_ups

```

Εικόνα 28. Λίστα των plugins

Τα `check_snmp_int.pl` και `check_iftraffic.pl` τα βρήκαμε από το Exchange Plugins του Nagios όπου και αυτά θα χρησιμοποιήσουμε.

3.6 Τι είναι το `check_snmp_int`

Το `check_snmp_int` είναι το plugin το οποίο θα ελέγχει την κατάσταση των θυρών Ethernet των δικτυακών εξοπλισμών. Το script θα μας δίνει αποτέλεσμα OK όταν η εκάστοτε θύρα του δικτυακού εξοπλισμού είναι ενεργή και λειτουργική. Ενώ Critical όταν θα είναι απενεργοποιημένη ή ενεργή αλλά δεν έχει κάποιο καλώδιο Ethernet συνδεδεμένο απάνω της.

Το συγκεκριμένο plugin χρησιμοποιεί την εντολή `snmpget` για να κάνει τους ελέγχους η οποία περιέχεται στο SNMP πακέτο του Nagios plugins. Για να είμαστε σίγουροι ότι έχουμε την πιο πρόσφατη έκδοση του `snmp` πακέτου, δίνουμε την παρακάτω εντολή εγκατάστασης:

```
root@debian9:~# apt-get install snmp
```

Για να δούμε τον τρόπο σύνταξης της εντολής για να τρέξει το script αλλά και για να δούμε τις παραμέτρους που χρησιμοποιεί, δίνουμε την παρακάτω εντολή:

```
root@debian9:/usr/lib/nagios/plugins# ./check_snmp_int.pl -help
```

ή αλλιώς βλέπουμε το παρακάτω σύνδεσμο:

http://nagios.manubulon.com/snmp_int.html

Εντολή ελέγχου της λειτουργικότητας των θυρών Ethernet του TEST_ROUTER:

Από το Terminal του Debian ξεκινάμε να συνδυάζουμε τις παραμέτρους του plugin για να κατανοήσουμε την λειτουργικότητα του.

Εντολή για δικαιώματα root.

```
root@debian9: su
```

Εντολή αλλαγής ευρετηρίου.

```
root@debian: cd
```

Εντολή μετάβασης στο φάκελο των plugins.

```
root@debian: cd /usr/lib/nagios/plugins/
```

Στο επόμενο βήμα θα συντάξουμε την εντολή για να τρέξει το script με τις παραμέτρους -H όπου είναι η IP του host του οποίου θα γίνει ο έλεγχος. Την -C όπου είναι το community string το οποίο χρησιμοποιούμε στο δίκτυο μας και τέλος την -n όπου είναι η περιγραφή του interface που πήραμε από το αποτέλεσμα που τρέξαμε το script icinga2-autod.

Προτού γίνει αυτό θα επιβεβαιώσουμε από το TEST_ROUTER ότι όλες οι θύρες Ethernet που θα δοκιμάσουμε είναι ενεργές και λειτουργικές. Αυτό φαίνεται και στην παρακάτω εικόνα από το output του Router.

```
TEST_ROUTER#sh int desc
Interface      Status      Protocol Description
AT0            admin down  down
Et0            admin down  down
Fa0            up          down
Fa1            up          up      CONNECTION TO ROUTER
Fa2            up          up      CONNECTION TO PC
Fa3            up          up      CONNECTION TO RASPBERRY
Vl1            up          up      LAN
TEST_ROUTER#
```

Εικόνα 29. Output από το TEST_ROUTER

Από το output του Router βλέπουμε ότι όλες οι θύρες Ethernet είναι λειτουργικές και έχουν συνδεδεμένες συσκευές απάνω τους, οι οποίες είναι λειτουργικές εκτός από την Fa0 η οποία δεν έχει κάποια συσκευή συνδεδεμένη απάνω της.

Παράδειγμα σύνταξης εντολής του script check_snmp_int.

```
./check_snmp_int.pl -H (Host) -C (community string) -n (name in desc_oid)
```

Τα αποτελέσματα της εντολής για όλες τις θύρες Ethernet του Router φαίνονται στην παρακάτω εικόνα.

```
root@debian9:/usr/lib/nagios/plugins# ./check_snmp_int.pl -H 192.168.2.100 -C Icinga -n FastEthernet0
FastEthernet0:DOWN: 1 int NOK : CRITICAL
root@debian9:/usr/lib/nagios/plugins# ./check_snmp_int.pl -H 192.168.2.100 -C Icinga -n FastEthernet1
FastEthernet1:UP:1 UP: OK
root@debian9:/usr/lib/nagios/plugins# ./check_snmp_int.pl -H 192.168.2.100 -C Icinga -n FastEthernet2
FastEthernet2:UP:1 UP: OK
root@debian9:/usr/lib/nagios/plugins# ./check_snmp_int.pl -H 192.168.2.100 -C Icinga -n FastEthernet3
FastEthernet3:UP:1 UP: OK
```

Εικόνα 30. Αποτελέσματα εντολής check_snmp_int

Η παραπάνω εικόνα μας επιβεβαιώνει αυτό που βλέπουμε και από το output του Router. Η θύρα Fa0 ενώ είναι administrative UP από το ρούτερ, δηλαδή ενεργή, δεν έχει καλώδιο συνδεδεμένο απάνω της και για αυτό το λόγο από το script παίρνουμε το αποτέλεσμα FastEthernet0:DOWN: 1 int NOK : CRITICAL

Για δοκιμή θα κλείσουμε την Fa3 του Router για να δούμε τι αποτέλεσμα θα μας δώσει η εντολή check_snmp_int. Ακολουθεί το output το Router με την Fa3 administratively απενεργοποιημένη.

```
TEST_ROUTER#sh int desc
Interface          Status      Protocol Description
AT0                 admin down  down
Et0                 admin down  down
Fa0                 up          down
Fa1                 up          up      CONNECTION TO ROUTER
Fa2                 up          up      CONNECTION TO PC
Fa3                 admin down  down    CONNECTION TO RASPBERRY
Vl1                 up          up      LAN
```

Εικόνα 31. Output Router με απενεργοποιημένη Fa3

Η check_snmp_int για την θύρα Ethernet 3.

```
root@debian9:/usr/lib/nagios/plugins# ./check_snmp_int.pl -H 192.168.2.100 -C Icinga -n
FastEthernet3
```

Το αποτέλεσμα αυτής είναι:

```
FastEthernet3:DOWN: 1 int NOK : CRITICAL
```

Το οποίο μας επιβεβαιώνει ότι η θύρα Ethernet του ρούτερ είναι απενεργοποιημένη.

3.7 check_snmp_int Configuration

Βήμα 1°

Το object command για το icingaweb2.

Εντολή για αρχείο commands.

