



**Χαροκόπειο Πανεπιστήμιο**

**Τμήμα Πληροφορικής & Τηλεματικής**

ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΕΙΔΙΚΕΥΣΗΣ

( ΜΔΕ )

Τηλεπικοινωνιακά Δίκτυα και Υπηρεσίες Τηλεματικής

## **Διπλωματική Εργασία**

Μελέτη , Εγκατάσταση, Παραμετροποίηση και  
Πιλοτική Λειτουργία ενός Network Management  
System

**Ζωή Φαρμάκη**

ΑΜ : 14210

Επιβλέπων Καθηγητής

**Δαλάκας Βασίλειος**

Μέλη της εξεταστικής επιτροπής

**Αναγνωστόπουλος Δημοσθένης, Νικολαΐδου Μάρα**

Αθήνα 2016



## Πρόλογος

Η παρούσα διπλωματική εργασία υλοποιήθηκε στα πλαίσια του Προγράμματος Μεταπτυχιακών Σπουδών του τμήματος Πληροφορικής και Τηλεματικής στο Χαροκόπειο Πανεπιστήμιο Αθηνών από τον Οκτώβρη του 2015 έως τον Φεβρουάριο του 2016.

Στα πλαίσια της διπλωματικής εργασίας ήρθα σε επαφή με ένα σύνολο από τεχνολογίες αλλά κυρίως με ένα σύνολο ανθρώπων τους οποίους θα ήθελα να ευχαριστήσω για την ηθική και τεχνική συμπαράσταση που μου παρείχαν.

Ειδικότερα θα ήθελα να ευχαριστήσω τον καθηγητή μου Δρ. Βασίλειο Δαλάκα, μέλος του Κέντρου Πληροφορικής & Δικτύων ο οποίος με εμπιστεύτηκε και με καθοδήγησε καθ' όλη της διάρκεια αυτής της προσπάθειας.

Η εργασία αυτή ολοκληρώθηκε με την αμέριστη συμπαράσταση και την ηθική στήριξη της οικογένειας μου και το ευχάριστο και δημιουργικό περιβάλλον στον Χαροκόπειο Πανεπιστήμιο.

*Στους Γονείς μου*

# Περιεχόμενα

<b>Πρόλογος .....</b>	<b>3</b>
<b>Περιεχόμενα .....</b>	<b>5</b>
<b>Κατάλογος πινάκων.....</b>	<b>7</b>
<b>Περίληψη .....</b>	<b>9</b>
<b>Λέξεις κλειδιά .....</b>	<b>10</b>
<b>Abstract .....</b>	<b>11</b>
<b>Keywords.....</b>	<b>12</b>
<b>1. Εισαγωγή .....</b>	<b>13</b>
1.1 Τί είναι ένα network management system.....	13
1.2 Από τι αποτελείται το μοντέλο διαχειριστή-αντιπροσώπου.....	14
1.3 Πρωτόκολλα διαχείρισης δικτύου (γενικά) και περιοχές διαχείρισης (management domains) ....	16
1.4 Αρχιτεκτονικές διαχείρισης.....	16
1.5 Το πρωτόκολλο SNMP .....	21
<b>2. Λειτουργικές απαιτήσεις – Επιλογή Λογισμικού .....</b>	<b>24</b>
2.1 Λειτουργικές απαιτήσεις του Κέντρου Πληροφορικής και Δικτύων .....	24
2.2 Κατάλληλο λογισμικό .....	25
2.3 Δημοφιλέστερα λογισμικά ανοιχτού κώδικα .....	25
2.4 Κατάλληλο λογισμικό .....	30
2.5 Σύντομη ιστορική αναδρομή για το icinga web 2 .....	30
2.6 Αρχιτεκτονική συστήματος icinga web 2.....	32
<b>3. Εγκατάσταση.....</b>	<b>34</b>
3.1 Περιβάλλον αρχικής εγκατάστασης .....	34
3.2 Διαδικασία εγκατάστασης.....	34
3.2.1 Διαδικασία εγκατάστασης με screenshot .....	38
3.2.2 Έλεγχος διαδικασίας εγκατάστασης.....	40
3.2.3 Παρεμετροποίηση Διαμόρφωση Σύνδεση με LDAP(Lightweight Directory Access Protocol) .....	41
<b>4. Παραμετροποίηση .....</b>	<b>44</b>
4.1 Διαδικασία απομακρυσμένης παρακολούθησης .....	44
4.1.2 Παράδειγμα monitoring εξυπηρετητή .....	44

4.1.3 Παράδειγμα monitoring δικτυακής συσκευής.....	48
4.2 Διαμόρφωση διαδικασίας ειδοποίησης μέσω email.....	50
4.3 Χρονοπρογραμματισμός ειδοποιήσεων μέσω email.....	52
4.4 Το conf.d directory γενικότερα .....	53
4.5 Αναβάθμιση.....	64
4.6 Απεγκατάσταση.....	64
<b>5. Πιλοτική Λειτουργία .....</b>	<b>66</b>
5.1 Η υποστηριζόμενη πλατφόρμα.....	66
5.2 Διαμόρφωση του web interface μέσα από το conf.d.....	70
5.3 Εύχρηστες υπηρεσίες πληροφόρησης μέσα από το web interface .....	80
5.3.1 Η καρτέλα History .....	80
5.3.2 Η καρτέλα Reporting.....	83
5.3.3 Η καρτέλα System.....	84
5.3.4 Η καρτέλα Configuration .....	85
<b>6. Συμπεράσματα.....</b>	<b>87</b>
<b>Παράρτημα Α.....</b>	<b>89</b>
<b>Παράρτημα Β.....</b>	<b>99</b>
Εγκατάσταση Exim internet mailer.....	99
<b>Παράρτημα Γ .....</b>	<b>101</b>
Αλλαγή λογότυπου στον web interface.....	101
<b>Βιβλιογραφία .....</b>	<b>103</b>

## Κατάλογος πινάκων

Εικόνα 1: Διαχειριστής - Αντιπρόσωπος στο πρωτόκολλο SNMP .....	15
Εικόνα 2: Κεντρική Αρχιτεκτονική Διαχείρισης.....	17
Εικόνα 3: Ιεραρχική Αρχιτεκτονική Διαχείρισης.....	19
Εικόνα 4: Κατανεμημένη Αρχιτεκτονική Διαχείρισης.....	20
Εικόνα 5: Αρχιτεκτονική SNMP πρωτοκόλλου .....	22
Εικόνα 6: Διαδικασία ανταλλαγής δεδομένων στο πρωτόκολλο SNMP .....	23
Εικόνα 7: Αρχιτεκτονική icinga 2 .....	32
Εικόνα 8: Αρχιτεκτονική icinga2_(ii) .....	33
Εικόνα 9: Είσοδος στο icinga web 2 .....	39
Εικόνα 10: Εντοπισμός στοιχείων λογαριασμού.....	41
Εικόνα 11: Παραμετροποίηση LDAP Server Hua .....	43
Εικόνα 12: Αρχικό περιβάλλον του συστήματος διαχείρισης.....	66
Εικόνα 13: host.conf σε συνδυασμό με το web interface.....	71
Εικόνα 14: services.conf σε συνδυασμό με το web interface .....	72
Εικόνα 15: users.conf .....	73
Εικόνα 16: users.conf (ii) .....	73
Εικόνα 17: users.conf (iii) .....	74
Εικόνα 18: users.conf (iv) .....	75
Εικόνα 19: users.conf (v) .....	75
Εικόνα 20: Παράδειγμα προβλήματος σε κάποιον εξυπηρετητή .....	76
Εικόνα 21: Εισερχόμενη ειδοποιητήρια αλληλογραφία.....	76
Εικόνα 22: Τύπος ειδοποίησης προβλήματος .....	76
Εικόνα 23: Τύπος ειδοποίησης ανάκτησης του προβλήματος .....	77
Εικόνα 24: Ομαδοποίηση των servers.....	78
Εικόνα 25: Ομαδοποίηση των switch.....	78
Εικόνα 26: Ομαδοποιημένος έλεγχος μόνο για Dick Checks .....	78
Εικόνα 27: Ομαδοποιημένος έλεγχος μόνο για HTTP Checks .....	79
Εικόνα 28: Ομαδοποιημένος έλεγχος μόνο για Ping Checks.....	79
Εικόνα 29: History/ Event Grid.....	80
Εικόνα 30: History/ Event Grid (ii).....	81
Εικόνα 31: History/ Event Grid (iii).....	81
Εικόνα 32: History/ Timeline .....	82
Εικόνα 33: Reporting - Alert Summary .....	83
Εικόνα 34: System/ Monitoring Health.....	84
Εικόνα 35: Configuration/ Authorization.....	85
Εικόνα 36: Configuration/ Modules.....	86
Εικόνα 37: Επιλογή μοντέλου διαμόρφωσης.....	89
Εικόνα 38: Τύπος αυθεντικοποίησης .....	90
Εικόνα 39: Προτίμηση αποθήκευσης στην database .....	90
Εικόνα 40: Διαμόρφωση Βάσης δεδομένων για τους χρήστες .....	91
Εικόνα 41: Βήμα Αυθεντικοποίησης/επιβεβαίωσης .....	92
Εικόνα 42: Δημιουργία του προσωπικού μου λογαριασμού .....	92
Εικόνα 43: Ρυθμίσεις παραμέτρων εφαρμογής.....	93
Εικόνα 44: Επαλήθευση στοιχείων διαμόρφωσης_a.....	94
Εικόνα 45: Επαλήθευση στοιχείων διαμόρφωσης_b .....	94
Εικόνα 46: Ολοκλήρωση επιτυχής διαδικασίας διαμόρφωσης.....	95

Εικόνα 47: Επιλογή βάσης δεδομένων για ανάκτηση πληροφοριών παρακολούθησης .....	95
Εικόνα 48: Συμπλήρωση στοιχείων διαδικασίας σύνδεσης IDO-mysql με το icinga web 2 .....	96
Εικόνα 49: Τρόπος μεταφοράς εντολών .....	97
Εικόνα 50: Προστασία του δικτύου διαχείρισης.....	97
Εικόνα 51: Ανακεφαλαιοποίηση όσων προηγήθηκαν και επιβεβαίωση .....	98
Εικόνα 52: Επιτυχής εγκατάσταση του icinga web 2 στον υπολογιστή μου .....	98
Εικόνα 53: Διαμόρφωση Mail Server .....	99



## Περίληψη

Στο σημείο αυτό θα ήθελα να αναφέρω τον λόγο που επέλεξα το συγκεκριμένο θέμα. Ως Μεταπτυχιακή φοιτήτρια της ειδικότητας Τηλεπικοινωνιακά Δίκτυα και Υπηρεσίες Τηλεματικής (MSc in Telecommunication Networks and Telematic Services) και έχοντας παρακολουθήσει και το CCNA/Cisco ήθελα να εμβαθύνω πρακτικά στον κλάδο αυτό και γι' αυτό επέλεξα το συγκεκριμένο θέμα μέσα από το οποίο αποκόμισα πολλές γνώσεις τόσο θεωρητικές όσο και πρακτικές γύρω από τον τομέα των δικτύων. Σε αυτό το σημείο θα ήθελα να ευχαριστήσω ξανά τον επιβλέπων Καθηγητή μου Βασίλειο Δαλάκα ο οποίος μου έδωσε την ευκαιρία να εμβαθύνω περισσότερο μέσα από την σωστή του καθοδήγηση.

Όσο αναφορά το σκοπό της διπλωματικής μου, το Κέντρο Πληροφορικής και Δικτύων (ΚΠΔ) που βρίσκεται στο Χαροκόπειο πανεπιστήμιο για την πιο ομαλή λειτουργία του χρειάζεται ένα μη εμπορικό σύστημα διαχείρισης δικτύων το οποίο θα βοηθούσε στον εντοπισμό και την επίλυση προβλημάτων από τους διαχειριστές, έτσι ώστε να μην αντιμετωπίζουν δυσχέρειες, χωρίς αυξημένο κόστος λειτουργίας.

Αντικείμενο της διπλωματικής εργασίας αυτής είναι η εγκατάσταση, η παραμετροποίηση και η πιλοτική λειτουργία ενός συστήματος διαχείρισης δικτύου στο Χαροκόπειο Πανεπιστήμιο με σκοπό την καλύτερη οργάνωση και αποδοτικότερη διαδικασία επίλυσης τεχνικών προβλημάτων στο δίκτυο του Πανεπιστημίου.

Ειδικότερα, ο κύριος στόχος της εργασίας αυτής είναι να απλοποιηθεί η διαδικασία επίλυσης προβλημάτων καθώς και να διευκολυνθεί η δουλειά του προσωπικού του ΚΠΔ. Αυτό κρίθηκε απαραίτητο να υλοποιηθεί μέσω μιας υπηρεσίας που να είναι προσβάσιμη από το διαδίκτυο. Με τον τρόπο αυτό μπορεί να διευκολυνθεί η μέχρι τώρα εργασία του προσωπικού και να οργανωθεί πολύ πιο αποδοτικά. Επίσης, με την δυνατότητα συνεχούς καταγραφής που θα δίνει η υπηρεσία αυτή και την εξαγωγή στατιστικών και συμπερασμάτων θα προκύψει συνεχής βελτίωση των παρεχόμενων υπηρεσιών.

Πλέον, πολύ σημαντικό εργαλείο στις μέρες μας είναι το λογισμικό ανοιχτού κώδικα το οποίο είναι ευρέως διαδεδομένο και υπάρχει μεγάλη ποικιλία επιλογής εφαρμογών. Οι δυνατότητες τους, τα χαρακτηριστικά τους και η αποδοτικότητα τους μπορούν να ανταγωνισθούν τα

εμπορικά λογισμικά που υπάρχουν. Όσον αφορά τα συστήματα που σκοπό τους έχουν την εξυπηρέτηση και επίλυση τεχνικών προβλημάτων, υπάρχουν πολλές διαθέσιμες επιλογές σε εφαρμογές ανοιχτού κώδικα. Γι' αυτό τον λόγο και εμείς μετά από έρευνα επιλέξαμε μια τέτοια εφαρμογή της οποίας το κόστος θα ήταν μηδενικό και η αποδοτικότητα ίδια με τα περισσότερα λογισμικά αυτού του τύπου για τα οποία απαιτείται η καταβολή χρημάτων για την απόκτησή τους ή την συντήρησή τους.

Ολοκληρώνοντας λοιπόν, έγινε η μελέτη τέτοιων λογισμικών και αντιμετωπίστηκαν ζητήματα σχετικά με την εγκατάσταση και την παραμετροποίηση ενός network management system σε συνεργασία με ανθρώπους σχετικούς με την επίλυση τεχνικών προβλημάτων. Το σύστημα τέθηκε σε πιλοτική λειτουργία και σκοπεύει σύντομα να βοηθήσει στην καταγραφή αλλά και συγχρόνως στην επίλυση τεχνικών προβλημάτων του Πανεπιστημίου μας.

## **Λέξεις κλειδιά**

ICINGA web2, VirtualBox, εγκατάσταση, παραμετροποίηση, επίλυση τεχνικών προβλημάτων, CLI( command-line-interface), τεχνικά ζητήματα, δίκτυα, debian8.

## **Abstract**

As a postgraduate student of the Telecommunication Networks and Telematics Services course (MSc in Telecommunication Networks and Telematics Services), and having attended the CCNA / CISCO, I was eager to gain both theoretical and practical knowledge in the field and in parallel acquire some practical knowledge about open source software. This is the key reason behind this thesis. At this point I would like to thank my supervisor Dr. Vasilis Dalakas who helped me during this process through his proper guidance.

For the operation of the Network Operations Centre (NOC), which is located in Harokopio University, the NOC needs a network management system (NMS), which would assist the administrators to identify and solve problems, in order for them not to face difficulties. Hence, the objective of this particular dissertation is the installation, parametrization (customization/ configuration) and pilot operation of an NMS at Harokopio University in order for the network of the University to be organized better and to be more efficient in the process of technical troubleshooting.

In particular, the main objective of this dissertation is to simplify the process of problem solving and also to facilitate the work of the staff of the NOC. This was considered necessary to be implemented through a service that is accessible from the internet. In that way, the work of the staff could be facilitated and organized much more effectively. In addition, due to the potentiality of continuous recording along with the exported statistics and conclusions that this service will provide, it can lead to constant improvement.

Contemporary open source software is a very important tool which is well-known and offers a great variety of applications. Their capabilities, their characteristics and their effective performance can compete with the commercial software available. Regarding the systems designed to provide service and technical NMS, there are many options available in open source applications. For this reason, after research, we have chosen such an application that will be of zero cost and its efficiency will be equal to most software of this type/kind, for which payment is required.

To conclude, the study of such software has been made, and issues –regarding installation and parametrization (customization/configuration) of a network management system– have been faced, in cooperation with people occupied with solving technical problems. The system has been set into pilot operation and will soon assist to the recording, as well as to the technical troubleshooting of our University.

## **Keywords**

ICINGA WEB 2, Linux, Debian8, VirtualBox, CLI (command-line-interface) installation, parametrization (customization/configuration), technical troubleshooting, technical issues, networks.

# 1. Εισαγωγή

## 1.1 Τί είναι ένα network management system

Ο όρος Network Management System ή Σύστημα διαχείρισης δικτύου αναφέρεται στην παρακολούθηση και τη συντήρηση δικτύων υπολογιστών. Καλύπτει διάφορους τομείς όπως την ασφάλεια, την απόδοση και την αξιοπιστία ενός δικτύου. Με τον όρο «**Διαχείριση Δικτύου**» εννοούμε τη διαδικασία του αυτόματου (ή όσο το δυνατόν αυτοματοποιημένου) ελέγχου ενός οποιουδήποτε δικτύου υπολογιστών ώστε το κόστος συντήρησης του να είναι κατά το δυνατόν μικρότερο και η απόδοση του η καλύτερη δυνατή.