```
root@debian9:~# nano /etc/icinga2/conf.d/commands.conf
```

```
object CheckCommand "snmp-int" {
    import "plugin-check-command"
    command = [ PluginDir + "/check_snmp_int.pl" ]

    arguments = {
        "-H" = "$address$"
        "-C" = "$cn$"
        "-n" = "$int$"
    }

    vars.cn = COMN
```

Στο Object command χρησιμοποιήσαμε τα arguments:

-H --> host IP ADDRESS

-C --> community string

-n --> interface description OID

Βήμα 2°

Είναι η παραμετροποίηση του service στο icingaweb2.

Η εντολή για να παραμετροποιήσουμε το service αρχείο το icingaweb2.

```
root@debian9:~# nano /etc/icinga2/conf.d/services.conf
```

```
apply Service "snmp_int:" for (int => config in host.vars.int) {  
    import "generic-service"  
    check_command = "snmp-int"  
  
    vars += config  
    assign where host.address  
}
```

Βήμα 3°

Είναι η παραμετροποίηση του host στο icingaweb2.

Η εντολή για να παραμετροποιήσουμε το host αρχείο στο icingaweb2.

```
root@debian9:~# nano /etc/icinga2/conf.d/hosts.conf
```

```

object Host "TEST_ROUTER" {
  import "generic-host"
  address = "192.168.2.100"
  vars.os = "Unknown"
    vars.type = "Unknown"
    vars.os_details = "Unknown"
    vars.description = "Cisco IOS Software, C800 Software (C800-
UNIVERSALK9-M), Version 15.3(3)M5, RELEASE SOFTWARE (fc3)^M"
    vars.ports = "Unknown"
  vars.location = "HOME"
  vars.vendor = "ciscoSystems"
    notes = "Technical Support: http://www.cisco.com/techsupport^M"
    vars.int["ATM0"] = {
      int = "ATM0"
    }
    vars.int["Ethernet0"] = {
      int = "Ethernet0"
    }
    vars.int["FastEthernet0"] = {
      int = "FastEthernet0"
    }
    vars.int["FastEthernet1"] = {
      int = "FastEthernet1"
    }
    vars.int["FastEthernet2"] = {
      int = "FastEthernet2"
    }
    vars.int["FastEthernet3"] = {
      int = "FastEthernet3"
    }
  }
}

```

Κεφάλαιο 4

4.1 Τι είναι το check_iftraffic Plugin.

Το check_iftraffic είναι το plugin το οποίο θα μας βοηθήσει να δούμε την κίνηση (utilization) της εκάστοτε θύρας Ethernet του δικτυακού μας εξοπλισμού. Στην προκειμένη περίπτωση του TEST_ROUTER. Το script σαν αποτέλεσμα θα μας επιστρέφει την συνολική κίνηση που έχει γίνει ως την στιγμή που δώσαμε την εντολή. Το μέγεθος του Bandwidth της θύρας το οποίο καταναλώνεται εκείνη την στιγμή, όπως και το ποσοστό πληρότητας της μέγιστης ταχύτητας της θύρας Ethernet.

Το συγκεκριμένο script χρησιμοποιεί και αυτό το SNMP πρωτοκόλλο για να μας δώσει το utilization της θύρας Ethernet. Το script αυτό όπως και τα πιο πολλά που χρησιμοποιούνται στο Nagios αλλά και στο Icinga έχουν γραφτεί σε γλώσσα προγραμματισμού Perl.

Για να προχωρήσουμε στην εγκατάσταση του plugin ακολουθούμε τον παρακάτω τρόπο με τις εξής εντολές:

Ξεκινάμε με την εντολή στο terminal su για να γίνει η αλλαγή του χρήστη σε root.

Εντολή ενημέρωσης του συστήματος

```
root@debian9:~# apt-get update
```

Εντολή αναβάθμισης

```
root@debian9:~# apt-get upgrade
```

Εντολή αλλαγής ευρετηρίου

```
root@debian9:~# cd /usr/lib/nagios/plugins/
```

Εντολή εγκατάστασης Perl

```
root@debian9:~# apt-get -y install libnet-snmp-perl
```

4.2 Εγκατάσταση check_iftraffic στο Icinga2.

Εντολή εγκατάστασης του check_iftraffic script.

```
root@debian9:~# wget https://github.com/NETWAYS/check_iftraffic.git
```

Εντολή μεταφοράς του script στο σωστό ευρετήριο

```
root@debian9:~# install -o root -g root -m755 *.pl /usr/lib/nagios/plugins/
```

Εντολή επανεκκίνησης του Icinga2

```
root@debian9:~# systemctl restart icinga2
```

Εντολή ελέγχου του status του Icinga2

```
root@debian9:~# systemctl status icinga2
```

Για να δούμε τον τρόπο σύνταξης της εντολής για να τρέξει το script αλλά και για να δούμε τις παραμέτρους που χρησιμοποιεί, δίνουμε την παρακάτω εντολή:

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -help
```

ή αλλιώς βλέπουμε το παρακάτω σύνδεσμο:

```
https://github.com/NETWAYS/check\_iftraffic#run
```

Εντολή ελέγχου του utilization των θυρών Ethernet του TEST_ROUTER:

Από το Terminal του Debian ξεκινάμε να συνδυάζουμε τις παραμέτρους του plugin για να κατανοήσουμε την λειτουργικότητα του.

Εντολή για δικαιώματα root.

```
root@debian9: su
```

Εντολή αλλαγής ευρετηρίου.

```
root@debian: cd
```

Εντολή μετάβασης στο φάκελο των plugins.

```
root@debian: cd /usr/lib/nagios/plugins/
```

Στο επόμενο βήμα θα συντάξουμε την εντολή για να τρέξει το script με τις παραμέτρους -H όπου είναι η IP του host του οποίου θα γίνει ο έλεγχος. Την -C όπου είναι το community string το οποίο χρησιμοποιούμε στο δίκτυο μας. Την -i όπου είναι η περιγραφή του interface που πήραμε από το αποτέλεσμα που τρέξαμε το script icinga2-autod. Την -b όπου ορίζουμε το μέγιστο μέγεθος της ταχύτητας της εκάστοτε θύρας Ethernet. Στην δικιά μας περίπτωση η τιμή της θα είναι 100 επειδή οι θύρες μας είναι FastEthernet. Την -u όπου ορίζουμε το ρυθμό μετάδοσης των δεδομένων μέσω της θύρας Ethernet, δηλαδή giga/mega/kilo ή bits ανά δευτερόλεπτο. Την -w όπου θα είναι η τιμή προειδοποίησης που θα ορίσουμε, εξ'ορισμού το script την έχει στο 85%. Τέλος την -c όπου είναι η τιμή της κρίσιμης κατάστασης της θύρας Ethernet, όπου εξόρισμού στο script είναι στο 98%.