Το NMS μπορεί να χρησιμοποιηθεί για την παρακολούθηση δύο στοιχείων: υλικού και λογισμικού σε ένα δίκτυο. Καταγράφει συνήθως δεδομένα από απομακρυσμένα σημεία ενός δικτύου για να πραγματοποιήσει την κεντρική αναφορά σε ένα διαχειριστή συστήματος

Ένα σύστημα διαχείρισης δικτύου είναι χρήσιμο για: Network device discovery (Εξεύρεση συσκευής δικτύου), Network device monitoring (Παρακολούθηση συσκευής δικτύου), Network performance analysis (Ανάλυση επιδόσεων ενός δικτύου), Network device management (Διαχείριση συσκευής δικτύου), Intelligent notifications or customizable alerts (Ευφυείς ειδοποιήσεις).

Η ποιότητα διαχείρισης ενός Συστήματος διαχείρισης δικτύου περιλαμβάνει τα ακόλουθα χαρακτηριστικά :

*Εξοικονόμηση χρημάτων:* Μόνο ένας διαχειριστής συστήματος απαιτείται σε μια συγκεκριμένη τοποθεσία για την παρακολούθηση και διαχείριση ολόκληρου του δικτύου, η οποία κόβει τις δαπάνες πρόσληψης και εξοικονομεί κόστος.

*Εξοικονόμηση χρόνου:* Κάθε πάροχος παίρνει άμεση πρόσβαση σε όλα τα δεδομένα, όταν απαιτείται. Όλα τα μέλη της ομάδας μπορούν απλά να εισάγουν ή να ανακτήσουν τα δεδομένα χρησιμοποιώντας τους δικούς τους σταθμούς εργασίας. Ταυτόχρονα, η πρόσβαση τους μπορεί να ελέγχεται από τον διαχειριστή του δικτύου.

*Αύξηση της παραγωγικότητας:* Το NMS εντοπίζει ένα ζήτημα το συντομότερο δυνατό που αυτό συμβαίνει για να διασφαλίσει ότι δεν υπάρχει επιβράδυνση της παραγωγικότητας ή απώλεια δεδομένων. [1]

## **1.2 Από τι αποτελείται το μοντέλο διαχειριστή-αντιπροσώπου**

Αρχικά πρέπει να αναφερθεί ότι μοιάζει με το μοντέλο client-server και αποτελείται: Από τους διαχειριστές (**managers**), τους αντιπροσώπους (**agents**) – Πληρεξούσιους αντιπροσώπους (**proxy agents**), τα διαχειριζόμενα στοιχεία δικτύου (**Network element, NE**) και την βάση δεδομένων πληροφοριών διαχείρισης (**MIB, Management Information Base**).

Ο διαχειριστής (**manager**) είναι ένα πρόγραμμα (λογισμικό) που εκτελείται σε κάποιο μηχάνημα του δικτύου και το οποίο χρησιμοποιεί ο υπεύθυνος συντήρησης του δικτύου (**network administrator**) για να στείλει εντολές διαχείρισης. Οι εντολές διαχείρισης μπορούν για παράδειγμα να αλλάζουν ρυθμίσεις σε μια δικτυακή συσκευή (χωρίς να χρειάζεται να μετακινηθούμε στο σημείο που βρίσκεται η συσκευή αυτή) ή ακόμα και να ελέγχει την κατάσταση λειτουργίας ενός τμήματος του δικτύου από μακριά.

Ο αντιπρόσωπος (**agent**) είναι πρόγραμμα (λογισμικό) το οποίο βρίσκεται εγκατεστημένο σε κάθε διαχειριζόμενο στοιχείο δικτύου με σκοπό να καταστήσει δυνατή την επικοινωνία του με τον διαχειριστή. Η διαχείριση γίνεται με τον εξής τρόπο: Ο διαχειριστής (**manager**) στέλνει τις κατάλληλες εντολές διαχείρισης και ελέγχου μέσω του πρωτοκόλλου διαχείρισης δικτύου. Οι εντολές αυτές λαμβάνονται από τους αντιπροσώπους στους οποίους απευθύνονται. Οι αντιπρόσωποι εκτελούν τις εντολές αυτές στα διαχειριζόμενα στοιχεία δικτύου (**NE**) που ελέγχουν.

Από την άλλη οπτική ο πληρεξούσιος αντιπρόσωπος (**proxy agent**) είναι ένα πρόγραμμα αντιπροσώπου το οποίο εκτελείται σε μια δικτυακή συσκευή, αλλά δεν ελέγχει την ίδια συσκευή στην οποία εκτελείται αλλά κάποια άλλη. Ο πληρεξούσιος αντιπρόσωπος αναλαμβάνει να λάβει τις εντολές από κάποιο διαχειριστή και να επικοινωνήσει με την συσκευή που χρειάζεται διαχείριση, μετατρέποντας τις οδηγίες που έχει λάβει με κατάλληλο τρόπο ώστε να είναι κατανοητές από την συσκευή. Ο πληρεξούσιος αντιπρόσωπος χρησιμοποιείται όταν η συσκευή που θέλουμε να διαχειριστούμε δεν έχει η ίδια την ικανότητα να εκτελέσει πρόγραμμα αντιπροσώπου (δεν μπορεί να κατανοήσει η ίδια τα πρωτόκολλα διαχείρισης). Αυτό συμβαίνει γενικά σε συσκευές που δεν έχουν δυνατότητες δικής τους αυτόνομης επεξεργασίας (π.χ. τα modems).

Τα διαχειριζόμενα στοιχεία (**network element, NE**) με την σειρά τους είναι δικτυακές συσκευές που συναντάμε σε ένα τοπικό δίκτυο όπως γέφυρες, δρομολογητές, modems, επαναλήπτες κλπ. Πολλές από αυτές τις συσκευές έχουν δυνατότητα απομακρυσμένης διαχείρισης. Για παράδειγμα ένας δρομολογητής μπορεί να μας επιτρέψει να αλλάζουμε τις ρυθμίσεις του (π.χ. πίνακας δρομολόγησης) από κάποιο μηχάνημα του δικτύου χρησιμοποιώντας ιστοσελίδες. Γενικά η διαχείριση των NE γίνεται με την βοήθεια ειδικών πρωτοκόλλων που ονομάζονται πρωτόκολλα Διαχείρισης Δικτύου (**Network Management Protocols, NMP**). Σε ένα σύστημα διαχείρισης, οι διαχειριζόμενες συσκευές εκτελούν προγράμματα αντιπροσώπων τα οποία δέχονται τις εντολές διαχείρισης από τους διαχειριστές και αλλάζουν ανάλογα τις ρυθμίσεις των συσκευών που ελέγχουν, δίνουν αναφορά κατάστασης και λαθών κλπ.

Τέλος, την βάση δεδομένων πληροφοριών διαχείρισης (**MIB, Management Information Base**) μοιράζονται μεταξύ τους οι διαχειριστές και αντιπρόσωποι και η οποία περιέχει πληροφορίες σχετικά με τα διαχειριζόμενα στοιχεία δικτύου (NE). Η βάση πληροφοριών διαχείρισης περιέχει επίσης πληροφορίες που καθορίζουν και την δομή του περιεχομένου της διαχειριζόμενης πληροφορίας (Πρόκειται για μια κανονική βάση δεδομένων: Περιέχει και πίνακες που περιγράφουν την δομή των πινάκων που περιέχουν τις πληροφορίες της βάσης). Σχεδιαστικά η MIB απεικονίζεται με μορφή δέντρου ενώ τα περιεχόμενα της παριστάνονται από τα φύλλα του δέντρου.[11]



**Εικόνα 1: Διαχειριστής - Αντιπρόσωπος στο πρωτόκολλο SNMP**

### **1.3 Πρωτόκολλα διαχείρισης δικτύου (γενικά) και περιοχές διαχείρισης (management domains)**

Τα πρωτόκολλα διαχείρισης δικτύου (NMP, Network Management Protocols) είναι τα πρωτόκολλα που χρησιμοποιούνται για την επικοινωνία μεταξύ διαχειριστή και αντιπροσώπων (ή πληρεξούσιων αντιπροσώπων) με σκοπό την ανταλλαγή πληροφοριών που έχουν να κάνουν με την διαχείριση των συσκευών του δικτύου. Τα πιο γνωστά πρωτόκολλα αυτής της κατηγορίας είναι το SNMP για δίκτυα TCP/IP και το CMIP για δίκτυα που βασίζονται στο πρότυπο OSI. Τα πρωτόκολλα διαχείρισης δικτύου καθορίζουν:

- Τον ακριβή τρόπο επικοινωνίας διαχειριστή – αντιπροσώπου.
- Τη μορφή και την σημασία των μηνυμάτων που ανταλλάσσονται.

Σε ένα σύνθετο σύστημα διαχείρισης μπορεί να υπάρχουν πολλοί διαχειριστές και αντιπρόσωποι. Επειδή η δομή του συστήματος είναι ιεραρχική, ένας διαχειριστής μπορεί να εκτελεί την διαδικασία διαχείρισης για τους δικούς του αντιπροσώπους αλλά ταυτόχρονα μπορεί να αποτελεί και ο ίδιος αντιπρόσωπο για κάποιον διαχειριστή που βρίσκεται πιο πάνω στην ιεραρχία (και από τον οποίο δέχεται εντολές). Με τον τρόπο αυτό διαχειριστές και αντιπρόσωποι ορίζουν μια μορφή ιεραρχίας από πεδία (περιοχές) διαχείρισης. Τα πεδία αυτά ονομάζονται, σύμφωνα με την ορολογία του μοντέλου αναφοράς OSI, **περιοχές διαχείρισης (management domains)**. [11]

### **1.4 Αρχιτεκτονικές διαχείρισης**

Για να λειτουργήσει μια πλατφόρμα διαχείρισης η οποία θα αναλυθεί πιο κάτω χρειάζεται τις αρχιτεκτονικές διαχείρισης οι οποίες κατηγοριοποιούνται σε : Κεντρική, Ιεραρχική και Κατανεμημένη.

- Κεντρική Αρχιτεκτονική διαχείρισης (centralized)

Πρόκειται για την πιο απλή και κλασική αρχιτεκτονική διαχείρισης. Η πλατφόρμα διαχείρισης βρίσκεται εγκατεστημένη σε ένα κεντρικό σταθμό εργασίας ο οποίος και αναλαμβάνει όλα τα καθήκοντα διαχείρισης του δικτύου. Η αρχιτεκτονική αυτή είναι συμβατή με το μοντέλο διαχειριστή – αντιπροσώπου. Η πλατφόρμα διαχείρισης επιτελεί τις παρακάτω λειτουργίες:

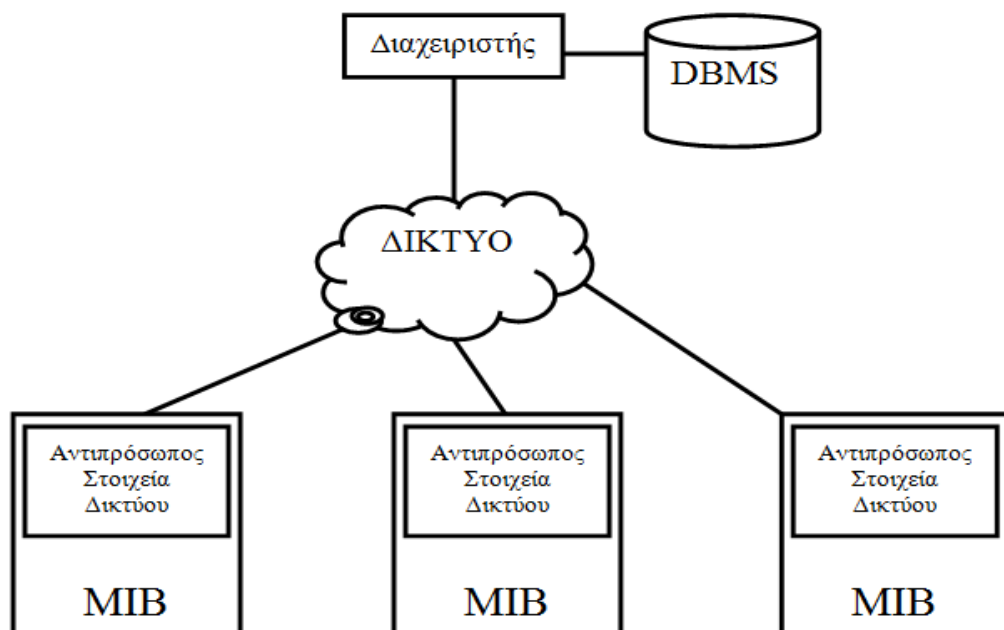
- ο Αναλαμβάνει την επικοινωνία με όλα τα διαχειριζόμενα στοιχεία μέσω των αντιπροσώπων και του πρωτοκόλλου διαχείρισης.



- Διαχειρίζεται την αποθήκευση των πληροφοριών διαχείρισης του δικτύου. Η αποθήκευση μπορεί να γίνεται κεντρικά σε ένα σημείο, ή για λόγους ασφαλείας να είναι κατανεμημένη σε πολλά μηχανήματα, αλλά ο έλεγχος όπως και όλος ο σχεδιασμός της αρχιτεκτονικής είναι κεντρικός (τα δεδομένα μπορούν να διαβάζονται και να γράφονται μόνο από το κεντρικό μηχάνημα).
- Παρέχει μια ενιαία εικόνα του διαχειριζόμενου δικτύου στον υπεύθυνο διαχειριστή μέσω κατάλληλου περιβάλλοντος επικοινωνίας με το χρήστη

Γενικά, για την κεντρική αρχιτεκτονική ισχύει: Είναι σύμφωνη με το γνωστό μοντέλο agent-manager. Περιέχει ένα κεντρικό διαχειριστή ο οποίος:

- Επικοινωνεί με όλα τα διαχειριζόμενα στοιχεία του δικτύου.
- Διαχειρίζεται την αποθήκευση των πληροφοριών του συστήματος
- Παρέχει μία ενοποιημένη εικόνα του διαχειριζόμενου δικτύου στο διαχειριστή μέσω κατάλληλου περιβάλλοντος επικοινωνίας με το χρήστη.



**Εικόνα 2: Κεντρική Αρχιτεκτονική Διαχείρισης**

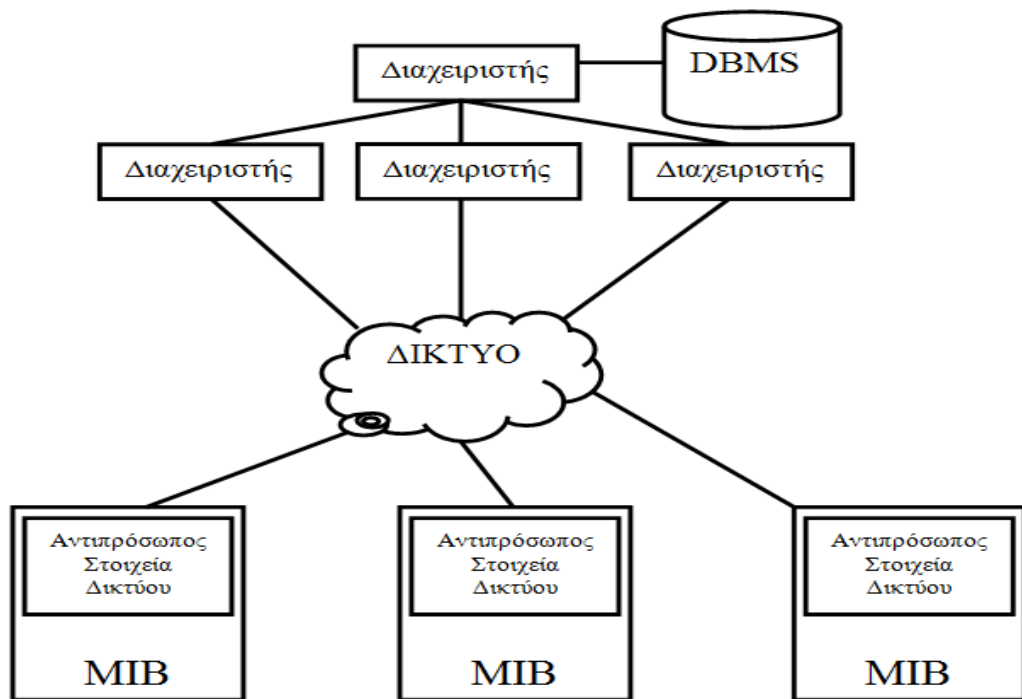
- Ιεραρχική Αρχιτεκτονική διαχείρισης (Hierarchical)

Χρησιμοποιούνται πολλές πλατφόρμες διαχείρισης. Η μία από αυτές λειτουργεί σαν κεντρικός σταθμός εξυπηρέτησης του δικτύου, ενώ οι άλλες σαν πελάτες. Αυτές δεν έχουν χωριστό DBMS (σύστημα διαχείρισης βάσης δεδομένων), αλλά χρησιμοποιούν το DBMS του κεντρικού σταθμού ενώ ο συντονισμός των λειτουργιών τους γίνεται

από το διαχειριστή που βρίσκεται υψηλότερα στην ιεραρχία. Ορισμένες από τις λειτουργίες στο ιεραρχικό σύστημα διαχείρισης εκτελούνται από τον κεντρικό διαχειριστή, ενώ άλλες ανατίθενται στους επιμέρους διαχειριστές.

Γενικά ο κεντρικός διαχειριστής συγκεντρώνει μόνο τις σημαντικές πληροφορίες αφήνοντας τις επιμέρους λεπτομέρειες στους διαχειριστές που βρίσκονται στο χαμηλότερο επίπεδο. Η επικοινωνία μεταξύ του κεντρικού διαχειριστή και των επιμέρους διαχειριστών μπορεί να γίνεται μέσω του ίδιου του δικτύου που διαχειρίζονται είτε και μέσω ανεξάρτητου δικτύου διαχείρισης. Στην δεύτερη αυτή περίπτωση, έχουμε καλύτερη αξιοπιστία αφού ακόμα και σε περίπτωση σοβαρής βλάβης του κανονικού δικτύου, θα έχουμε επικοινωνία των διαχειριστών μεταξύ τους. [11]

Η ιεραρχική αρχιτεκτονική παρέχει καλύτερο έλεγχο και υψηλότερη απόδοση στη διαχείριση του δικτύου. Ένα από τα βασικά της πλεονεκτήματα είναι ότι απομονώνει από το επίπεδο του κεντρικού διαχειριστή τα πρωτόκολλα και τους διαχειριστές που βρίσκονται στα χαμηλότερα επίπεδα. Αυτό είναι ιδιαίτερα σημαντικό σε ετερογενή δίκτυα (όπου χρησιμοποιούνται λειτουργικά συστήματα και πρωτόκολλα διαφορετικά μεταξύ τους αλλά διασυνδεδεμένα) γιατί έτσι εμφανίζεται στο χρήστη το περιβάλλον διαχείρισης σαν ομοιογενές ακόμα και αν στην πραγματικότητα δεν είναι.



**Εικόνα 3: Ιεραρχική Αρχιτεκτονική Διαχείρισης**

- Κατανεμημένη Αρχιτεκτονική διαχείρισης (distributed)

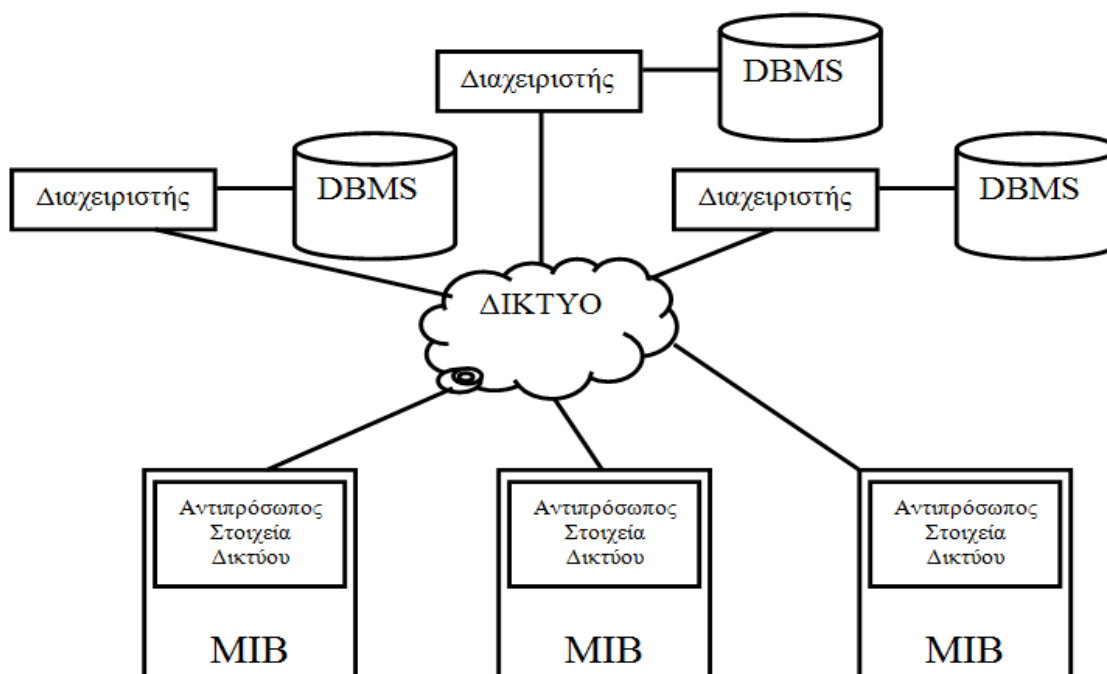
Αποτελεί ουσιαστικά συνδυασμό της κεντρικής και της ιεραρχικής. Χρησιμοποιεί πολλές ομότιμες πλατφόρμες διαχείρισης, κάθε μία από τις οποίες αποτελεί ένα κεντρικό σύστημα. Μια πλατφόρμα είναι επικεφαλής μιας σειράς από ομότιμα συστήματα διαχείρισης (όπως στην ιεραρχική αρχιτεκτονική). Ωστόσο κάθε πλατφόρμα μπορεί να έχει το δικό της σύστημα διαχείρισης βάσης δεδομένων (DBMS) με στοιχεία που να αφορούν οποιοδήποτε σημείο του δικτύου.

Επειδή η διαχείριση κατανέμεται σε τοπικούς διαχειριστές έχει γενικά μικρότερες απαιτήσεις σε υλικό και υπολογιστική ισχύ. Καθ' ένας από τους τοπικούς διαχειριστές διαχειρίζεται μόνο τον τομέα της αρμοδιότητας του και δεν έχει το βάρος της παρακολούθησης ολόκληρου του δικτύου. Αν όμως χρειαστεί πληροφορίες για περιοχή του δικτύου που δεν του ανήκει μπορεί να τις ζητήσει από τον αντίστοιχο ομότιμο διαχειριστή.

Τα πλεονεκτήματα της:

- Συνδυάζει την κεντρική με την ιεραρχική

- Έχει μία κεντρική πλατφόρμα διαχείρισης ή μια ιεραρχία από πλατφόρμες εξυπηρετητή πελάτη
- Χρησιμοποιεί ομότιμες πλατφόρμες διαχείρισης που καθεμιά τους χωριστά αποτελεί ένα κεντρικό σύστημα. [11]



**Εικόνα 4: Κατανεμημένη Αρχιτεκτονική Διαχείρισης**

## 1.5 Το πρωτόκολλο 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.

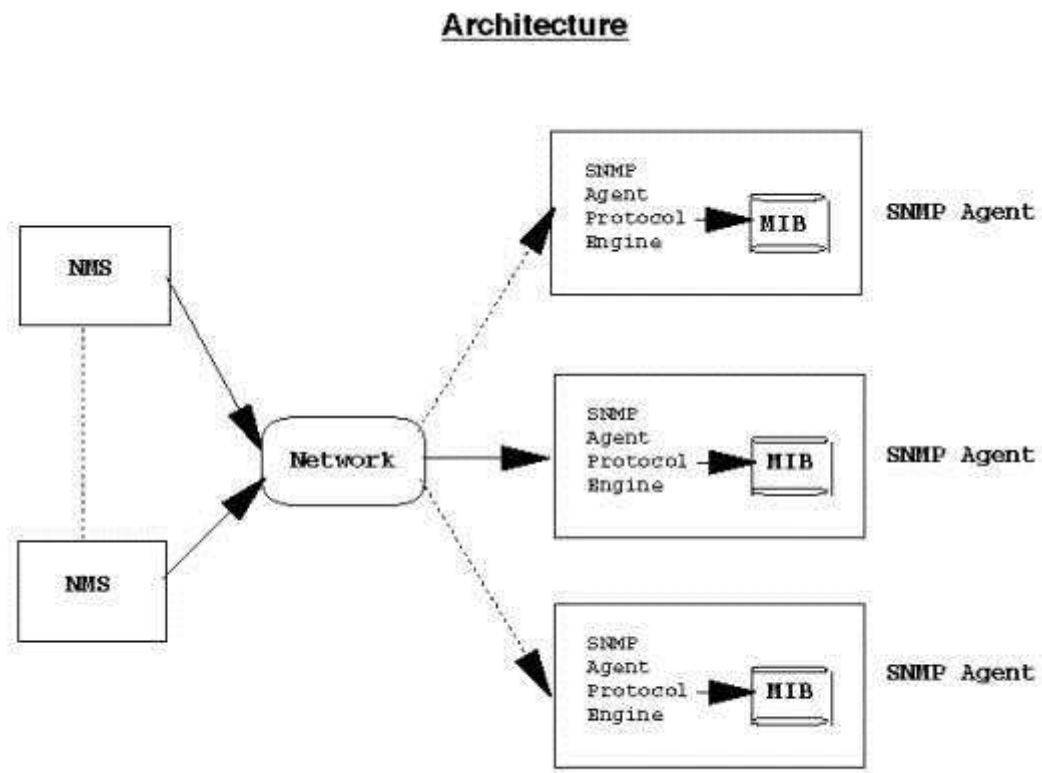
Ένα δίκτυο το οποίο διαχειρίζεται με SNMP έχει τρία βασικά στοιχεία: διαχειριζόμενες συσκευές, πράκτορες (agents) και συστήματα διαχείρισης δικτύου (Network Management Systems- NMS).

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.



**Εικόνα 5: Αρχιτεκτονική SNMP πρωτοκόλλου**

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.

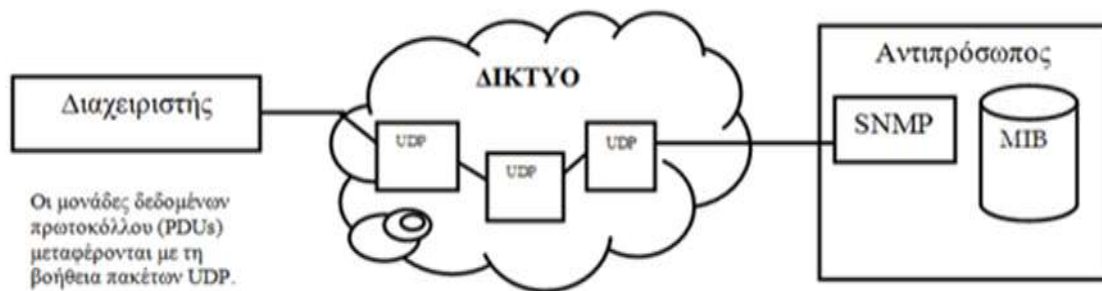
SNMPv2: Αποτελεί μια εξέλιξη του SNMPv1. Λειτουργεί επίσης με τις προδιαγραφές του SMI και προσφέρει κάποιες βελτιώσεις σε σχέση με το SNMP όπως να επεκτείνει το πρότυπο ώστε να μπορεί να χρησιμοποιηθεί (εκτός από το TCP/IP) και σε πρωτόκολλα που ακολουθούν το πρότυπο OSI.

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.

Παρακάτω περιγράφεται η διαδικασία επικοινωνίας μεταξύ διαχειριστών και αντιπροσώπων γίνεται με την ανταλλαγή μονάδων δεδομένων πρωτοκόλλου (PDU, Protocol Data Units). Κάθε μία από τις μονάδες αυτές κωδικοποιείται μέσα σε ένα και μοναδικό πακέτο UDP (User Datagram, αυτοδύναμο πακέτο) και ανταλλάσσεται με την χρήση του πρωτοκόλλου UDP. Το SNMP είναι δομημένο με τέτοιο τρόπο ώστε να εφαρμόζεται ακριβώς πάνω από το UDP. [2]



**Εικόνα 6: Διαδικασία ανταλλαγής δεδομένων στο πρωτόκολλο SNMP**

## **2. Λειτουργικές απαιτήσεις – Επιλογή Λογισμικού**

### **2.1 Λειτουργικές απαιτήσεις του Κέντρου Πληροφορικής και Δικτύων**

Το Κέντρο Πληροφορικής και Δικτύων (ΚΠΔ) του Χαροκοπείου Πανεπιστημίου έχει αναλάβει την ανάπτυξη, την απρόσκοπτη λειτουργία και τη διαχείριση του δικτύου δεδομένων και φωνής του Πανεπιστημίου, με σκοπό την διασφάλιση υψηλής ποιότητας υπηρεσιών για το σύνολο των μελών του Πανεπιστημίου. Μία από αυτές τις δραστηριότητες είναι η παρακολούθηση της λειτουργίας του δικτύου και η διαχείριση των πόρων του. Ακολούθως αναφέρονται κάποια από τα χαρακτηριστικά που απαιτήθηκε να έχει ένα τέτοιο λογισμικό:

- Κύρια απαίτηση είναι η παρακολούθηση της κατάστασης των εξυπηρετητών του πανεπιστημίου καθώς και των δικτυακών συσκευών.
- Επιπλέον επιθυμητή είναι η παρακολούθηση σε επίπεδο παρεχόμενων υπηρεσιών όπως το SSH. Το τελευταίο είναι ένα δικτυακό πρωτόκολλο που ορίζει έναν τρόπο για ασφαλείς, κρυπτογραφημένες συνδέσεις μεταξύ δύο μηχανημάτων. Οι κρυπτογραφημένες αυτές συνδέσεις είναι δυνατόν να επιτυγχάνονται και μέσω ενός επισφαλούς, εν γένει, δικτύου, όπως είναι το Internet. Το SSH συνήθως χρησιμοποιείται για την απομακρυσμένη, πλήρη πρόσβαση στη γραμμή εντολών διαφόρων Unix boxes -- αλλά όχι μόνο. Λόγω της πολύ σημαντικής χρησιμότητας του είναι απαραίτητη η παρακολούθηση του σχετικά με τους servers του πανεπιστημίου.
- Επίσης σε έναν εξυπηρετητή διαδικτύου (Web server) είναι πολύ σημαντικό να είμαστε ενήμεροι σχετικά με: τη χωρητικότητα και την ταχύτητα των δίσκων, την μνήμη RAM, επεξεργαστική ισχύ (C.P.U).
- Θα πρέπει να υπάρχει η δυνατότητα σύνδεσης με το AD του πανεπιστημίου ώστε να εξουσιοδοτηθεί ένας κάτοχος λογαριασμού υπηρεσιών του Πανεπιστημίου, ο οποίος θα αναλάβει να παρακολουθεί την υπηρεσία.
- Καταγραφή των συμβάντων με επίπεδα κρισιμότητας
- Δυνατότητα εξαγωγής στατιστικών συμπερασμάτων, για παράδειγμα αναφορές που δείχνουν τον χρόνο καθυστέρησης ή επίλυση κάποιου προβλήματος θα μπορούσε να βοηθούσε το ΚΠΔ να βελτιώσει την απόδοση τους εξάγοντας τα στατιστικά τους και ελέγχοντας με το πέρασμα του χρόνου.



- Επιθυμητή είναι η επίσης η δυνατότητα αποστολής ειδοποιήσεων μέσω email στους διαχειριστές του συστήματος για όλα τα ζητήματα που δημιουργήθηκαν, που επιλύθηκαν ή όχι. Η τελευταία θα βοηθούσε το ΚΠΔ να ελέγχει την κατάσταση του δικτυακού του εξοπλισμού για την υπάρχουσα κατάσταση του.

## **2.2 Κατάλληλο λογισμικό**

Οι επιλογές μας ήταν η ανάπτυξη μιας εφαρμογής εξ' αρχής η ή χρήση ενός υπάρχοντος λογισμικού επίλυσης τεχνικών προβλημάτων. Η εξ' αρχής ανάπτυξη θα απαιτούσε περισσότερο χρόνο και η λειτουργία του ίσως να μην ήταν τόσο αποδοτική οπότε αποσκοπήσαμε στην ανάπτυξη κάποιου λογισμικού ήδη υπάρχοντος με παραμετροποιήσεις.

Όσον αφορά τις επιλογές στις υπάρχουσες εφαρμογές, υπάρχει δυνατότητα να χρησιμοποιηθεί λογισμικό ανοιχτού κώδικα που δεν απαιτεί καταβολή χρημάτων όπως ένα εμπορικό προϊόν. Σε αποτέλεσμα μετά από έρευνα διαπιστώθηκε πως οι διαφορές μεταξύ των δημοφιλέστερων εφαρμογών ανοιχτού κώδικα στα συστήματα υποστήριξης και των εμπορικών προϊόντων είναι από μικρές έως μηδαμινές. Επίσης, όσον αφορά τις εφαρμογές ανοιχτού κώδικα υπάρχει μεγάλη ποικιλία επιλογών, που συνεπάγεται την ικανοποίηση των περισσοτέρων, αν όχι όλων, των απαιτήσεων του ΚΠΔ μετά από σωστή επιλογή λογισμικού.

## **2.3 Δημοφιλέστερα λογισμικά ανοιχτού κώδικα**

Το λειτουργικό σύστημα Linux και η κοινότητα ανοιχτού κώδικα παρέχει αμέτρητες εφαρμογές. Επίσης παρέχει τρόπους υποστήριξης ακόμη και για μη τεχνικά τμήματα. Υπάρχουν πολλά λογισμικά ανοιχτού κώδικα που εκτελούν την διαχείριση των αιτημάτων των πελατών/χρηστών με αποτελεσματικότητα. Χωρίς καμία χρέωση παρέχουν τον πηγαίο κώδικα έτσι ώστε κάποιος να μπορεί να τα διαμορφώσει βάσει των απαιτήσεων του. Γενικότερα επικρατεί η αντίληψη πως τα λογισμικά ανοιχτού κώδικα δεν είναι έμπιστα και μπορούν να θέσουν σε κίνδυνο το άτομο ή την επιχείρηση που τα επιλέγει. Όμως κάτι τέτοιο δεν ισχύει στην πραγματικότητα, καθώς είναι το ίδιο καλά με τα αντίστοιχα εμπορικά πακέτα, με πολύ ανταγωνιστικά χαρακτηριστικά και πολλά από αυτά τα λογισμικά χρησιμοποιούνται από μεγάλες επιχειρήσεις σε όλο τον κόσμο.