Προτού γίνει αυτό θα επιβεβαιώσουμε από το TEST_ROUTER ότι όλες οι θύρες Ethernet που θα δοκιμάσουμε είναι ενεργές και λειτουργικές. Αυτό φαίνεται και στην παρακάτω εικόνα από το output του Router.

```

TEST_ROUTER#sh int desc
Interface          Status      Protocol Description
AT0                admin down  down
Et0                admin down  down
Fa0                up          down
Fa1                up          up      CONNECTION TO ROUTER
Fa2                up          up      CONNECTION TO PC
Fa3                up          up      CONNECTION TO RASPBERRY
Vl1                up          up      LAN
TEST_ROUTER#

```

Εικόνα 32. Output από το TEST_ROUTER

Από το output του Router βλέπουμε ότι όλες οι θύρες Ethernet είναι λειτουργικές και έχουν συνδεδεμένες συσκευές απάνω τους, οι οποίες είναι λειτουργικές εκτός από την Fa0 η οποία δεν έχει κάποια συσκευή συνδεδεμένη απάνω της.

Παράδειγμα σύνταξη εντολής του check_iftraffic script.

```

root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i (oid
description) -b (1000/100/10) -u (g/m/k) -w 85 -c 98

```

Τα αποτελέσματα της εντολής για όλες τις θύρες Ethernet του Router φαίνονται παρακάτω.

```

root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i
FastEthernet0 -b 100 -u m -w 85 -c 98

Total RX Bytes: 0.00 MB, Total TX Bytes: 0.00 MB<br>Average Traffic: 0.00 kB/s (0.0%) in, 0.00
kB/s (0.0%) out|inUsage=0.0%;85;98 outUsage=0.0%;85;98 inAbsolut=c outAbsolut=c

root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i
FastEthernet1 -b 100 -u m -w 85 -c 98

Total RX Bytes: 1151.45 MB, Total TX Bytes: 503.72 MB<br>Average Traffic: 1.10 kB/s (0.0%) in,
1.01 kB/s (0.0%) out|inUsage=0.0%;85;98 outUsage=0.0%;85;98 inAbsolut=1179068c
outAbsolut=515796c

root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i
FastEthernet2 -b 100 -u m -w 85 -c 98

Total RX Bytes: 3748.07 MB, Total TX Bytes: 0.00 MB<br>Average Traffic: 0.44 kB/s (0.0%) in, -
0.88 kB/s (-0.0%) out|inUsage=0.0%;85;98 outUsage=-0.0%;85;98 inAbsolut=2329162c
outAbsolut=3028547c

root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i
FastEthernet3 -b 100 -u m -w 85 -c 98

Total RX Bytes: 342.67 MB, Total TX Bytes: 969.90 MB<br>Average Traffic: 0.13 kB/s (0.0%) in,
0.30 kB/s (0.0%) out|inUsage=0.0%;85;98 outUsage=0.0%;85;98 inAbsolut=350885c
outAbsolut=993162c

```

Το αποτέλεσμα αυτό μας επιβεβαιώνει ότι έχουμε κίνηση και στις τρεις θύρες Ethernet του Router.

Για τις ανάγκες αλλά και για την ευκολία των δοκιμών μας θα ελέγχουμε μόνο μια θύρα Ethernet. Αργότερα εφόσον έχουμε βγάλει αξιόπιστα αποτελέσματα μπορούμε να εντάξουμε και τις υπόλοιπες θύρες στον έλεγχο μέσω του script. Επομένως για τις ανάγκες των δοκιμών μας από εδώ και πέρα θα χρησιμοποιούμε την θύρα Fa1 του Router η οποία στο Icinga έχει την ονομασία FastEthernet1.

Σαν πρώτη δοκιμή βάζουμε ένα αρχείο να κατεβαίνει για να δούμε την κίνηση περνάει από την θύρα Ethernet. Η κίνηση είναι σταθερή οπότε δίνοντας συνεχόμενα την εντολή ελέγχου θα μπορούμε να βλέπουμε τις διακυμάνσεις του Bandwidth. Στο παρακάτω φαίνονται τα αποτελέσματα.

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 85 -c 98
```

Total RX Bytes: 155.28 MB, Total TX Bytes: 225.56 MB
Average Traffic: 3.62 MB/s (28.9%) in, 179.91 kB/s (1.4%) out | inUsage=28.9%;85;98 outUsage=1.4%;85;98 inAbsolut=147895c outAbsolut=230437c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 85 -c 98
```

Total RX Bytes: 177.59 MB, Total TX Bytes: 226.67 MB
Average Traffic: 5.58 MB/s (44.6%) in, 284.12 kB/s (2.2%) out | inUsage=44.6%;85;98 outUsage=2.2%;85;98 inAbsolut=159005c outAbsolut=230976c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 85 -c 98
```

Total RX Bytes: 200.78 MB, Total TX Bytes: 227.81 MB
Average Traffic: 5.80 MB/s (46.4%) in, 290.65 kB/s (2.3%) out | inUsage=46.4%;85;98 outUsage=2.3%;85;98 inAbsolut=181856c outAbsolut=232112c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 85 -c 98
```

Total RX Bytes: 256.75 MB, Total TX Bytes: 230.46 MB
Average Traffic: 5.57 MB/s (44.6%) in, 274.68 kB/s (2.1%) out | inUsage=44.6%;85;98 outUsage=2.1%;85;98 inAbsolut=240082c outAbsolut=234889c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 85 -c 98
```

Total RX Bytes: 335.06 MB, Total TX Bytes: 233.94 MB
Average Traffic: 6.02 MB/s (48.2%) in, 274.63 kB/s (2.1%) out | inUsage=48.2%;85;98 outUsage=2.1%;85;98 inAbsolut=262910c outAbsolut=235987c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 85 -c 98
```

Total RX Bytes: 549.14 MB, Total TX Bytes: 243.40 MB
Average Traffic: 5.68 MB/s (45.4%) in, 252.66 kB/s (2.0%) out | inUsage=45.4%;85;98 outUsage=2.0%;85;98 inAbsolut=411194c outAbsolut=242670c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 85 -c 98
```

Total RX Bytes: 625.69 MB, Total TX Bytes: 246.58 MB
Average Traffic: 4.87 MB/s (39.0%) in, 208.10 kB/s (1.6%) out | inUsage=39.0%;85;98 outUsage=1.6%;85;98 inAbsolut=595798c outAbsolut=250620c

Από τους διαδοχικούς ελέγχους μέσω της εντολής βλέπουμε τις διακυμάνσεις στην ταχύτητα. Αυτός όμως που παρατηρείται είναι ότι το ποσοστό που μας δίνει σαν αποτέλεσμα για την πληρότητα της γραμμής μας δεν αντιστοιχεί στην πραγματικότητα. Όπως έχουμε προαναφέρει οι δοκιμές μας γίνονται σε μια VDSL γραμμή προδιαγραφών 50Mbps στο downlink και 5Mbps στο uplink. Επομένως για να μας εμφανίζονται ορθά οι ενημερώσεις για τα Warning και Critical θα πρέπει να φτιάξουμε τις αντίστοιχες παραμέτρους στην εντολή μας. Η μέγιστη ταχύτητα που μπορεί να περάσει στην θύρα Ethernet μέσω της VDSL γραμμής μας είναι περίπου 5,80MB/s στο downlink, ενώ στο uplink 500KB/s. Άρα σαν warning μπορούμε ορίσουμε το 28% της μέγιστης ταχύτητας της θύρας Ethernet και σαν Critical να ορίσουμε το 42% της μέγιστης ταχύτητας της θύρας Ethernet. Αυτά τα ποσοστά αντιστοιχούν στις -w και -c παραμέτρους στην εντολή του Icinga.

Άρα βάση των παραπάνω η εντολή για τον έλεγχο της θύρας Ethernet θα έχει την παρακάτω διατύπωση.

```
./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 28 -c 42
```

Επομένως δίνοντας την εντολή με τις καινούργιες παραμέτρους παίρνουμε τα παρακάτω αποτελέσματα.

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 28 -c 42
```

Total RX Bytes: 3665.63 MB, Total TX Bytes: 1746.07 MB
Average Traffic: 4.95 MB/s (39.6%) in, 236.13 kB/s (1.8%) out
WARNING bandwidth utilization.

|inUsage=39.6%;28;42 outUsage=1.8%;28;42 inAbsolut=3621952c outAbsolut=1781838c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 28 -c 42
```

Total RX Bytes: 3739.03 MB, Total TX Bytes: 1749.56 MB
Average Traffic: 5.65 MB/s (45.2%) in, 274.73 kB/s (2.1%) out
CRITICAL bandwidth utilization.

|inUsage=45.2%;28;42 outUsage=2.1%;28;42 inAbsolut=3753609c outAbsolut=1787977c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 28 -c 42
```

Total RX Bytes: 3750.62 MB, Total TX Bytes: 1750.20 MB
Average Traffic: 3.86 MB/s (30.9%) in, 217.39 kB/s (1.7%) out
WARNING bandwidth utilization.

|inUsage=30.9%;28;42 outUsage=1.7%;28;42 inAbsolut=3828762c outAbsolut=1791548c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 28 -c 42
```

Total RX Bytes: 3782.23 MB, Total TX Bytes: 1751.69 MB
Average Traffic: 5.00 MB/s (40.0%) in, 257.80 kB/s (2.0%) out
WARNING bandwidth utilization.

|inUsage=40.0%;28;42 outUsage=2.0%;28;42 inAbsolut=3862773c outAbsolut=1793217c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 28 -c 42
```

Total RX Bytes: 3803.14 MB, Total TX Bytes: 1752.72 MB
Average Traffic: 4.83 MB/s (38.6%) in, 257.03 kB/s (2.0%) out
WARNING bandwidth utilization.

|inUsage=38.6%;28;42 outUsage=2.0%;28;42 inAbsolut=3884529c outAbsolut=1794276c

```
root@debian9:/usr/lib/nagios/plugins# ./check_iftraffic.pl -H 192.168.2.100 -C Icinga -i FastEthernet1 -b 100 -u m -w 28 -c 42
```

Total RX Bytes: 3933.01 MB, Total TX Bytes: 1758.77 MB
Average Traffic: 5.78 MB/s (46.3%) in, 319.84 kB/s (2.5%) out
CRITICAL bandwidth utilization.

|inUsage=46.3%;28;42 outUsage=2.5%;28;42 inAbsolut=4015564c outAbsolut=1800339c

Παρατηρώντας τα αποτελέσματα βλέπουμε ότι πλέον έχουμε τις ενημερώσεις που θέλουμε βάσει της κίνησης της VDSL γραμμής την οποία δοκιμάζουμε. Οι προηγούμενες τιμές των παραμέτρων `-w` και `-c` όπου ήταν 85 και 98 αντίστοιχα θα λειτουργούσαν άψογα εάν η ταχύτητα της γραμμής μας ήταν συμμετρική με χαρακτηριστικά 100 Mbps στο downlink και 100 Mbps στο uplink.

4.3 check_iftraffic configuration

Βήμα 1°

To object command για το icingaweb2.

Εντολή για αρχείο commands.

```
root@debian9:~# nano /etc/icinga2/conf.d/commands.conf
```

```
object CheckCommand "traffic-int" {
    import "plugin-check-command"
    command = [ PluginDir + "/check_iftraffic.pl" ]

    arguments = {
        "-H" = "$address$"
        "-C" = "$cn$"
        "-i" = "$int$"
        "-w" = "$snmp_warn$"
        "-c" = "$snmp_crit$"
        "-b" = "100"
        "-u" = "m"
    }
    vars.cn = COMN
}
```

Στο Object command χρησιμοποιήσαμε τα arguments:

-H --> host IP ADDRESS

-C --> community string

-i --> interface description OID

-w --> warning constant

-c --> critical constant

-b --> bandwidth integer

-u --> unit

Βήμα 2°

Είναι η παραμετροποίηση του service στο icingaweb2.

Η εντολή για να παραμετροποιήσουμε το service αρχείο το icingaweb2.

```
root@debian9:~# nano /etc/icinga2/conf.d/services.conf
```

```

apply Service "traffic_int:" for (int => config in host.vars.int) {
  import "generic-service"
  check_command = "traffic-int"

  vars += config
  assign where host.address
}

```

Βήμα 3°

Είναι η παραμετροποίηση του host στο icingaweb2.