Ακολουθεί η σύντομη περιγραφή και η σύγκριση κάποιων βασικών χαρακτηριστικών των δημοφιλέστερων από αυτά.

**ICINGA WEB 2 (<https://www.icinga.org>)**

Το Icinga 2 ξεκίνησε ως εφαρμογή παρακολούθησης του συστήματος Nagios. Είναι η επέκταση του icinga η οποία ξαναγράφηκε για να δώσει στους χρήστες της ένα σύγχρονο περιβάλλον εργασίας και υποστήριξης με πολλαπλές βάσεις δεδομένων. Με out-of-the-box εξισορρόπηση φορτίου, τις ειδοποιήσεις, και τη διαμόρφωση, το Icinga 2 μειώνει το χρόνο από την εγκατάσταση για σύνθετα περιβάλλοντα. Το Icinga 2 υποστηρίζει γραφήματα, παρέχοντας στους διαχειριστές γραφικά απόδοσης σε πραγματικό χρόνο, χωρίς καμία δυσκολία. Οι διαχειριστές μπορούν να δουν τα φίλτρα, και να ιεραρχήσουν τα προβλήματα, για την παρακολούθηση των οποίων έχουν ήδη ληφθεί μέτρα. Μια νέα άποψη διεπαφής , επιτρέπει στους διαχειριστές να βλέπουν hosts και services σε μία σελίδα. Μπορείτε να δείτε τα γεγονότα κατά τη διάρκεια μιας συγκεκριμένης χρονικής περιόδου ή να φιλτράρετε τα συμβάντα για να καταλάβετε ποιες χρειάζονται άμεση προσοχή.

Το πρόγραμμα του ICINGA 2 ξεκίνησε τον Φεβρουάριο του 2015 ως επέκταση του icinga το οποίο ήταν η επέκταση του Nagios λόγω της στασιμότητας στην ανάπτυξης του. 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 ICINGA2 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 είναι ελεύθερο λογισμικό όπως επίσης είναι και ίσως το μόνο που είναι διαθέσιμο σε 21 γλώσσες εκτός των Αγγλικών.

Τέλος, αλλά εξίσου σημαντικό είναι το γεγονός ότι παρέχει στο διαδίκτυο πληθώρα πληροφοριών σχετικά με την εγκατάσταση και την παραμετροποίηση του.

**OpenNMS** (<http://www.opennms.org/>)

Μια εξαιρετικά ευέλικτη λύση διαχείρισης δικτύου, το OpenNMS μπορεί να χειριστεί οποιαδήποτε εργασία διαχείρισης δικτύου, είτε πρόκειται για την διαχείριση συσκευής, την

παρακολούθηση της απόδοσης εφαρμογών, τον έλεγχο απογραφής ή διαχείριση συμβάντων. Με την υποστήριξη του IPv6, ένα ισχυρό σύστημα ειδοποιήσεων, που έχει την ικανότητα να καταγράφει τα σενάρια χρήστη να δοκιμάζει τις εφαρμογές Web, το OpenNMS έχει ότι χρειάζονται οι διαχειριστές και δοκιμαστές του δικτύου. Το OpenNMS έχει αναπτυχθεί, σε κινητές συσκευές, που ονομάζεται OpenNMS , που επιτρέπει στους επαγγελματίες των δικτύων να ελέγχουν το δίκτυό τους, ακόμη και όταν είναι εκτός.

Η iOS έκδοση της εφαρμογής, που είναι διαθέσιμη στα iTunes εμφανίζει διακοπές λειτουργίας, κόμβους και συναγερμούς. Η επόμενη έκδοση θα προσφέρει επιπλέον λεπτομέρειες, γραφικές παραστάσεις και πληροφορίες σχετικά με την IP και τις διεπαφές του SNMP. Η Android εκδοχή υπάρχει διαθέσιμη στο Google Play, εμφανίζει την διαθεσιμότητα του δικτύου, τις διακοπές λειτουργίας και συναγερμούς στο ταμπλό της εφαρμογής OpenNMS. Ισχύει για πελάτες κινητών συσκευών που είναι συμβατές με OpenNMS Horizon 1.12 ή μεγαλύτερη και με OpenNMS Meridian 2015.1.0 ή μεγαλύτερη. Η γλώσσα που έχει γραφεί είναι η java και υποστηρίζεται σε Windows, Linux, Mac OS X και Solaris. Εκδόθηκε σύμφωνα με τους όρους της GNU General Public License έκδοση 2 και είναι ελεύθερο λογισμικό

#### **Nagios (<https://www.nagios.org/>)**

Το nagios είναι από τα παλαιότερα συστήματα παρακολούθησης δικτύου. Έχει γραφεί σε γλώσσα C και είναι γρήγορο, αξιόπιστο, και εξαιρετικά προσαρμόσιμο. Το Nagios μπορεί να είναι μια πρόκληση για τους νεοεισερχόμενους, αλλά η περίπλοκη διαμόρφωση του είναι μια δύναμη, καθώς μπορεί να προσαρμοστεί ακριβώς για οποιοδήποτε έργο παρακολούθησης. Πρώτη φορά εμφανίστηκε το 1999 και υποστηρίζει τις εξής βάσεις δεδομένων MySQL, Postgres, DB2, Oracle, Microsoft SQL Server (MSSQL). Οι γλώσσες που χρησιμοποιεί είναι C ++, Perl, Ruby, Python, PHP, C #, κλπ) αλλά είναι γραμμένο σε C. Πρώτη κυκλοφορία έγινε τον Μάρτιο του 1999 και εκδόθηκε σύμφωνα με τους όρους της GNU General Public License έκδοση 2 και είναι ελεύθερο λογισμικό.

Έχει την δυνατότητα να εγκατασταθεί σε windows, Linux, Mac OS X, και άλλα συστήματα που είναι Unix- like.

#### **Zenoss Core (<http://www.zenoss.org/>)**

Ένα άλλο πρόγραμμα ανοιχτού κώδικα είναι το Zenoss core το οποίο δίνει στους διαχειριστές δικτύου μια ολοκληρωμένη, one-stop λύση για την παρακολούθηση και διαχείριση όλων των εφαρμογών, servers, αποθήκευσης, δικτύωσης εξαρτημάτων,

εργαλεία εικονικά, καθώς και άλλα στοιχεία της υποδομής των επιχειρήσεων. Οι διαχειριστές μπορούν να βεβαιωθούν ότι το υλικό λειτουργεί αποτελεσματικά και να επωφεληθούν από τον σχεδιασμό του για να συνδέουν ZenPacks για εκτεταμένη λειτουργικότητα.

Ο Zenoss Core 5, που κυκλοφόρησε τον φετινό Φεβρουάριο, παίρνει το ήδη ισχυρό εργαλείο και το βελτιώνει περαιτέρω, με μια ενισχυμένη διεπαφή χρήστη και ένα εκτεταμένο ταμπλό. Η διαδικτυακή κονσόλα και τα ταμπλό της εφαρμογής ήταν ήδη εξαιρετικά ευέλικτα στις προσαρμογές και δυναμικά, και η νέα έκδοχή τώρα επιτρέπει τους διαχειριστές να εμπεριέχουν πολλαπλά διαγράμματα σε ένα και μοναδικό γράφημα. Σκεφτείτε το σαν ένα εργαλείο για καλύτερο εντοπισμό της πρωταρχικής αιτίας και ανάλυση αιτίας/αποτελέσματος. Η βάση δεδομένων που υποστηρίζει είναι MySQL και μπορεί να χρησιμοποιηθεί σε Linux, Mac OS X και Solaris.

Νέα διαγράμματα HTML5 μπορούν να εξαχθούν έξω από το εργαλείο. Η Zenoss Κέντρο Ελέγχου επιτρέπει την out-of-band διαχείριση και την παρακολούθηση όλων των εξαρτημάτων Zenoss.

Είναι γραμμένο σε Python 90%, Java 10% και εκδόθηκε σύμφωνα με τους όρους της GNU General Public License έκδοση 2 και είναι ελεύθερο λογισμικό.

#### **NeDi (<http://www.nedi.ch/>)**

Το NeDi δεν είναι τόσο γνωστό όσο τα υπόλοιπα αλλά είναι μια πολύ καλή λύση για τον εντοπισμό συσκευών μέσω διαδικτύου. Παρέχει την τρέχουσα θέση οποιασδήποτε συσκευής, καθώς και ένα ιστορικό. Μπορεί να χρησιμοποιηθεί για να εντοπίσει κλεμμένες συσκευές ειδοποιώντας σας εάν εμφανιστεί ξανά στο δίκτυο. Λειτουργεί σε Linux και σε Ubuntu, CentOS, Suse, Mac OSX, FreeBSD και η βάση δεδομένων που χρησιμοποιεί είναι η MySQL.

#### **Observium (<http://www.observium.org>)**

Η αρχική του έκδοση βγήκε τον Οκτώβρη 2006 και υποστηρίζει λειτουργικό σύστημα unix-like. Βασίζεται στην PHP γλώσσα προγραμματισμού και MySql βάση δεδομένων. Το Observium είναι μια αυτό-ανακάλυψη του δικτύου πλατφόρμας παρακολούθησης που υποστηρίζει ένα ευρύ φάσμα από πλατφόρμες υλικού και λειτουργικών συστημάτων, συμπεριλαμβανομένων της Cisco, unix-like, Linux, HP, Juniper, Dell, FreeBSD, Brocade ,

Netscaler, NetApp και πολλά άλλα. Επιδιώκει να παρέχει ένα ισχυρό αλλά απλό και διαισθητικό περιβάλλον για την υγεία και την κατάσταση των δικτύων.

### **Zabbix (<http://www.zabbix.com>)**

Ίσως από τα παλαιότερα λογισμικά ανοιχτού κώδικα κυκλοφόρησε τον Μάρτιο του 2004 γραμμένο σε C και PHP. Χρησιμοποιεί για βάσεις δεδομένων MySQL, PostgreSQL, SQLite, Oracle ή IBM DB2. Είναι μια επιχείρηση παρακολούθησης λύση ανοικτού κώδικα για τα δίκτυα και τις εφαρμογές και έχει σχεδιαστεί για να παρακολουθεί και να παρακολουθείτε την κατάσταση των διαφόρων υπηρεσιών δικτύου, διακομιστές, και άλλων υλικού δικτύου. Εκδόθηκε σύμφωνα με τους όρους της GNU General Public License έκδοση 2 και είναι ελεύθερο λογισμικό.

### **Πίνακας σύγκρισης λειτουργικών συστημάτων**

OPEN SOURCE	Server	Λειτουργικά συστήματα	Βάσεις δεδομένων	Γλώσσα που γράφτηκε	Αρχική έκδοση
ICINGA WEB2	Apache web server	Windows, Linux	MySQL, PostgreSQL, Oracle, IDO DB	mixed	2015
OPEN NMS		Windows, Linux, Mac OS X, Solaris	PostgreSQL	Java	1999
NAGIOS		Cross-platform	MySQL, Postgres,DB2, Oracle, Microsoft SQL Server (MSSQL)	C	1999
ZENOSS CORE		Linux, Mac OS X,	MySQL	Python90%, Java10%	2015

NeDi		Solaris			
		Linux, Mac OS X, Windows, Solaris	MySQL		
OBSERVUM	Apache web server	Unix-like	MySQL	Php	2006
ZABBIX		Cross-platform	MySQL, PostgreSQL, SQLite, Oracle, IBM DB2	C, Php	2004

## 2.4 Κατάλληλο λογισμικό

Παρατηρούμε λοιπόν ότι υπάρχει μεγάλη ποικιλία επιλογών και σ' αυτή την ενότητα παρουσιάστηκαν μόνο τα πιο δημοφιλή. Μέτα την σύγκριση των πιο δημοφιλέστερων συστημάτων επιλέξαμε να χρησιμοποιήσουμε το icinga web 2 διότι με την ισχύουσα κατάσταση στο Κέντρο Πληροφορικής και Δικτύων αυτή η εφαρμογή συμπληρώνει το σύνολο των λειτουργικών απαιτήσεων όπως περιγράφηκαν στην ενότητα 2.1. Επίσης αυτό που ξεχώρισε το icinga web 2 από τα υπόλοιπα λειτουργικά συστήματα ανοιχτού κώδικα είναι το γεγονός ότι ήταν το τελευταίο λειτουργικό που κυκλοφόρησε στην αγορά εργασίας ( Φεβρουάριος 2015) καθώς είχε και τις περισσότερες δυνατότητες από τα υπόλοιπα. Η νέα διεπαφή με ένα εντελώς νέο σχεδιασμό και πολλές φιλικές προς το χρήστη βελτιώσεις συγκριτικά με τα υπόλοιπα μας ικανοποίησε περισσότερο αντί των υπολοίπων.

## 2.5 Σύντομη ιστορική αναδρομή για το icinga web 2

Το icinga web 2 κυκλοφόρησε πρώτη φορά τον Φεβρουάριο του 2015. Είναι η επέκταση του icinga το οποίο είναι η επέκταση του ισχυρού Nagios. Σε αντίθεση με τις προηγούμενες εκδόσεις, η διεπαφή αυτή έχει μια λεπτομερή εικόνα για την υπηρεσία υποδοχής και για έλεγχοι που δεν εκτελούνται στο χρόνο. Προστέθηκαν πολλά νέα χαρακτηριστικά, όπως για παράδειγμα την πρόσβαση σε μονάδες απαιτεί πλέον κατάλληλα δικαιώματα όπως και για τους ρόλους πλέον περιέχουν δικαιώματα και περιορισμούς που μπορεί να εκχωρηθεί σε

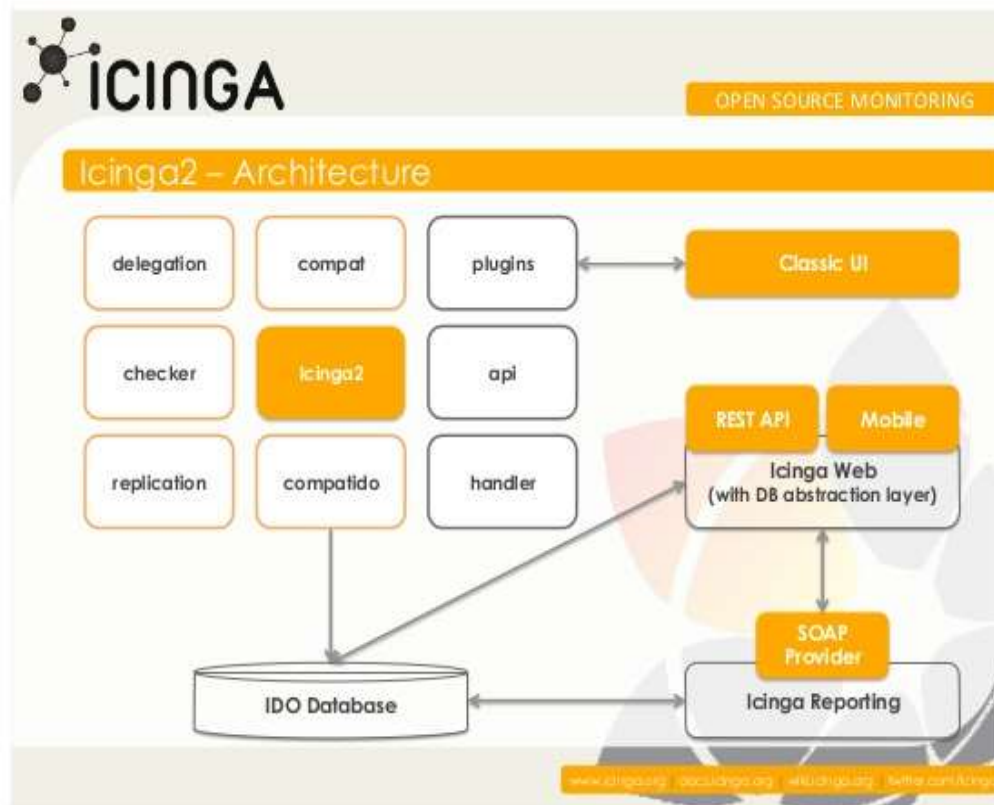
χρήστες και ομάδες χρηστών, προκειμένου να επιτρέπουν ή να περιορίζουν την πρόσβασή τους.

Έχει βελτιωμένη την υπηρεσία ενσωμάτωσης υπηρεσίες καταλόγου Active directory και σε άλλους διακοσμητές 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 directry. Επίσης σημαντική πρόοδος φαίνεται η δημιουργία των υπηρεσιών hosts και servers. Μια μεγάλη ευκαιρία είναι η δημιουργία host συστημάτων κ υπηρεσιών εξυπηρέτησης αντί να προσδιορίζονται μέσω των επιβεβαιωμένων αρχείων του icinga κάθε χρήστης τώρα μπορεί να τα δημιουργήσει μέσω της διεπαφής του δικτύου κ ακόμα να τα κοινοποιήσει σε άλλους.

Η διαμόρφωση επιτρέπει να προσθέσετε 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. Βασικός έλεγχος ταυτότητας για πρόσβαση είναι επίσης ένα νέο χαρακτηριστικό που διευκολύνει τη χρήση του API.