Η εντολή για να παραμετροποιήσουμε το host αρχείο στο icingaweb2.

```
root@debian9:~# nano /etc/icinga2/conf.d/hosts.conf
```

```

object Host "TEST_ROUTER" {
  import "generic-host"
  address = "192.168.2.100"
  vars.os = "Unknown"
  vars.type = "Unknown"
  vars.os_details = "Unknown"
  vars.description = "Cisco IOS Software, C800 Software (C800-
UNIVERSALK9-M), Version 15.3(3)M5, RELEASE SOFTWARE (fc3)^M"
  vars.ports = "Unknown"
  vars.location = "HOME"
  vars.snmp_warn = "28"
  vars.snmp_crit = "42"
  vars.vendor = "ciscoSystems"
  notes = "Technical Support: http://www.cisco.com/techsupport^M"
  vars.int["ATM0"] = {
    int = "ATM0"
  }
  vars.int["Ethernet0"] = {
    int = "Ethernet0"
  }
  vars.int["FastEthernet0"] = {
    int = "FastEthernet0"
  }
  vars.int["FastEthernet1"] = {
    int = "FastEthernet1"
  }
  vars.int["FastEthernet2"] = {
    int = "FastEthernet2"
  }
  vars.int["FastEthernet3"] = {
    int = "FastEthernet3"
  }
}

```

Κεφάλαιο 5

Τα `check_snmp_int` και `check_iftraffic` στο περιβάλλον του Icinga Web2

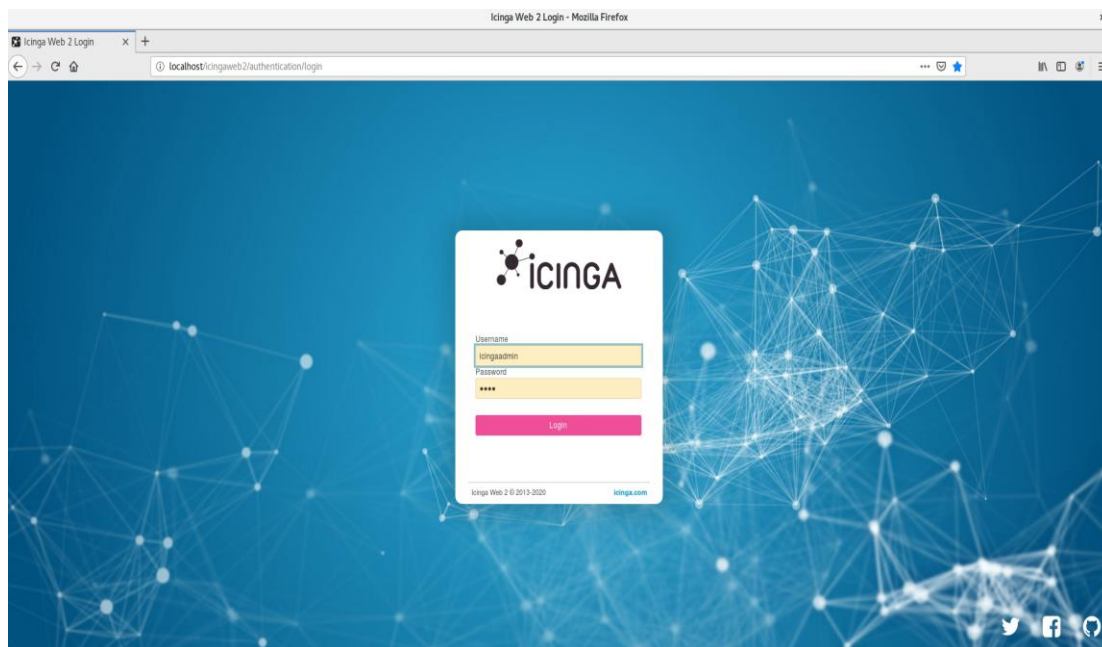
Κάνοντας login στον localhost για το icinga Web2 <http://localhost/icingaweb2/authentication/login> εισερχόμαστε στο περιβάλλον UI (user interface) του Icinga Web 2.

Το Icinga Web 2 έχει ένα ισχυρό πλαίσιο PHP για web εφαρμογές που προσφέρεται με καθαρό και μειωμένο σχεδιασμό. Είναι γρήγορο, ευαίσθητο, προσβάσιμο και εύκολα επεκτάσιμο με τις μονάδες.

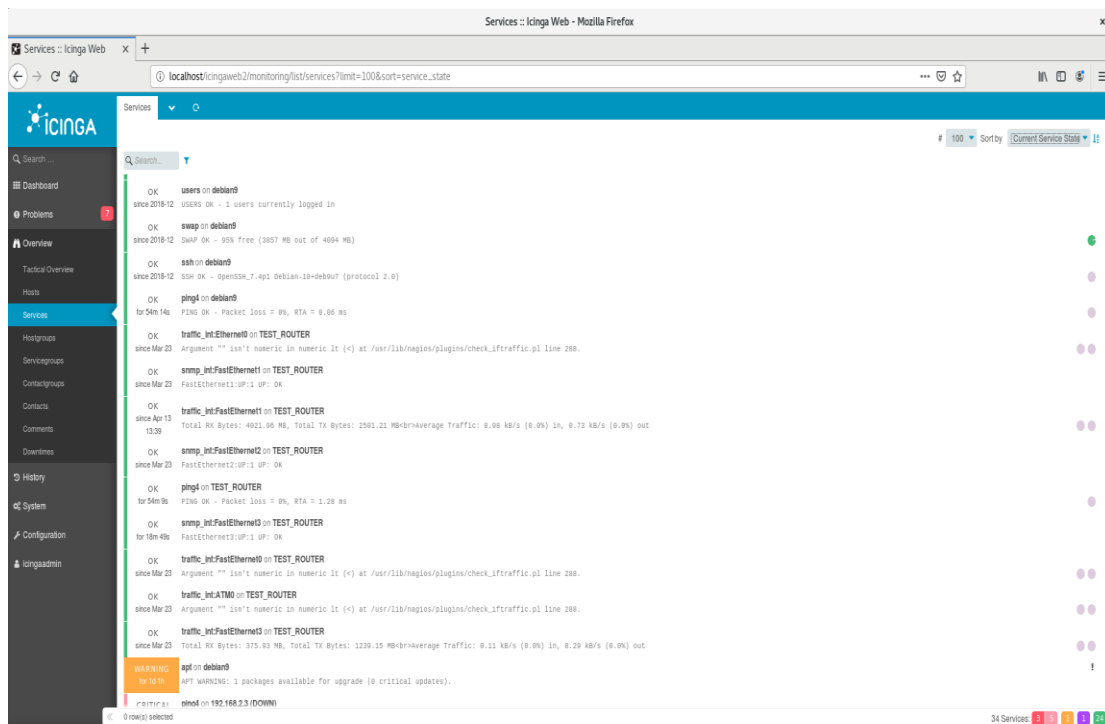
Αυτή είναι η βασική ενότητα για τους περισσότερους χρήστες του Icinga Web 2.

Παρέχει μια έξυπνη διεπαφή χρήστη για παρακολούθηση με το Icinga 2. Ειδικά υπάρχουν πολλές προβολές λίστας και λεπτομερειών (π.χ. για κεντρικούς υπολογιστές και υπηρεσίες) που μπορείτε να ταξινομήσετε και να φιλτράρετε ανάλογα με το τι θέλετε να δείτε.

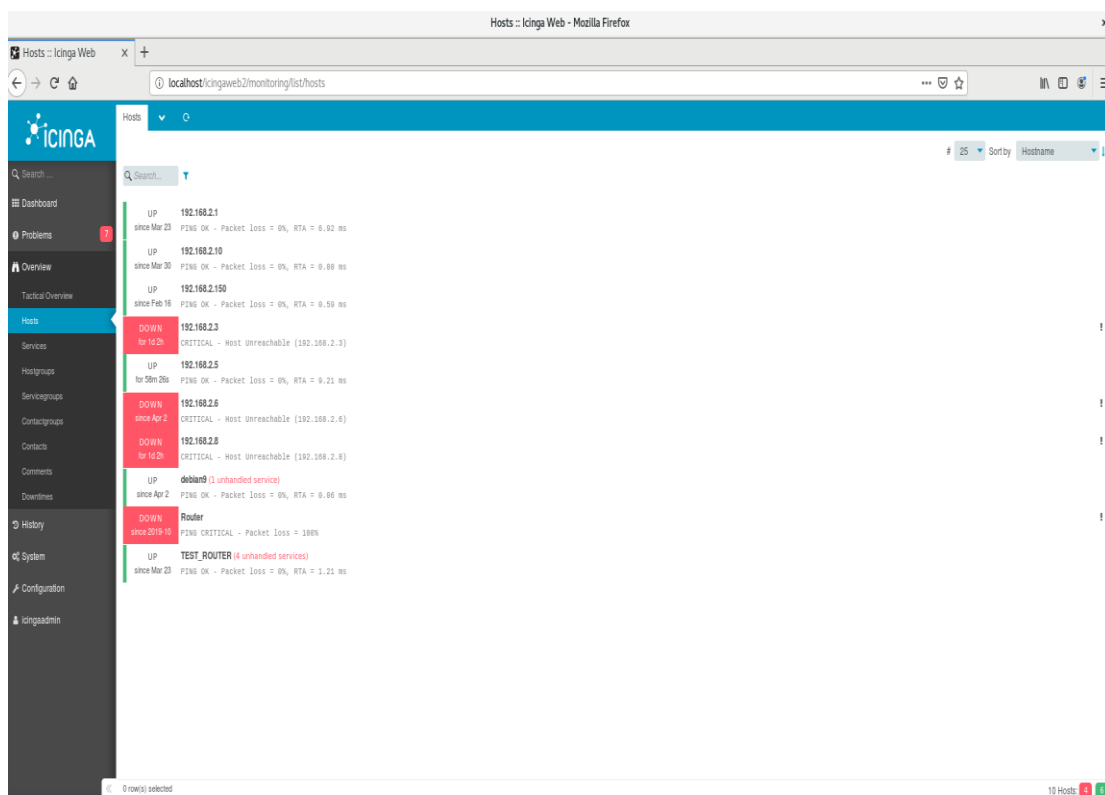
Μπορείτε επίσης να ελέγξετε την ίδια την διαδικασία παρακολούθησης στέλνοντας εξωτερικές εντολές στο Icinga. Οι περισσότερες από αυτές τις ενέργειες (όπως η αναδιάταξη ενός ελέγχου) μπορούν να γίνουν με ένα μόνο κλικ.



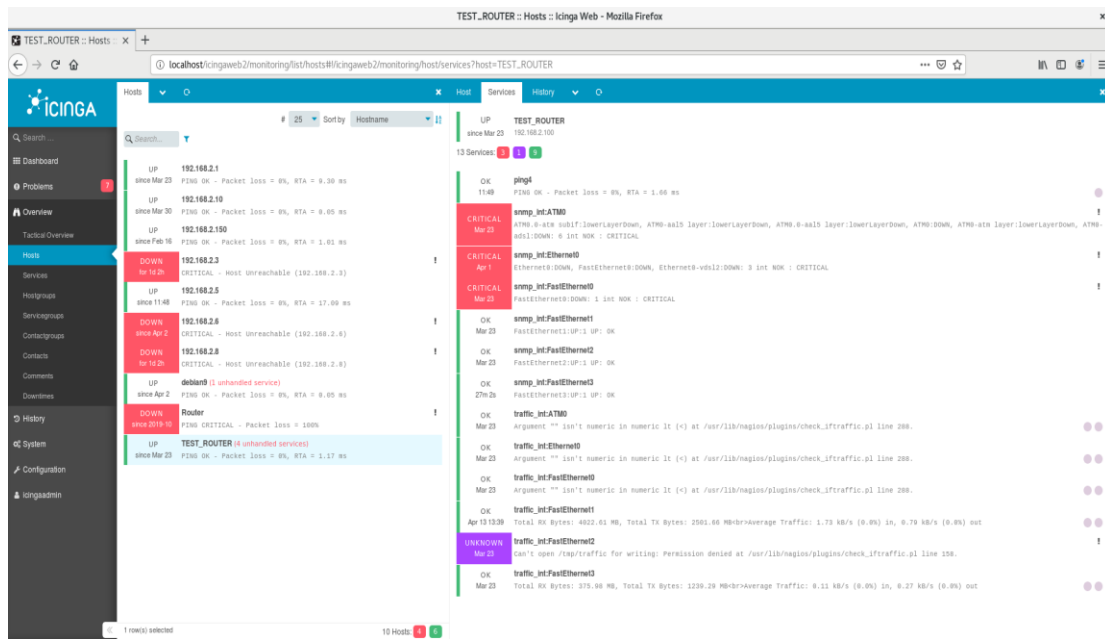
Εικόνα 33. Εισαγωγική φόρμα στο Icinga Web 2.