Για το μέλλον, μπορούμε να περιμένουμε περισσότερες δράσεις API και τη δημιουργία ενοτήτων για την ενσωμάτωση διαφόρων δημοφιλών εργαλείων για την ενίσχυση της στοίβας DevOps. [3]

## 2.6 Αρχιτεκτονική συστήματος icinga web 2



Εικόνα 7: Αρχιτεκτονική icinga 2

Το Icinga2 είναι η ανάπτυξη προς αντικατάσταση των Nagios και Icinga 1, και είναι συμβατό με όλα τα υπάρχοντα plugins, διεπαφές χρήστη (π.χ.. Classic UI, Icinga Web) και addons. Έχει σχεδιαστεί όμως, για να είναι πιο εύκολο στην εγκατάσταση, και είναι απίστευτα πιο γρήγορο και ισχυρό.

*Απλή εγκατάσταση:* Το icinga 2 κάνει την εύκολης εργασία της εγκατάστασης. 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 όπως φαίνεται και στην εικόνα 8.





**Εικόνα 8: Αρχιτεκτονική icinga2\_(ii)**

Ολοκληρώνεται η διαδικασία εγκατάστασης με τα συνηθισμένα addons χωρίς να χρειάζεται να καθαρίζουμε από το διαδίκτυο για τα εργαλεία και τα σενάρια. Απλά εισάγουμε “χαρακτηριστικό ενεργοποίησης icinga2” η “χαρακτηριστικό απενεργοποίησης icinga2” με την εντολή cli κατά τη ρύθμιση, για να ενεργοποιήσουμε τις λειτουργίες που χρειάζονται.

*Εύκολη και αποτελεσματική add-on ενσωμάτωση:* Όπως το Icinga 2 έρχεται επίσης με πολλαπλά backends, ακριβώς για οποιοδήποτε addon μπορεί να ενσωματωθεί εύκολα. Ειδικότερα, με τη μονάδα Perfddata ενεργοποιημένη, το Icinga 2 απλοποιεί την ενοποίηση του με δημοφιλή εργαλεία γραφημάτων απόδοσης, όπως PNP4Nagios, Graph και Graphite. Στην πραγματικότητα, το Icinga 2 υποστηρίζει Graphite. Έτσι, μπορούμε να απολαύσουμε σε πραγματικό χρόνο γραφικές παράστασης παρακολούθησης, όπως το Icinga2 μεταδίδει δεδομένα απόδοσης μέσω μιας άμεσης σύνδεσης με την υποδοχή του σε TCP. Αυτό σημαίνει μειωμένο φορτίο του συστήματος, όπως τα δεδομένα που δεν είναι γραμμένα και διαγράφονται στο σκληρό δίσκο πάνω από ένα script όπως συμβαίνει στην Icinga 1 και Nagios.

*Απόδοση προσανατολισμένη:* Το Icinga 2 είναι φτιαγμένο για να είναι γρήγορο. Χάρη στον πολυνηματικό σχεδιασμό του, μπορεί να τρέξει χιλιάδες ελέγχους κάθε δευτερόλεπτο χωρίς καμία ένδειξη ενόχλησης στην CPU του. Αυτό συμβαίνει ακριβώς για να είμαστε σίγουροι, ότι έχουμε ενσωματωμένα συμπλέγματα ελέγχου που παράγουν δεδομένα απόδοσης για το Icinga2 μας σε διάφορα περιστατικά η για μία ομάδα χρηστών. [13]

### 3. Εγκατάσταση

#### 3.1 Περιβάλλον αρχικής εγκατάστασης

Καθώς το τελικό κριτήριο για την επιτυχή έκβαση της διπλωματικής αυτής εργασίας είναι η επιτυχημένη εγκατάσταση, παραμετροποίηση και η πιλοτική λειτουργία μιας υπηρεσίας (NMS) για τους διαχειριστές του Πανεπιστημίου θα έπρεπε να εργασθούμε σε ένα ελεγχόμενο περιβάλλον προκειμένου να αποφύγουμε τυχόν προβλήματα.

Για τον λόγο αυτό η εγκατάσταση του ICINGA WEB2 και η παραμετροποίηση του δεν έγινε απευθείας στον server του Πανεπιστημίου. Προτιμήσαμε να χρησιμοποιήσουμε ένα λογισμικό 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 διατίθενται δωρεάν καθώς και οι διαφορές μεταξύ τους ελάχιστες. Κριτικές αναφέρουν τον VirtualBox ως πιο γρήγορο στην απόδοση του.

Γι' αυτό λοιπόν επιλέξαμε να χρησιμοποιήσουμε το VirtualBox το οποίο μπορείς κανείς να το εγκαταστήσει από την ιστοσελίδα <https://www.virtualbox.org/wiki/Downloads>. Μέσω αυτού του προγράμματος καταφέραμε ένα εικονικό μηχάνημα (virtual machine). Σ' αυτό το εικονικό μηχάνημα εγκαταστήσαμε λειτουργικό σύστημα Debian8 (64 bit) την έκδοση Jessie, η οποία είναι η τελευταία και θα υποστηρίζεται για τα επόμενα 5 χρόνια την οποία την κατεβάσαμε από την ιστοσελίδα <https://www.debian.org/releases/stable/debian-installer/>.

Την τελευταία έκδοση του ICINGA WEB2 μπορεί να την βρει κάποιος στην επίσημη ιστοσελίδα <https://www.icinga.org/download/>.

Στην ενότητα 3.2 που ακολουθεί πρόκειται να παρουσιαστούν αναλυτικά τα βήματα που ακολουθήθηκαν και οι εντολές που εκτελέστηκαν με σκοπό την εγκατάσταση της εφαρμογής και των προαπαιτούμενων της.

#### 3.2 Διαδικασία εγκατάστασης

Πρώτα από όλα ξεκινήσαμε την εγκατάσταση του icinga 2. Για να γίνει αυτό σωστά πρέπει να επιλέξουμε τα επίσημα “package repositories” τα οποία ταιριάζουν με το λειτουργικό σύστημα μας, τα οποία θα επιλέξουμε από <http://packages.icinga.org/debian/>.

Έπειτα, προσθέτουμε το κατάλληλο repository package για του λειτουργικό σύστημα μας το οποίο σε αυτό το σημείο να υπενθυμίσω είναι debian 8 Jessie. Πριν από αυτό όμως, έχουμε σιγουρευτεί ότι ανοίξαμε το τερματικό του root η αλλιώς κάναμε `sudo su-` στο απλό τερματικό του debian και γράφουμε τις εντολές:

Βήμα 1:

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

Στην συνέχεια εγκαθιστούμε το icinga 2 με το σωστό πακέτο που ταιριάζει με το λειτουργικό μας σύστημα (debian) μέσα από τις εντολές

Βήμα 2

- `apt-get install icinga2`
- `/etc/init.d/icinga2 start`

Το Icinga 2 μπορεί να χρησιμοποιηθεί με το Icinga Web 2 η ακόμα και έναν αριθμό άλλων διεπαφών ιστού. Σε αυτό το σημείο θα εξηγήσουμε πώς ρυθμίσαμε το Icinga Web 2. Η DB IDO (Βάση Δεδομένων δεδομένων εξόδου) διαμορφώνει την εξαγωγή όλων των ρυθμίσεων και την κατάσταση των πληροφοριών σε μια βάση δεδομένων. Η βάση δεδομένων IDO χρησιμοποιείται από έναν αριθμό έργων, συμπεριλαμβανομένων Icinga

Web 2, Icinga ή Icinga Web 1.x. Επί του παρόντος, υποστηρίζεται η MySQL και PostgreSQL. Παρατίθεται σε αυτό το σημείο τον τρόπο με τον οποίο υλοποιήθηκε αυτό:

Βήμα 3: Εγκατάσταση MySQL βάση δεδομένων για debian

- **apt-get install mysql-server mysql-client**

Στο παράθυρο που θα μας ανοίξει πληκτρολογούμε τον κωδικό μας και πατάμε OK.

Βήμα 4: Εγκατάσταση βάση δεδομένων IDO MySQL

- **apt-get install icinga2-ido-mysql**

Θα εμφανιστούν 4 νέα παράθυρα. Το πρώτο μας ρωτάει αν θέλουμε να ενεργοποιήσουμε τη λειτουργία Ido-mysql. Την οποία απορρίπτουμε επιλέγοντας *NO*, διότι θα ενεργοποιηθεί αμέσως μετά. Στο δεύτερο παράθυρο λοιπόν 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 όμως να έχουμε σημειώσει τις διαφορετικές ρυθμίσεις, θα μας χρησιμεύσει για τη διαμόρφωση του web interface.

Βήμα 5: Εγκατάσταση Ido-MySQL

- **cat/etc/icinga2/features-available/ido-mysql.conf**

**Και έπειτα συμπληρώνουμε**

```
user= "icinga"
password= "13011990"
host= "localhost"
database= "icinga1"
```

Μετά την ενεργοποίηση της Ido-MySQL-έχουμε τη δυνατότητα να κάνουμε επανεκκίνηση το Icinga 2 το οποίο κάνουμε :

- `service icinga2 restart`

Στο επόμενο βήμα 6 Πριν από την εγκατάσταση της διεπαφής ιστού, βεβαιωνόμαστε ότι έχουμε μια λειτουργία web server. Εάν δεν υπάρχει, εγκαθιστούμε με apache2.

- `apt-get install apache2`

Ενεργοποιούμε την εντολή External Command Pipe χρησιμοποιώντας CLI:

- `icinga2 feature enable command`

Τα πακέτα Debian χρησιμοποιούν nagios ως προεπιλεγμένο χρήστη και για όνομα ομάδας. Όμως αλλάζουν το icingacmd σε nagios. Οι χρήστες του webserver είναι διαφορετικά

διαθέσιμοι οπότε πρέπει να αλλάξουμε `www-data` to `wwwrun`, `www`, η `apache`. Αλλάζουμε "`www-data`" στον χρήστη που χρησιμοποιούμε για να εκτελεστούν οι απαιτήσεις αφετέρου.

Υπάρχει δυνατότητα επαλήθευσης σωστής καταχώρησης μέσα από την εντολή

- `$ id<your-webserver-user> (nagios)`

Τέλος στο βήμα 7 εγκαθιστούμε το `icinga web 2` και αποκτούμε και το κλειδί πρόσβασης στο περιβάλλον του.

- `apt-get install icingaweb2`
- `icingacli setup token create`

Σε περίπτωση που στον server δεν δουλέψει η εντολή εισάγουμε repositories wheezy backport ( στην παρούσα δεν δούλεψε γι αυτό και αναφέρετε).

### 3.2.1 Διαδικασία εγκατάστασης με screenshot

Πλέον μπορούμε να ανοίξουμε ένα browser πληκτρολογώντας αυτό που φαίνεται στην γραμμή αναζήτησης παρακάτω και να περιηγηθούμε. [4]

`localhost/icingaweb2/setup/login?renderLayout`



**Εικόνα 9: Είσοδος στο icinga web 2**

Στο παράρτημα Α θα μπορέσουμε να δούμε αναλυτικά τα βήματα εγκατάστασης του συστήματος αυτού με αναλυτικές εικόνες. Στο κεφάλαιο αυτό επισυνάπτουμε μόνο το πρώτο βήμα που είναι η είσοδος στο σύστημα και το τελευταίο που είναι η επιτυχής ολοκλήρωση εγκατάστασης όπως φαίνεται ακριβώς από κάτω.

Επιτυχής ενημέρωση εγκατάστασης του προγράμματος στον server του Πανεπιστημίου:

```
General configuration has been successfully written to: /etc/icingaweb2/config.ini
Authentication configuration has been successfully written to:
/etc/icingaweb2/authentication.ini
Account "vdalakas" has been successfully defined as initial administrator.
User Group Backend configuration has been successfully written to:
/etc/icingaweb2/groups.ini
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
Command transport configuration has been successfully created:
/etc/icingaweb2/modules/monitoring/commandtransports.ini
```

```
Monitoring security configuration has been successfully created:
/etc/icingaweb2/modules/monitoring/config.ini
Module "doc" has been successfully enabled.
Module "iframe" has been successfully enabled.
Module "monitoring" has been successfully enabled.
```

### 3.2.2 Έλεγχος διαδικασίας εγκατάστασης

Μέσα από αυτή την διαδικασία θεωρητικά, γιατί πρακτικά δεν υλοποιήθηκε, θα μπορούσαμε να έχουμε τον έλεγχο του συστήματος διαχείρισης δικτύου του πανεπιστημίου. Αναφέρομαι αφενός θεωρητικά διότι πρακτικά κάτι τέτοιο δεν ήταν εφικτό.

Το επόμενο βήμα λοιπόν έτσι ώστε να φθάσουμε στην επιτυχής διαχείριση του δικτύου ήταν να ακολουθήσουμε ακριβώς την ίδια εγκατάσταση όπως αναφέρεται στην ενότητα 3.2 αλλά πλέον στον server του πανεπιστημίου.

Στην παρακάτω εικόνα 23 φαίνεται η διαδικασία καταχώρησης στοιχείων έτσι ώστε το σύστημα διαχείρισης δικτύου μας να έχει περισσότερες πληροφορίες για να εντοπίζει τα στοιχεία του λογαριασμού μας.



**Authentication Backend**

Before you are able to authenticate using the LDAP connection defined earlier you need to provide some more information so that Icinga Web 2 is able to locate account details.

Backend Type \*

Backend Name \*

LDAP Connection \*

LDAP User Object Class

LDAP Filter

LDAP User Name Attribute

LDAP Base DN

**Εικόνα 10: Εντοπισμός στοιχείων λογαριασμού**

### **3.2.3 Παραμετροποίηση Διαμόρφωση Σύνδεση με LDAP(Lightweight Directory Access Protocol)**

Σε συνέχεια της προηγούμενης ενότητας, θα αναφερθούμε στην διαδικασία που παραμετροποιήσαμε τον LDAP. Πριν από αυτό όμως θα ήθελα να εξηγήσω την διαδικασία και την ιδιότητα του LDAP πρωτοκόλλου.

Το LDAP είναι ένα πρωτόκολλο ανοικτού προτύπου για την πρόσβαση σε υπηρεσίες καταλόγου το οποίο βασίζεται στο πρωτόκολλο X.500. Οι υπηρεσίες καταλόγου που βασίζονται στο LDAP πρωτόκολλο, υλοποιούνται χρησιμοποιώντας το μοντέλο client – Server. Η υλοποίηση μιας τέτοιας υπηρεσίας αποτελείται από έναν τουλάχιστον LDAP Server, και έναν τουλάχιστον LDAP Client. Ο όρος "υπηρεσίες καταλόγου" (Directory Services) αναφέρεται σε τεχνολογίες πρόσβασης πληροφοριών που βρίσκονται αποθηκευμένες σε κατάλληλα δομημένες βάσεις δεδομένων που ονομάζονται "κατάλογοι" (directories). Κατάλογος είναι ένα σύνολο στοιχείων τα οποία είναι ταξινομημένα με κάποια λογικό και ιεραρχικό τρόπο, όπως για παράδειγμα ένας τηλεφωνικός κατάλογος.

Στόχος της υπηρεσίας καταλόγου είναι να εξυπηρετεί σε αναζητήσεις στοιχείων χρηστών και γενικότερα δομημένων μονάδων ενός οργανισμού, κάποιου ιδρύματος ή μιας επιχείρησης και να λειτουργήσει ως βάση ανάπτυξης νέων υπηρεσιών προς τους χρήστες.

Πολύ σημαντική λειτουργία της υπηρεσίας καταλόγου είναι ο τρόπος αποθήκευσης των πληροφοριών, για να είναι άμεσα δυνατή η ανάκτηση αυτών είτε με αναζήτηση κάποιου μοναδικού χαρακτηριστικού (πχ userid χρήστη) είτε με αναζήτηση ανάμεσα σε άλλα σχετικά χαρακτηριστικά (πχ το πλήρες όνομα ενός χρήστη).

Το πρωτόκολλο LDAP τρέχει πάνω από το επίπεδο μεταφοράς ενός δικτύου, στην περίπτωση του Διαδικτύου αυτό είναι το TCP. Χρησιμοποιεί τη δικτυακή διαστρωμάτωση TCP/IP για τα επίπεδα δικτύου και μεταφοράς, σε αντίθεση με την περίπλοκη διαστρωμάτωση του μοντέλου OSI.

Σε έναν LDAP κατάλογο είναι δυνατόν να υπάρχουν εγγραφές με πληροφορίες για άτομα, για οργανωτικές μονάδες και γενικότερα για οτιδήποτε πόρους (servers, εκτυπωτές κλπ.)

Βασίζεται στο πρωτόκολλο X.500, αλλά είναι απλούστερο. Υποστηρίζει το πρωτόκολλο TCP/IP. Χρησιμοποιείται για την εύρεση οργανισμών, ατόμων και άλλων πόρων όπως αρχεία ή συσκευές όπως αναφέρθηκε και πιο πάνω.

Στη συγκεκριμένη περίπτωση η υπηρεσία καταλόγου παρέχει πληροφορίες για τους χρήστες του δικτύου του Χαροκοπείου. Η πληροφορία αυτή προέρχεται από το αρχείο χρηστών του κεντρικού υπολογιστή.

Ο κατάλογος του Πανεπιστημίου είναι διασυνδεδεμένος με την Υπηρεσία Ευρετηρίου του ΕΔΕΤ.

Επιστρέφοντας στην αρχική αναφορά της ενότητας 3.2.2, θα εξηγήσουμε ποια διαδικασία ακολουθήσαμε για να παραμετροποιήσουμε τον LDAP server του Πανεπιστημίου.

The screenshot shows the 'LDAP Resource' configuration step in the Icinga Configuration Wizard. The progress bar at the top indicates the current step is 'Configuration'. The form contains the following fields:

- Resource Name: icingaweb2\_idap
- Host: earth.hua.gr
- Port: 3268
- Encryption: None
- Root DN: DC=hua,DC=gr
- Bind DN: dltdemo11
- Bind Password: (masked with asterisks)

At the bottom, there are three buttons: 'Back', 'Next', and 'Validate Configuration'. A legend indicates that an asterisk (\*) denotes a required field.