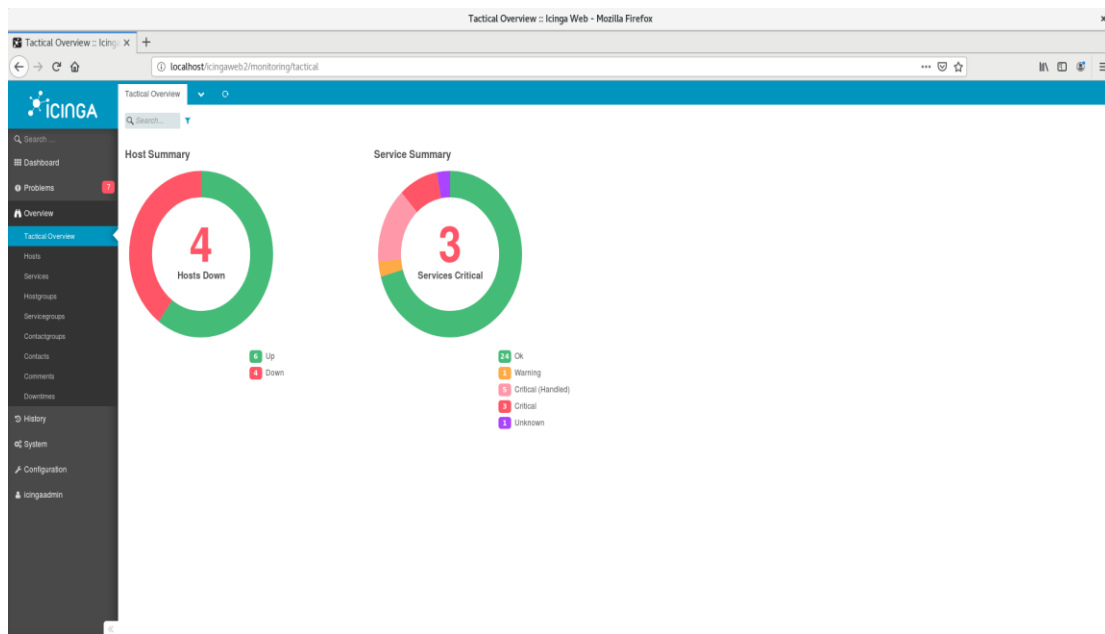
Εικόνα 34. Τα services που είναι σε λειτουργία στο Icinga Web 2.



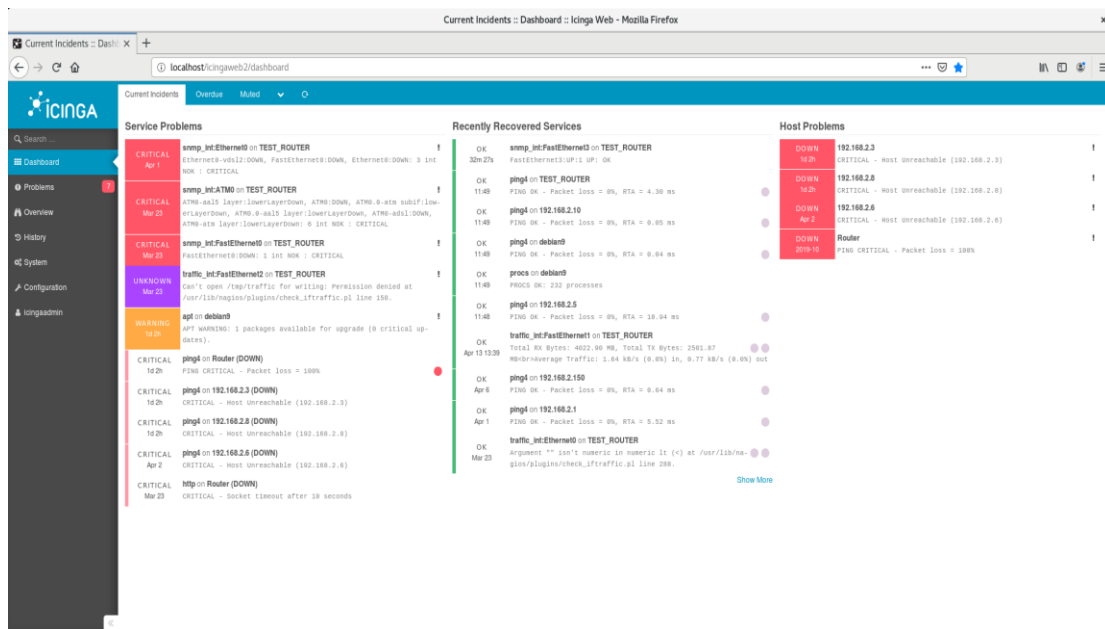
Εικόνα 35. Οι Hosts στο Icinga Web2.



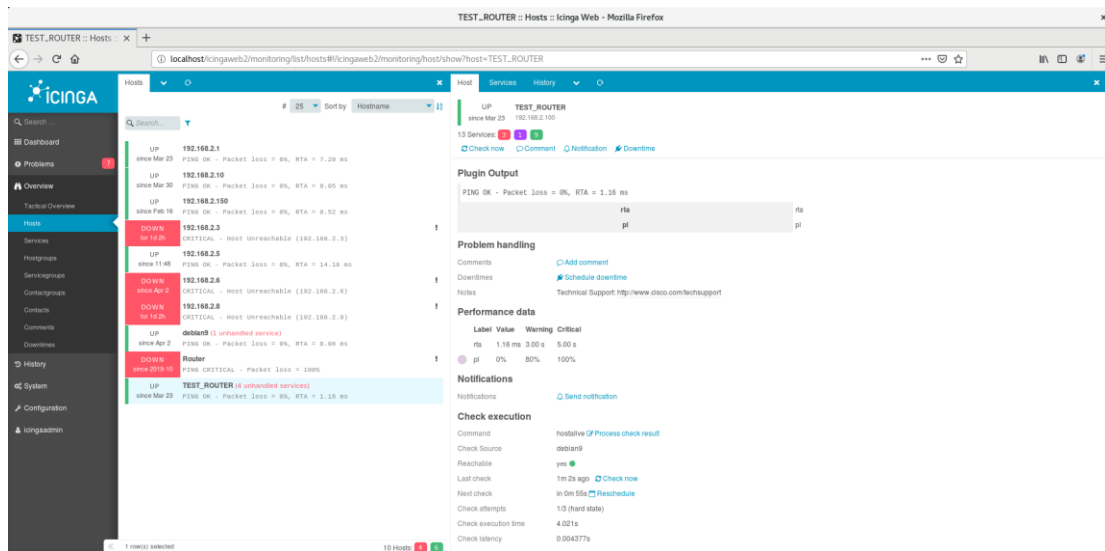
Εικόνα 36. Τα services για τον TEST_ROUTER



Εικόνα 37. Το Tactical Overview του Icinga Web 2.



Εικόνα 38. Τα incidents του Icinga Web 2.



Εικόνα 39. TEST_ROUTER Host

Συμπεράσματα

Καθημερινά η τεχνολογία γίνεται ολοένα και πιο πολύπλοκη το ίδιο και η ασφάλεια των δικτύων των υπολογιστών μας. Ο έλεγχος ασφάλειας σε αυτά καθώς η αποτελεσματικότητα και ακεραιότητα τους είναι το σημαντικότερο τμήμα τόσο για ένα σύστημα επιτήρησης δικτύου όσο για τους ανθρώπους που είναι επιφορτισμένοι με αυτό.