**Εικόνα 11: Παραμετροποίηση LDAP Server Hua**

- Δηλώνουμε σαν directory server τον earth.hua.gr στην θύρα 3268
- Χρησιμοποιήσαμε για search root την έκφραση: DC=hua, DC=gr

Η ανωτέρω διαδικασία ακολουθήθηκε διότι μέσα από αυτή την υλοποίηση απέκτησα δικαιώματα “διαχειριστή”. Αυτό σημαίνει πρακτικά ότι πλέον υπάρχει προσβασιμότητα στο σύστημα διαχείρισης δικτύου του πανεπιστημίου μέσα από το link: [galaxy.hua.gr/icingaweb2/](http://galaxy.hua.gr/icingaweb2/) και έχοντας ως User id και password τα προσωπικά μου στοιχεία.

Οι παραμετροποιήσεις ωστόσο, όσο αφορά την εισαγωγή και μελέτη των συσκευών – μηχανημάτων του πανεπιστημίου θα καταχωρηθούν μέσα από το λειτουργικό σύστημα του debian8 απο το virtual box όπου και εγκαταστήσαμε όπως αναφέρθηκε στο κεφάλαιο 3.1.

## 4. Παραμετροποίηση

### 4.1 Διαδικασία απομακρυσμένης παρακολούθησης

Η ενότητα αυτή θα αναλύσει πως μετά το περάς των διαδικασιών που έγινε αναφορά για την εγκατάσταση ενός συστήματος ανοιχτού κώδικα (open source) για την παρακολούθηση του συστήματος μας, ποιές διαδικασίες ακολουθήσαμε έτσι ώστε να μπορούμε να βλέπουμε τα μηχανήματα και να τα παρακολουθούμε σχετικά με την λειτουργία τους.

#### 4.1.2 Παράδειγμα monitoring εξυπηρετητή

Εάν η απομακρυσμένη υπηρεσία παρακολούθησης είναι διαθέσιμη χρησιμοποιώντας ένα πρωτόκολλο δικτύου καθώς και πόρτας (port) και τα `check_plugins`, δεν χρειάζεται απαραίτητα η εγκατάσταση ενός τοπικού client. Αυτό σημαίνει ότι αντί να επιλέξουμε ένα plugin και να κάνουμε τις παραμετροποιήσεις πάνω σε αυτό, η βιβλιοθήκη προτύπων του icinga 2 έχει πολλές περιπτώσεις που μπορούνε να μας εξυπηρετήσουμε χωρίς την παραμετροποίηση των Plugins. Ονομαστικά αυτά είναι :

- ping4, ping6, fping4, fping6, hostalive
- tcp, udp, ssl
- http, ftp
- smtp, ssmtp, imap, simap, pop, spop
- ntp\_time
- ssh
- dns, dig, dhcp

Η ανωτέρω διαδικασία αναφέρεται και ως Agent-less Checks.

Agent-less Checks δεν βασίζονται σχετικά με εγκατεστημένο πρόγραμμα εξ' αποστάσεως, και αυτό είναι πολύ χρήσιμο για φορητές συσκευές οι οποίες δεν έχουν πολύ μνήμη για να φιλοξενήσουνε και κάποιο άλλο πρόγραμμα. Αυτό είναι επίσης χρήσιμο στην περίπτωση που ο client σας , σας προσφέρει μια με δύο υπηρεσίες και δεν αξίζει να αφιερώσετε πολύ χρόνο για την εγκαταστήσετε αλλά και να ρυθμίσετε τους κόμβους (nodes). Για παράδειγμα, αν θέλετε να ελέγξετε αν είναι διαθέσιμο το SSH (secure shell) σε ένα απομακρυσμένο

υπολογιστή, η αν ο διακοσμητής http λειτουργεί η απλώς να δούμε αν ο άλλος υπολογιστής λειτουργεί κανονικά. Από αυτό θα ξεκινήσουμε και εμείς.

Υπάρχουν πολλοί τρόποι να ρυθμίσουμε το Icinga 2. Μπορούμε να διαλέξουμε τα ήδη προϋπάρχον hosts.conf η services.conf η δεδομένου ότι ο κατάλογος conf.d διαβάζετε κατά τη εκκίνηση μπορούμε να χρησιμοποιήσουμε ένα αρχείο ανά host. Θα χρησιμοποιήσουμε το τελευταίο λόγω απλότητας καθώς και για την πιο εύκολη τακτοποίηση των αρχείων. Το σημείο πρόσβασης (access point) ονομάζεται galaxy.conf. [5]

Πρώτα από όλα μπαίνουμε στο κατάλογο conf.d και ύστερα στο αρχείο galaxy.conf και στο οποίο πραγματοποιούμε τις διαδικασίες μέσα απο τις οποίες θα φθάσουμε να βλέπουμε το μηχανήμα δηλαδή το galaxy.conf δηλαδή τον server του πανεπιστημίου απομακρυσμένα.

- `cd /etc/icinga2/conf.d`
- `pico galaxy.conf` ή `getid galaxy.conf`

Έπειτα

```
/*
* File:      /etc/icinga2/conf.d/galaxy.conf
* Title:     galaxy.conf Host Configuration File
* Description: Host and services definition for Icinga2 for
               the galaxy server.
* Host:      galaxy.hua.gr
* System:    Debian
* License:    Creative Commons Zero - Public Domain
* Version:    1.0
* Date:      January 12, 2016
*/

//
// Host Declaration Block
//
object Host "galaxy.hua.gr" {

    // Define the host IPv4 Address
```

```

address      = "195.130.90.169"
// Define a basic functionality test
// Hostalive does a basic ICMP ECHO to the target
// specified in the address directive.
check_command = "hostalive"
}

```

Οι υπηρεσίες που υποστηρίζει αυτός ο εξυπηρετητής είναι δύο: HTTP, SSH.

Ο Apache είναι ο πιο διάσημος διακομιστής που χρησιμοποιείται στο web. Είναι διαθέσιμος για μια ποικιλία διαφορετικών λειτουργικών συστημάτων, με το UNIX να είναι το πιο διάσημο. Στόχος του HTTP Apache προγράμματος, είναι η δημιουργία ενός ασφαλούς, αποδοτικού και επεκτάσιμου web διακομιστή, ο οποίος θα μπορεί να παρέχει http υπηρεσίες, βάση των http προτύπων (standards). Ακριβώς από κάτω φαίνεται το configuration που χρειάστηκε:

```

//
// Service Declaration Block
// Service: HTTP
// Note: The address specified in the above directive
//      is carried into the service declaration
//      blocks via specifying the host_name variable
//

object Service "https" {
    host_name      = "galaxy.hua.gr"
    // Enable HTTPS checking
    vars.http_ssl   = "1"
    // Time till warning is issued
    vars.http_warn_time = "5"
    // Time till critical notification
    vars.http_critical_time = "10"
    check_command    = "http"
    // Change port
    vars.http_port = "10000"
    check_commaabd = "http"
}

```

Στην συνέχεια βλέπουμε το configuration για την υπηρεσία SSH. Το SSH (Secure Shell) είναι ένα ασφαλές δικτυακό πρωτόκολλο το οποίο επιτρέπει τη μεταφορά δεδομένων μεταξύ δύο υπολογιστών. Το SSH όχι μόνο κρυπτογραφεί τα δεδομένα που ανταλλάσσονται κατά τη συνεδρία, αλλά προσφέρει ένα ασφαλές σύστημα αναγνώρισης

```
//  
// Service Declaration Block  
// Service: SSH  
// Note: The address specified in the above directive  
//      is carried into the service declaration  
//      blocks via specifying the host_name variable  
//  
object Service "ssh" {  
  
    host_name    = "galaxy.hua.gr"  
    check_command = "ssh"  
}
```

Και τέλος

- service icinga2 restart
- service icinga2 status

Έτσι, με αυτή την διαδικασία πλέον μπορούμε να συνδεθούμε στην ιστοσελίδα μας, στο περιβάλλον του icinga2, και θα είναι ορατό το μηχανήμα και επίσης έχουμε τρεις ελέγχους γι' αυτό :An ICMP Echo (hostalive), A HTTPS Up Check (http), A SSH Up check (ssh). Οι έλεγχοι αυτοί θα αναλυθούνε στο επόμενο κεφάλαιο σχετικά με την χρησιμότητα τους και την λειτουργικότητάς τους.

Επίσης υπάρχει και η διαδικασία που ονομάζεται Agent-based Checks για Linux/Unix. Ο agent λειτουργεί ως δαίμονας και επικοινωνεί με τον master ζητώντας έναν έλεγχο που

εκτελείται ή τοπικές αποθηκευμένες πληροφορίες( SNMP OID). Στο Icinga 2 ο client συνεχίζει να εκτελεί ελέγχους όταν δεν υπάρχει σύνδεση και δεν χρειάζεται τον master ως προγραμματιστή ελέγχου όπως οι άλλοι agents. [3]

- Icinga 2 Client
- SSH
- SNMP
- NRPE

### 4.1.3 Παράδειγμα monitoring δικτυακής συσκευής

Το συγκεκριμένο μηχανήμα, το core switch το οποίο βρίσκεται στο κτίριο στην ομήρου στο τμήμα πληροφορικής και τηλεματικής και λειτουργεί με telnet(telecommunication NETwork) και SNMP(Simple Network Management Protocol) πρωτόκολλα. Αυτό σημαίνει ότι θα πρέπει να ρυθμίσουμε τις παραμετροποιήσεις μας με βάση αυτά τα στοιχεία έτσι ώστε να μπορούμε να έχουμε ορατό το μηχανήμα στο περιβάλλον του icinga2 και να μπορούμε στην συνέχεια να το ελέγξουμε.

Το πρώτο βήμα που πρέπει να κάνουμε είναι να γίνει έλεγχος για την εγκατάσταση των SNMP Plugins. Πληκτρολογώντας την εντολή:

- apt-get install nagios-snmp-plugins

Επιβεβαιώνουμε πως τα είχαμε εγκαταστήσει στην αρχή οπότε προχωρούμε στο επόμενο βήμα το οποίο είναι να δημιουργήσουμε ένα αρχείο στον κατάλογο conf.d (εξηγήθηκε στην ενότητα 3.3.1). Αυτό έγινε με τον εξής τρόπο:

- cd /etc/icinga2/conf.d
- vi omirou.conf

Και έπειτα προσθέτουμε τις παραμετροποιήσεις στο αρχείο μας έτσι ώστε να είναι ορατό το μηχανήμα στην διεπαφή του icinga2. Ο τρόπος με τον οποίο έγινε αυτό ακολουθείται ακριβώς από κάτω:



```
//
// Host Declaration Block
//
object Host "omirou_core" {
    // Define the host IPv4 Address
    address          = "10.100.51.1"
    // Define a basic functionality test
    // Hostalive does a basic ICMP ECHO to the target
    // specified in the address directive.
    check_command     = "hostalive"
    // Define SNMP Variables
    vars.snmp_address = "10.100.51.1"
    vars.snmp_community = "huaprivatehua"
    // These are not strictly needed. I add them
    // so I know at a glance what version of snmp
    // I am using.
    vars.snmp_v2      = "true"
    vars.snmp_v3      = "false"
    vars.os = "switch"
}
```

- service icinga2 restart
- service icinga2 status

Έπειτα από αυτή την διαδικασία πλέον το μηχάνημα είναι ορατό στην διεπαφή μας. [6]

## 4.2 Διαμόρφωση διαδικασίας ειδοποίησης μέσω email

Στο σημείο αυτό θα γίνει ανάλυση της διαδικασίας, πώς μετά την εγκατάσταση των διάφορων μηχανημάτων που επιθυμούμε να παρακολουθούμε από τον χώρο του πανεπιστημίου είναι εφικτό να λαμβάνουμε ειδοποίηση μέσω ηλεκτρονικού ταχυδρομείου όπου και αν βρισκόμαστε με την βοήθεια του SMTP (Simple Mail Transfer Protocol (SMTP)).

Ουσιαστικά, Το SMTP (Simple Mail Transfer Protocol) είναι ένα πρωτόκολλο για την αποστολή μηνυμάτων ηλεκτρονικού ταχυδρομείου μεταξύ των server. Τα περισσότερα συστήματα ηλεκτρονικού ταχυδρομείου για να στείλουν μηνύματα από έναν server σε έναν άλλον μέσω Διαδικτύου χρησιμοποιούν το SMTP και έπειτα τα μηνύματα αυτά μπορούν να ανακτηθούν με προγράμματα e-mail client χρησιμοποιώντας POP ή IMAP πρωτόκολλο. Γενικά, το SMTP πρωτόκολλο χρησιμοποιείται για να στείλει τα μηνύματα από έναν mail client σε έναν mail server. Γι' αυτό πρέπει να διευκρινίζεται και ο POP ή IMAP server καθώς και ο SMTP όταν διαμορφώνετε την εφαρμογή ηλεκτρονικού ταχυδρομείου σας.

Το πρώτο βήμα για αυτή την διαδικασία ήταν να βρούμε τον σωστό κατάλογο (directory) έτσι ώστε να παραμετροποιήσουμε σωστά το αρχείο που χρειαζόμαστε.

Αρχικά, είμαστε στον κατάλογο `conf.d` και έπειτα πηγαίνουμε στο αρχείο `user.conf` και ακολουθούμε την διαδικασία αυτή:

- `cd /etc/icinga2/conf.d`
- `ls -l`
- `pico user.conf`

Και προσθέτουμε τη παραμετροποίηση:

```
object User "Zoi Farmaki hua" {  
  
  import "generic-user"  
  display_name = "Zoi Farmaki 2 Admin"  
  groups = [ "icingaadmins" ]  
  email = "itp14210@hua.gr"  
}  
object UserGroup "icingaadmins" {  
  display_name = " Admins Group"  
}
```

Και στην συνέχεια

- service icinga2 restart

Στην παραπάνω παραμετροποίηση μας δίνεται η δυνατότητα να ορίσουμε σε ποιο email η emails θα λαμβάνουμε τις ειδοποιήσεις. Πρέπει να αναφέρουμε ότι η διαδικασία αυτή είναι επιτυχής μόνο για emails του Χαροκοπέιου και πιο συγκεκριμένα για τους χρήστες που είναι στον LDAP (Lightweight Directory Access Protocol). Αναλυτικές λεπτομέρειες σχετικά με την λειτουργία του αναφέρθηκαν στην ενότητα 3.2.3)

Ολοκληρώνοντας την διαδικασία αυτή, ήμασταν έτοιμοι να λάβουμε ειδοποιήσεις στα emails που εισάγαμε προηγουμένως. Η συγκεκριμένη διαδικασία, μόνο με αυτή την παραμετροποίηση δεν δούλεψε πρακτικά. Ουσιαστικά, υπήρχαν στην διεπαφή του icinga web2 τα στοιχεία που εισάγαμε απλώς η δυνατότητα να λάβουμε ειδοποίηση, ενώ μας ενημέρωνε ότι στάλθηκε ειδοποίηση, δεν ήταν εφικτή. Στην παράρτημα Β αναλύεται η επιπρόσθετη διαδικασία παραμετροποίησης που χρειάστηκε για να δουλέψει η υπηρεσία αυτή.

## 4.3 Χρονοπρογραμματισμός ειδοποιήσεων μέσω email

Εφόσον τέθηκε σε λειτουργία η υπηρεσία των ειδοποιήσεων στα emails που είχαμε ορίσει, έπρεπε να ορίσουμε τους χρόνους που επιθυμούμε να γίνεται αυτό και ανά πόσα τακτικά χρονικά διάστημα. Κατανοούμε ότι, η πολύ τακτική αποστολή ειδοποιήσεων στο email μας θα δημιουργούσε πρόβλημα στην λειτουργία τους και επίσης θα ήταν δύσχρηστο και σε εμάς να διαμορφώνουμε την στοίβα των email μας όπως επιθυμούμε.

Το icinga web 2 ως πρωτοποριακό λογισμικό ανοιχτού κώδικα και ένα από τα καλύτερα της εποχής, φρόντισε να παρέχει στους χρήστες του την δυνατότητα της παραμετροποίησης των χρόνων που επιθυμούμε να λαμβάνουμε ειδοποιήσεις.

Κάτι τέτοιο γίνεται υλοποιήσιμο με την εξής διαδικασία:

- `cd /etc/icinga2/conf.d`
- `ls -l`
- `pico templates.conf`

Και έπειτα προσθέτουμε τα διάστημα που επιθυμούμε να λαμβάνουμε ειδοποιήσεις για τους hosts πρώτα όπως φαίνεται στην διαδικασία ακριβώς από κάτω:

```
/*
 * Generic template examples.
 */
/**
 * Provides default settings for hosts. By convention
 * all hosts should import this template.
 *
 * The CheckCommand object `hostalive` is provided by
 * the plugin check command templates.
 * Check the documentation for details.
 */
template Host "generic-host" {
    max_check_attempts = 3
    check_interval = 2m
    retry_interval = 1m
    check_command = "hostalive"
}
```

Συγκεκριμένα έχουμε ορίσει για τους hosts να γίνονται 3 προσπάθειες, σε διάστημα χρόνου 1m και να ξανά γίνεται προσπάθεια ελέγχου μετά από 1m.

Κάτι αντίστοιχο γίνεται στην συνέχεια αλλά για την παραμετροποίηση των χρόνων για ειδοποιήσεις στα services. Αυτό εφαρμόζεται με τον ακόλουθο τρόπο:

```
/*
 * Generic template examples.
 */
/**
 * Provides default settings for services. By convention
 * all services should import this template.
 */
template Service "generic-service" {
    max_check_attempts = 5
    check_interval = 1m
    retry_interval = 30s
}
```

Εδώ έχουμε ορίσει για τα services να γίνονται 5 προσπάθειες, σε διάστημα χρόνου 1m και μετά να ξανά γίνεται προσπάθεια ελέγχου μετά από 30s θέτοντας έτσι σε μεγαλύτερη προτεραιότητα για εμάς τα services. [4]

Οι χρόνοι αναπροσαρμόζονται ανά πάσα στιγμή, ανάλογα με την κατάσταση του συστήματος μας αποσκοπώντας στο να μην το φορτώνουμε πολύ με αυτούς τους ελέγχους.

## 4.4 Το conf.d directory γενικότερα

Ο κατάλογος conf.d περιέχει τύπους διαμόρφωσης που μας βοήθησαν να ξεκινήσουμε την παρακολούθηση του τοπικού κεντρικού υπολογιστή και τις υπηρεσίες του. Σε αυτό τον κατάλογο περιλαμβάνεται το αρχείο ρυθμίσεων του icinga2.conf από προεπιλογή. Μπορεί να χρησιμοποιηθεί ως παράδειγμα αναφοράς για τη δική μας στρατηγική διαμόρφωσης. Απλά έπρεπε να είχαμε κατά νου να περιλαμβάνει τους βασικούς καταλόγους στο αρχείο icinga2.conf. Κάτι τέτοιο δεν είναι δεσμευτικό, μπορούμε να το αφαιρέσουμε και να

διαμορφώσουμε τον δικό μας τρόπο για την ανάπτυξη της διαμόρφωσης του Icinga 2. Ωστόσο, όπως διαπιστώθηκε και στην ενότητα 4.1, 4.2 και 4.3 επιλέξαμε να κρατήσουμε τον κατάλογο ως έχει με την ονοματολογία `icinga2.conf`.

Περαισσότερες λεπτομέρειες σχετικά με τη διαμόρφωση βέλτιστων πρακτικών κ πως δημιουργήσαμε τη δική μας στρατηγική περιγράφεται σε αυτό το κεφάλαιο.

Διαθέσιμα αρχεία ρυθμίσεων που εγκαθίστανται από προεπιλογή:

- `hosts.conf`
- `services.conf`
- `users.conf`
- `notifications.conf`
- `commands.conf`
- `groups.conf`
- `templates.conf`
- `downtimes.conf`
- `timeperiods.conf`
- `satellite.conf`
- `apt.conf`

### ***host.conf***

Το αρχείο `hosts.conf` περιέχει ένα παράδειγμα υποδοχής με βάση τη ρύθμιση `NodeName` μας στο `constants.conf`. Μπορούμε να χρησιμοποιήσουμε τις παγκόσμιες σταθερές για τα ονόματα αντικειμένου μας, αντί με τιμές συμβολοσειράς. Η λέξη-κλειδί εισαγωγής χρησιμοποιείται για την εισαγωγή του προτύπου `host template` που φροντίζει για τη εγκατάσταση των ελέγχων των `host` με την εντολή `hostalive`.

Τα χαρακτηριστικά των `vars` μπορούν να χρησιμοποιηθούν για να καθορίσουν προσαρμοσμένα χαρακτηριστικά που είναι διαθέσιμα για τις εντολές ελέγχου και ειδοποιήσεων. Τα περισσότερα από τις εντολές των `plugin Check` στην Βιβλιοθήκη του Icinga απαιτούν μια χαρακτηριστική διεύθυνση.

Το συνηθισμένο χαρακτηριστικό του `os` αξιολογείται από το `group linux-servers` στο `directory groups.conf` καθιστώντας το `local host` μέλος.

Το παράδειγμα του `host` θα μας δείξει τι εννοούμε ακριβώς:

- Καθορίζουν τα χαρακτηριστικά *http\_vhost* για την υπηρεσία http εφαρμόζοντας του κανόνες που ορίζεται στο *services.conf*.
- Καθορίζουν τους δίσκους και των χαρακτηριστικών τους εφαρμόζοντας τον κανόνα που ορίζεται στο *services.conf*.
- Καθορίζουν τους τύπους ειδοποιήσεων(mail) και ποιες ομάδες αποδίδουν. Αυτό θα χρησιμοποιηθεί από τις ρυθμίσεις ειδοποιήσεων που εφαρμόζεται στο *notifications.conf*.

Στην ενότητα 4.1.2 και 4.1.3 φαίνεται και το configuration που ακλουθήσαμε έτσι ώστε με τις αντίστοιχες τροποποιήσεις να υπάρχει η ανάλογη τροποποίηση στο περιβάλλον μας.

### *services.conf*

Αυτές οι υπηρεσίες εφαρμόζουν κανόνες που μας δείχνουν πως θα παρακολουθήσουμε ένα local host αλλά και πως μας επιτρέπει να επαναχρησιμοποιήσουμε ή να το τροποποιήσουμε για τις δικές σας απαιτήσεις. Θα πρέπει να έχουμε καθορίσει όλες τις υπηρεσίες μας εφαρμόζοντας τους κανόνες στον κατάλογο *services.conf* οποιαδήποτε άλλη κεντρική τοποθεσία διατηρούνται οργανωμένα. Από προεπιλογή, το local host θα παρακολουθεί τις ακόλουθες υπηρεσίες:

SERVICES	APPLIED ON HOSTS
load, procs, swap, users, icinga	The <i>NodeName</i> host only
ping4, ping6	All hosts with <i>address</i> resp. <i>address6</i> attribute
ssh	All hosts with <i>address</i> and <i>vars.os</i> set to Linux
http, optional: Icinga Web 2	All hosts with custom attribute <i>http_vhosts</i> defined as dictionary
disk	All hosts with custom attribute <i>disks</i> defined as dictionary

Ο κώδικας διαμόρφωσης για αυτά ακλουθείτε αμέσως:

Η υπηρεσίες του ping4 και του ping6 μας δείχνουμε πόσα πακέτα χάθηκαν και για πόσο χρονικό διάστημα.

```

/*
 * These are generic `ping4` and `ping6`
 * checks applied to all hosts having the
 * `address` resp. `address6` attribute
 * defined.
 */
apply Service "ping4" {
    import "generic-service"
    check_command = "ping4"
    assign where host.address
}
apply Service "ping6" {
    import "generic-service"
    check_command = "ping6"
    assign where host.address6
}

```

Το SSH (Secure Shell) είναι ένα ασφαλές δικτυακό πρωτόκολλο το οποίο επιτρέπει τη μεταφορά δεδομένων μεταξύ δύο υπολογιστών. Το SSH όχι μόνο κρυπτογραφεί τα δεδομένα που ανταλλάσσονται κατά τη συνεδρία, αλλά προσφέρει ένα ασφαλές σύστημα αναγνώρισης

```

*
 * Apply the `ssh` service to all hosts
 * with the `address` attribute defined and
 * the custom attribute `os` set to `Linux`.
 */
apply Service "ssh" {
    import "generic-service"
    check_command = "ssh"
    assign where (host.address || host.address6) && host.vars.os == "Linux"
    ignore where host.name == "localhost" /* for upgrade safety */
}

```

Στόχος του HTTP Apache, είναι η δημιουργία ενός ασφαλούς, αποδοτικού και επεκτάσιμου web διακομιστή, ο οποίος θα μπορεί να παρέχει http υπηρεσίες, βάση των http προτύπων (standards).



```
apply Service for (http_vhost => config in host.vars.http_vhosts) {  
  import "generic-service"  
  check_command = "http"  
  vars += config  
}
```

Ελέγχει τον χώρο του δίσκου

```
apply Service for (disk => config in host.vars.disks) {  
  import "generic-service"  
  check_command = "disk"  
  vars += config  
}
```

Ελέγχει την κατάσταση του icinga

```
apply Service "icinga" {  
  import "generic-service"  
  check_command = "icinga"  
  assign where host.name == NodeName  
}
```

Κάνει έλεγχο στην CPU του συστήματος και προγραμματίζει πότε θα γίνει backup του συστήματος

```
apply Service "load" {  
  import "generic-service"  
  check_command = "load"  
  /* Used by the ScheduledDowntime apply rule in `downtimes.conf`. */  
  vars.backup_downtime = "02:00-03:00"  
  assign where host.name == NodeName  
}
```

Εμφανίζει αν τρέχει κάποια συγκεκριμένη διαδικασία/ process η διαδικασίες.

```
apply Service "procs" {  
  import "generic-service"  
  check_command = "procs"  
  assign where host.name == NodeName  
}
```

Το swap μας δείχνει τι όρια υπάρχουνε για την CPU του συστήματος. Τι χώρος χρησιμοποιείται κ τι είναι διαθέσιμος.

```
apply Service "swap" {  
  import "generic-service"  
  check_command = "swap"  
  assign where host.name == NodeName  
}
```

Η υπηρεσία αυτή μας δείχνει πόσοι χρήστες συνδέθηκαν

```
apply Service "users" {  
  import "generic-service"  
  check_command = "users"  
  assign where host.name == NodeName  
}
```

Επίσης έχει αναλυθεί και στο κεφάλαιο 4.1.2 και 4.1.3.

### ***users.conf***

Καθορίζει τον icingaadmin User και το icingadmins UserGroup. Το τελευταίο χρησιμοποιείται στο hosts.conf για τον καθορισμό χαρακτηριστικών του host, αργότερα χρησιμοποιείται στο notification.conf για ειδοποιήσεις εφαρμόζοντας του κανόνες.

Μπορούμε να ανατρέξουμε στο κεφάλαιο 4.2 σελίδα 56 για να δούμε το configuration στο συγκεκριμένο αρχείο.

### ***notifications.conf***

Οι Ειδοποιήσεις ελέγχων αποτελούν αναπόσπαστο μέρος για την παρακολούθηση του συστήματος διαχείρισης μας. Σε αυτό το αρχείο ορίζουμε δυο ειδοποιήσεις εφαρμόζοντας τους κανόνες για τα hosts και τα services. Και τα δυο ισχύουν για την ίδια κατάσταση: Οι συντελεστές εφαρμόζονται μόνο εάν το ένθετο λεξικό χαρακτηριστικό notification.mail έχει οριστεί. Η λέξη-κλειδί είναι σημαντική για τις ενημερώσεις, διότι εφαρμόζονται κανόνες που καθορίζουν το αν αυτές οι ειδοποιήσεις ισχύει για τα hosts και τα services. Η λέξη-κλειδί εισάγει τα συγκεκριμένα πρότυπα στο mail templates που καθορίζεται στο αρχείο template.conf. Το διάστημα δεν έχει οριστεί - από προεπιλογή είναι για 30 λεπτά.

Με το καθορισμό των user.groups η τιμή που παρέχεται από το αντίστοιχο host.vars.notification.mail θα χρησιμοποιηθεί στην ομάδα icingaadmins UserGroup που ορίζεται στο users.conf. Ακολουθεί ο κώδικας παραμετροποίησης:

```
/**
 * The example notification apply rules.
 *
 * Only applied if host/service objects have
 * the custom attribute `notification` defined
 * and containing `mail` as key.
 *
 * Check `hosts.conf` for an example.
 */
apply Notification "mail-icingaadmin" to Host {
    import "mail-host-notification"
    user_groups = host.vars.notification.mail.groups
    assign where host.vars.notification.mail
}
apply Notification "mail-icingaadmin" to Service {
    import "mail-service-notification"
    user_groups = host.vars.notification.mail.groups
    assign where host.vars.notification.mail
}
```

### ***groups.conf***

Ουσιαστικά σε αυτό το αρχείο μας δίνεται η δυνατότητα να ομαδοποιήσουμε τα services μας και τα hosts μας έτσι ώστε να είναι πιο εύκολη η παρακολούθησής τους. Στο επόμενο κεφάλαιο θα αναλυθεί η συγκεκριμένη διαδικασία με screenshots.

Στο πρώτο configuration ομαδοποιούμε τα hosts με τους Linux servers να εμφανίζονται όλοι μαζί στο πεδίο με όνομα “Linux”, ακριβώς το ίδιο για τους windows servers αλλά στο πεδίο με τίτλο “Windows” και τέλος για τα switch να διακρίνονται στο πεδίο με τίτλο “switch”

```
/**
 * Host group examples.
 */

object HostGroup "linux-servers" {
  display_name = "Linux Servers"
  assign where host.vars.os == "Linux"
}

object HostGroup "windows-servers" {
  display_name = "Windows Servers"
  assign where host.vars.os == "Windows"
}

object HostGroup "switch" {
  display_name = "switch"
  assign where host.vars.os == "switch"
}
```

Στο δεύτερο μέρος βλέπουμε πως ομαδοποιούνται οι υπηρεσίες των services. Μια καρτέλα για το καθένα δηλαδή: στην καρτέλα “ping” θα υπάρχουν όλοι οι hosts που χρησιμοποιούν την υπηρεσία αυτή, κάτι αντίστοιχο για το “http” και για το “disk”

```
/**
 * Service group examples.
 */

object ServiceGroup "ping" {
  display_name = "Ping Checks"
  assign where match("ping*", service.name)
}

object ServiceGroup "http" {
  display_name = "HTTP Checks"
```

```

    assign where match("http*", service.check_command)
}
object ServiceGroup "disk" {
    display_name = "Disk Checks"
    assign where match("disk*", service.check_command)
}

```

### *templates.conf*

Μέσα από αυτό το αρχείο μας δίνεται η δυνατότητα του χρονοπρογραμματισμού ειδοποιήσεων. Βλέπε κεφάλαιο 4.3.

### *downtimes.conf*

Η υπηρεσία *load* εφαρμόζει κανόνες που ορίζονται στο *services.conf* για την δημιουργία αντιγράφων ασφάλειας σε περίπτωση διακοπής της λειτουργίας του συστήματος. Η προγραμματισμένη *downtime* εφαρμόζει κανόνες που χρησιμοποιούνται σε χαρακτηριστικά τα οποία καθορίζουν τις προεπιλεγμένες τιμές για συγκεκριμένο χρονικό εύρος που απαιτείται για τις επαναλαμβανόμενες υποδοχές *downtime*. Το configuration αυτού:

```

**
* The example downtime apply rule.
*/
apply ScheduledDowntime "backup-downtime" to Service {
    author = "icingaadmin"
    comment = "Scheduled downtime for backup"
    ranges = {
        monday = service.vars.backup_downtime
        tuesday = service.vars.backup_downtime
        wednesday = service.vars.backup_downtime
        thursday = service.vars.backup_downtime
        friday = service.vars.backup_downtime
        saturday = service.vars.backup_downtime
        sunday = service.vars.backup_downtime
    }
    assign where service.vars.backup_downtime != ""
}

```

## *timeperiods.conf*

Αυτό το αρχείο περιέχει τις προεπιλεγμένες περιόδους σε προκαθορισμένο χρόνο για 24x7, 9to5 και ποτέ. Τα χρονικά διάστημα των αντικειμένων αναφέρονται σε Αντικείμενα περιόδου όπως hosts, services και notifications. Το configuration αυτού:

```
/**
 * Sample timeperiods for Icinga 2 requiring
 * 'legacy-timeperiod' template from the Icinga
 * Template Library (ITL).
 * Check the documentation for details.
 */
object TimePeriod "24x7" {
    import "legacy-timeperiod"
    display_name = "Icinga 2 24x7 TimePeriod"
    ranges = {
        "monday"      = "00:00-24:00"
        "tuesday"     = "00:00-24:00"
        "wednesday"   = "00:00-24:00"
        "thursday"    = "00:00-24:00"
        "friday"      = "00:00-24:00"
        "saturday"    = "00:00-24:00"
        "sunday"      = "00:00-24:00"
    }
}

object TimePeriod "9to5" {
    import "legacy-timeperiod"
    display_name = "Icinga 2 9to5 TimePeriod"
    ranges = {
        "monday"      = "09:00-17:00"
        "tuesday"     = "09:00-17:00"
        "wednesday"   = "09:00-17:00"
        "thursday"    = "09:00-17:00"
        "friday"      = "09:00-17:00"
    }
}

object TimePeriod "never" {
    import "legacy-timeperiod"
    display_name = "Icinga 2 never TimePeriod"
    ranges = {
    }
}
```

### *satellite.conf*

Περιλαμβάνει στάνταρ, τυποποιημένους, βασικούς πίνακες και αλληλεξαρτήσεις για να ελέγχεις απομακρυσμένους πελάτες που χρησιμοποιούν την εύρεση της υπηρεσίας και να κάνει config ο master. Μπορεί να αγνοηθεί/αφαιρεθεί στις ρυθμίσεις που δεν χρησιμοποιούν αυτό το χαρακτηριστικό. Συγκεκριμένα εμείς δεν χρησιμοποιήσαμε το αρχείο αυτό ωστόσο παραθέτω τον κώδικα παραμετροποίησης:

```
*
* Host and Service templates for the Agent Setup.
*/
/**
* Provides settings for satellite hosts managed by 'icinga2 repository'.
* Define your global attributes here, for example custom
* attributes used for notifications, etc.
*/
template Host "satellite-host" {
    vars.notification["mail"] = {
        groups = [ "icingaadmins" ]
    }
}
/**
* Provides settings for satellite services managed by 'icinga2 repository'.
* Define your global satellite attributes here, for example custom
* attributes used for notifications, etc.
*/
template Service "satellite-service" {
    vars.notification["mail"] = {
        groups = [ "icingaadmins" ]
    }
}
apply Dependency "satellite-host" to Host {
    parent_host_name = host.zone
    assign where host.zone != "" && "satellite-host" in host.templates
}
```

### *apt.conf*

Παρέχει το προεπιλεγμένο αντικείμενο της εφαρμογής του Icinga που ονομάζεται “apt” για πρόσθετες ρυθμίσεις, όπως για τα πόσα πακέτα είναι διαθέσιμα για αναβαθμίσεις.

```
apply Service "apt" {  
  import "generic-service"  
  check_command = "apt"  
  assign where host.name == NodeName  
}
```

Θα ήθελα να επισημάνω σε αυτό το σημείο συμπερασματικά πως στο κεφάλαιο 4.4 έγινε ανάλυση του καταλόγου/directory conf.d ως προς τον κώδικά σχετικά με τα services και τα configuration αρχεία. Δηλαδή τι αλλάζει το κάθε .conf σε σχέση με το περιβάλλον. Στο επόμενο κεφάλαιο θα γίνει ανάλυση ως προς αυτή την αλλαγή δηλαδή για περιβάλλον του συστήματος διαχείρισης που έχουμε υλοποιήσει με επεξηγηματικές εικόνες. Τι αλλαγή δηλαδή δημιουργεί στο περιβάλλον και πως τροποποιήσουμε τα αρχεία που αναφέρθηκαν προηγουμένως. [4]

.