Λαμβάνοντας υπόψη ότι οι εφαρμογές ανοιχτού κώδικα έχουν φανεί αξιόπιστες και ανταγωνίζονται σε μεγάλο βαθμό, τις εφαρμογές που απαιτείται καταβολή χρηματικού ποσού για την απόκτηση τους η επιλογή ανοιχτού κώδικα λογισμικού ήταν η καλύτερη δυνατή.

Επιλέγοντας έτσι το IcingaWeb2 μας δόθηκε η δυνατότητα να έρθουμε σε επαφή με την δημιουργία ενός τέτοιου συστήματος διαχείρισης δικτύου ανοικτού λογισμικού από την αρχή. Τα στάδια της εγκατάστασης της διαμόρφωσης του και της λειτουργίας του μας έφεραν αντιμέτωπους με διάφορα τεχνικά ζητήματα που μας έκαναν να κατανοήσουμε ακόμα περισσότερο την αρχιτεκτονική του.

Στην παρούσα μελέτη καταφέραμε να δημιουργήσουμε και να παραμετροποιήσουμε δύο διαφορετικά plugins, το `check_snmp_int` και το `check_iftraffic` που τα εφαρμόσαμε σε ένα δοκιμαστικό Router.

Με το `check_snmp_int` μπορούσαμε να ελέγχουμε την λειτουργικότητα της εκάστοτε θύρας Ethernet, ενώ με το `check_iftraffic` μπορούσαμε να ελέγξουμε την κίνηση που περνάει από την εκάστοτε θύρα Ethernet και να ορίσουμε τα αντίστοιχα threshold ενημέρωσης.

Συμπερασματικά το πλεονέκτημα αναβαθμίσεων και η προσθήκη νέων δυνατοτήτων στο Icinga web 2 αλλά και η τεχνογνωσία που αποκτούμε όσο το χρησιμοποιούμε καθιστά το σύστημα ευκολότερο στην επίλυση προβλημάτων και αποδοτικότερο.

Βιβλιογραφία

1. General Information.

<https://icinga.com/>

2. Monitoring Basics

<https://icinga.com/docs/icinga2/latest/doc/03-monitoring-basics/#notification-commands>

3. Debian Installation

<https://debian-handbook.info/browse/stable/sect.installation-steps.html>

4. Icinga WEB 2 Installation

<https://www.linuxtechi.com/install-configure-icinga2-centos-7-rhel-7/>

5. Check_cas_auth

https://exchange.nagios.org/directory/Plugins/Websites,-Forms-and-Transactions/check_cas_auth/details

6. Check_logfiles

https://labs.consol.de/nagios/check_logfiles/

7. Network Management System

<https://www.pcwld.com/best-network-monitoring-tools-and-software>

8. Central Authentication Service(CAS)

https://en.wikipedia.org/wiki/Central_Authentication_Service

9. Oracle VirtualBox

<https://www.virtualbox.org/wiki/Downloads>

10. Network management system list

<https://www.pcwld.com/best-network-monitoring-tools-and-software>

11. Icinga instructions

<https://www.terena.org/activities/tf-noc/meeting12/slides/20150408-icinga.pdf>

12. Check_http

https://www.monitoring-plugins.org/doc/man/check_http.html

https://nagios-plugins.org/doc/man/check_http.html

13. Icinga Troubleshooting

<https://icinga.com/docs/icinga2/latest/doc/15-troubleshooting>

14. Icinga Network Monitoring – Viranch Mehta by Packt Publishing

Παράρτημα

A) Παραμετροποίηση TEST_ROUTER

```
TEST_ROUTER#sh run
```

```
Building configuration...
```

```
Current configuration : 3315 bytes
```

```
!
```

```
! Last configuration change at 17:11:44 GMT+2 Wed Apr 1 2020 by adroul
```

```
version 15.3
```

```
service timestamps debug datetime msec
```

```
service timestamps log datetime msec
```

```
no service password-encryption
```

```
!
```

```
hostname TEST_ROUTER
```

```
!
```

```
boot-start-marker
```

```
boot-end-marker
```

```
!
```

```
aqm-register-fnf
```

```
!
```

```
aaa new-model
```

```
!
```

```
aaa authentication login default local enable
```

```
aaa authorization console
```

```
aaa authorization exec default local
```

```
!
```

```
aaa session-id common
```

memory-size iomem 10

clock summer-time GMT+2 recurring last Sun Mar 3:00 last Sun Oct 4:00

!

multilink bundle-name authenticated

!

license udi pid C887VA-K9 sn FCZ1924C254

!

username adroul privilege 15 secret 5 \$1\$dgZQ\$fAJL9RzLYPvBpiv3HF3Yx.

!

controller VDSL 0

!

interface ATM0

no ip address

shutdown

no atm ilmi-keepalive

!

interface Ethernet0

no ip address

shutdown

!

interface FastEthernet0

no ip address

!

interface FastEthernet1

description CONNECTION TO ROUTER

no ip address

!

```
interface FastEthernet2

description CONNECTION TO PC

no ip address

!

interface FastEthernet3

description CONNECTION TO RASPBERRY

no ip address

!

interface Vlan1

description LAN

ip address 192.168.2.100 255.255.255.0

!

ip forward-protocol nd

no ip http server

no ip http secure-server

!

ip route 0.0.0.0 0.0.0.0 192.168.2.1

!

ip access-list standard management

permit 192.168.2.10

permit 192.168.2.150

!

snmp-server community Icinga RO 10

access-list 10 remark ICINGA

access-list 10 permit 192.168.2.10

access-list 10 permit 192.168.2.150

!
```

control-plane

!

mgcp behavior rsip-range tgcp-only

mgcp behavior comedia-role none

mgcp behavior comedia-check-media-src disable

mgcp behavior comedia-sdp-force disable

!

mgcp profile default

!

banner login ^C

* This system is for the use of authorized personel only. *

* Individuals using this computer system without authority, or in *

* excess of their authority, are subject to having all of their *

* activities on this system monitored and recorded by system *

* personnel. *

*

*

* In the course of monitoring individuals improperly using this system, *

* or in the course of system maintenance, the activities of authorized *

* users may also be monitored. *

*

*

* Anyone using this system expressly consents to such monitoring and is *

* advised that if such monitoring reveals possible evidence of criminal *

* activity, system personnel may provide the evidence of such monitoring*

* to law enforcement officials. *

```
*
TEST_ROUTER
*
*****
^C
!
line con 0
logging synchronous
no modem enable
line aux 0
line vty 0 4
access-class management in
logging synchronous
transport input all
!
scheduler allocate 20000 1000
!
end
```