## 4.5 Αναβάθμιση

Μέχρι την στιγμή της συγγραφής αυτής της διπλωματικής εργασίας δεν υπήρξε κάποια προβλεπόμενη διαδικασία αναβάθμισης του icinga web 2. Αναφερόταν πως αναμένονταν για το σύντομο κοντινό μέλλον.

## 4.6 Απεγκατάσταση

Η διαδικασία απεγκατάστασης είναι πολύ εύκολη και απαραίτητη να την γνωρίζουμε. Η απεγκατάσταση του συστήματος ίσως φανεί χρήσιμη σε περιπτώσεις που πρέπει να ξανά εγκαταστήσουμε το σύστημα από την αρχή, π.χ. σε ένα αποτυχημένο αποτέλεσμα απόπειρας αναβάθμισης.

1. Διαγράφουμε το icinga2

- **sudo apt-get remove icinga2**

2. Απεγκατάσταση των dependencies

- **sudo apt-get remove - -auto-remove icinga2**



Αυτό θα αφαιρέσει από το icinga2 τα package Αυτό θα αφαιρέσει το πακέτο icinga2 και όποια άλλα πακέτα εξαρτώνται από αυτό και δεν χρειάζονται πλέον.

3. Διαγραφή config/data

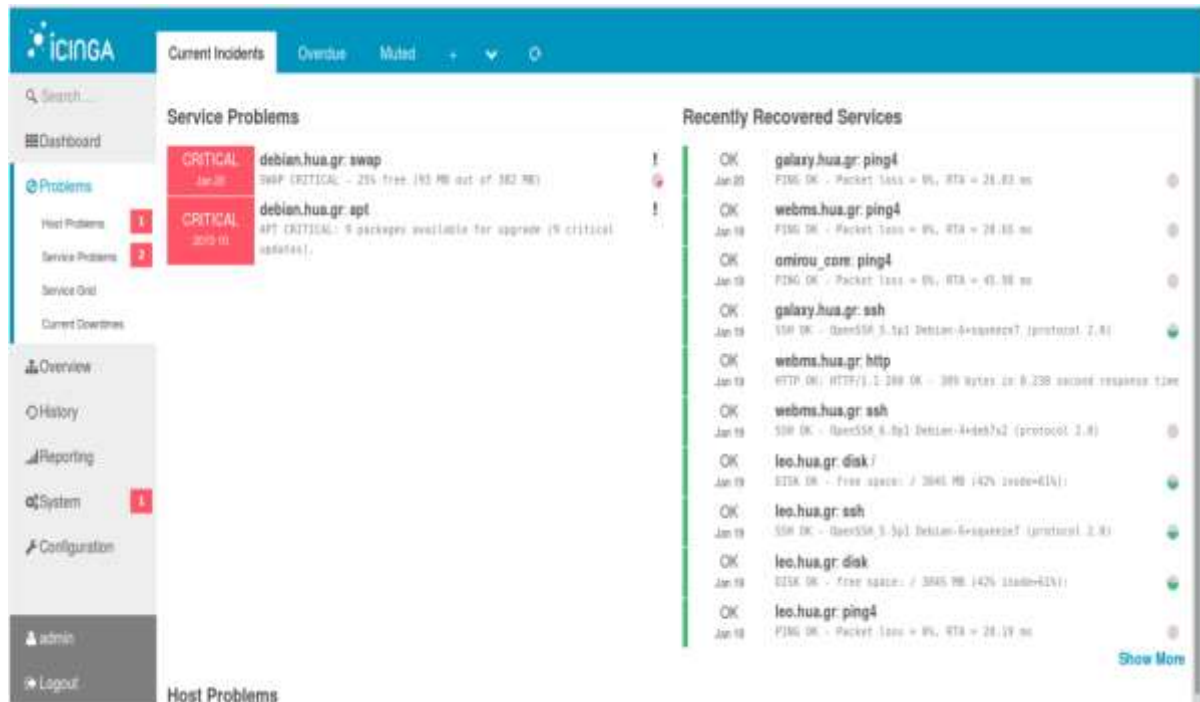
Αν επίσης θέλουμε να διαγράψουμε τα local / config αρχεία μας για το icinga2 τότε αυτό θα λειτουργήσει με την εξής διαδικασία

• **sudo apt-get purge icinga2**

## 5. Πιλοτική Λειτουργία

### 5.1 Η υποστηριζόμενη πλατφόρμα

Το icinga web 2 χρησιμοποιεί web interface έτσι ώστε να μπορεί να παρουσιάσει τις πληροφορίες που λαμβάνει από τους hosts και τις υπηρεσίες τους που γίνονται monitor. Ας ρίξουμε μια πρώτη ματιά στο interface που προσφέρει το icinga web 2:



Εικόνα 12: Αρχικό περιβάλλον του συστήματος διαχείρισης

Σε αυτή την εικόνα βλέπουμε την home του Icinga web 2. Όπως παρατηρείται αποτελείται από δυο frames, το αριστερό frame όπου χρησιμοποιείται ως menu επιλογών για τον χρήστη ενώ στο κεντρικό-δεξί frame εμφανίζονται οι πληροφορίες του κάθε option του αριστερού frame.

Αναλυτικά το menu πλοήγησης του icinga web2 :

- **Dashboard**

Είσοδος στην home σελίδα της διεπαφής του icinga web 2.

- **Problems**

Εμφανίζει πληροφορίες σχετικά με τα προβλήματα που εμφανίστηκαν κατά την διάρκεια του monitoring των hosts και των υπηρεσιών τους αλλά και τα Network outages (διακοπές δικτύου)

- **Host Problems**

Πληροφορίες σχετικά με τα hosts που γίνονται Monitor.

Μπορεί να είναι σε οποιαδήποτε από τις ακόλουθες καταστάσεις:

UP            The host is available.

DOWN        The host is unavailable.

- **Service problem**

Πληροφορίες σχετικά με τα services του κάθε host που γίνονται Monitor.

Μπορεί να είναι σε οποιαδήποτε από τις ακόλουθες καταστάσεις:

OK            Η υπηρεσία λειτουργεί σωστά

WARNING     Η υπηρεσία αντιμετωπίζει κάποια προβλήματα, αλλά εξακολουθεί να θεωρείται ότι είναι σε κατάσταση λειτουργίας

CRITICAL     Η υπηρεσία είναι σε κρίσιμη κατάσταση

UNKNOWN     Ο έλεγχος δεν μπορεί να προσδιορίσει την κατάσταση του server

- **Service Grid**

- **Current downtimes**

- **Overview**

- **Tactical Overview**

Συγκεντρωτικές πληροφορίες για το Monitoring στους hosts και στα service τους

- **Hosts**

Πληροφορίες σχετικά με τους hosts που γίνονται monitor.

- **Services**

Πληροφορίες σχετικά με τα services του κάθε host που γίνονται Monitor

- **Host groups**

Πληροφορίες σχετικά με ομαδοποιημένα hosts. Το icinga web2 έχει την δυνατότητα να παρέχει την ομαδοποίηση των hosts διότι μπορούμε να ελέγχουμε ταυτόχρονα (π.χ. routers, printers, DNS server κ.α.). Έτσι έχουμε την δυνατότητα να παίρνουμε συνολικές πληροφορίες γι αυτά εκτός από τις πληροφορίες που παίρνουμε για το καθένα ξεχωριστά( πεδίο Hosts) αλλά και σε περιπτώσεις που γίνονται Monitor πάρα πολλά objects γίνεται πιο εύκολη η παρακολούθηση μέσω της ομάδας στην οποία το έχουμε βάλει να ανήκει.

- **Service groups**

Πληροφορίες σχετικά με ομαδοποιημένες υπηρεσίες. Κάτι αντίστοιχο με τα hosts groups αλλά εδώ για υπηρεσίες.

- **Contact groups**

Δυνατότητα δημιουργίας πολλαπλών admin και επικοινωνίας μαζί τους.

- **Contact**

Δυνατότητα επικοινωνίας με έναν ή περισσότερους admin.

- **Comments**

Δυνατότητα παράθεσης σχολίου σχετικά με τα hosts και τα services. Αυτό σημαίνει ότι ο διαχειριστής σε αυτό το πεδίο βλέπει τί σχόλιο έχει κάνει βλέποντας το πρόβλημα.

- **Downtime**

Δυνατότητα δημιουργίας προγραμματισμένων απενεργοποιήσεων όσο αναφορά ένα ή περισσότερα hosts ή κάποιων services.

- **History**

Δυνατότητα δημιουργίας histogram.

- **Event grid**

Για να δούμε τα γεγονότα από πολλαπλούς μήνες ή έναν ολόκληρο χρόνο με μια ματιά. Το στιγμιότυπο αυτό δημιουργείται από ένα περιβάλλον που έχει μερικά εκατομμύρια γεγονότα στη βάση δεδομένων του. Χάρη στην ένταση του χρώματος, είναι εύκολο να διακρίνουμε ιδιαίτερα απασχολημένες ημέρες από τις υπόλοιπες.

- **Event overview**

Στην συνέχεια του “event grid” μέσω της IDODB, με ένα κλικ σε μια συγκεκριμένη ημέρα καλεί τον κατάλογο των φιλτραρισμένων γεγονότων τα οποία φαίνονται στο πεδίο ακριβώς από κάτω “event overview”. Από εδώ, μπορούν να εξαχθούν ή να φιλτράρονται περαιτέρω τα γεγονότα.

- **Notifications**

Αν κάποιο host ή service έχει κάνει enable τα notifications τότε με αυτή την επιλογή το icinga web2 αναγράφει όλα τα Notifications για αυτά τα services, hosts σε μια συγκεκριμένη μορφή σε top-down list. Επίσης στέλνονται με μορφή email στα contacts ή contacts groups του icinga web2 που ορίστηκαν και συνδέονται με το συγκεκριμένο host ή service.

Επίσης, σε ενημερώνει που έχει σταλθεί ειδοποίηση προβλήματος και τι τύπος προβλήματος είναι, όπως επίσης σου επιτρέπει να καταχωρήσεις κάποιο σχόλιο πάνω στο πρόβλημα αυτό.

- **Timeline**

Γραφική αναπαράσταση από το πέρασμα του χρόνου ως μια γραμμή όσο αναφορά: notifications, hard state changes, comments, acknowledgements, started downtimes, ended downtimes.

- **Reporting**

Παρέχει την δυνατότητα στον χρήστη να μπορεί να δημιουργήσει reports για το χρονικό διάστημα που θα ζητήσει ο χρήστης για κάποιο host - hostgroup - service - servicegroup.

- **Alert Summary**

Με τα alerts ο χρήστης έχει την δυνατότητα να δει κάθε session του icinga πληροφορίες σχετικά με αλλαγές σε καταστάσεις των services και των hosts αλλά και του ίδιου του icinga web2. Ο χρήστης έχει την δυνατότητα να δει σε διαρκεί από μιας ημέρας ως ένα χρόνο τις ειδοποιήσεις σε συνάρτηση με τις ελλείψεις και το μέσο όρο αντίδρασης.

- **System**

- **About**

Πληροφορίες σχετικά με το σύστημα όπως ονομασία, έκδοση,

- **Monitoring Health**

- **Process information**: Πληροφορίες σχετικά με το σύστημα σχετικά με την ημερομηνία τελευταίας παραμετροποιήσεις, έκδοση, υπενθύμιση ποιο services η host λειτουργεί η δεν λειτουργεί.
    - **Stats**: Συνοπτική εικόνα όλου του συστήματος σχετικά με την λειτουργία του.

- **Configuration**

Προβολή των objects που έχουν δημιουργηθεί.

- **Application**

- **Authentication**

Δυνατότητα δημιουργίας νέου χρήστη backend στο σύστημα η ακόμα και νέου group backend.

- **Authorization**

Δικαίωμα επεξεργασίας του ρόλου του διαχειριστή, μετονομασία ονόματος καθώς την άδεια δικαιωμάτων χρήσης (Permission role). Καθώς επίσης και δημιουργία νέου χρήστη ή group χρηστών.

- **Shared Navigation**

Κοινόχρηστη πλοήγηση στοιχείων.

- **Modules**

Έλεγχος σχετικά με τα module που έχουμε εγκαταστήσει. Ονομασία, έκδοση, περιγραφή που μας χρησιμεύει και σχετικά με την άδεια δικαιωμάτων. Σε εμάς στην εφαρμογή μας είναι τρία Module: monitoring, doc, setup.

- **Admin**

Δυνατότητα αλλαγής προτιμήσεων για τον χρήστη σχετικά με την γλώσσα που θέλει να χρησιμοποιεί το σύστημα και την ώρα της τοποθεσίας που βρίσκεται. Επίσης δίνεται η δυνατότητα κοινής χρήσης (navigation) ενός είδους στο menu του συστήματος.

- **Logout**

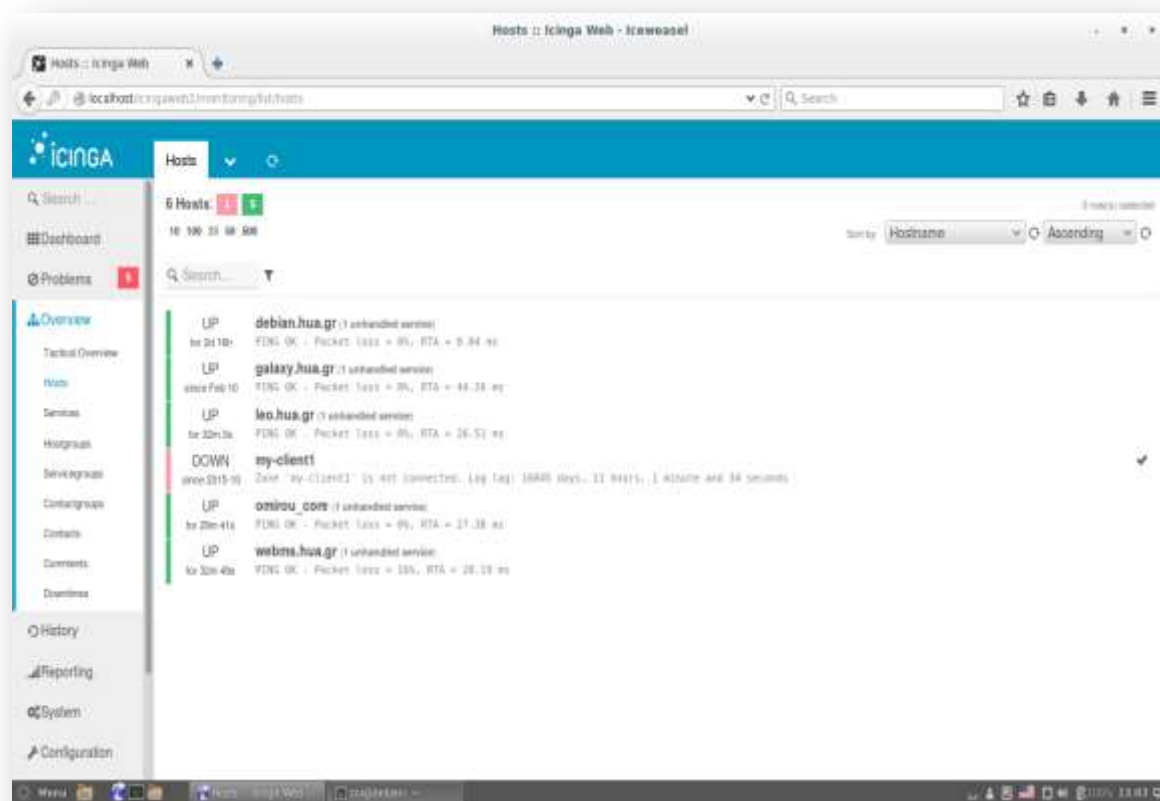
Έξοδος από το σύστημα.

## **5.2 Διαμόρφωση του web interface μέσα από το conf.d**

Στο κεφάλαιο 4.4 έγινε λεπτομερή ανάλυση σχετικά με τον κατάλογο/directory *conf.d* ο οποίος περιέχει τύπους διαμόρφωσης που μας βοήθησαν να ξεκινήσουμε την παρακολούθηση του τοπικού κεντρικού υπολογιστή και τις υπηρεσίες του. Στο κεφάλαιο αυτό θα δούμε πώς γίνεται αυτό δηλαδή τι αλλάζει το κάθε .conf σε σχέση με το περιβάλλον μας. Θα γίνει ανάλυση με screenshots και έπειτα θα ξεκινήσουμε την ανάλυση αυτή με την σειρά που αναφέρθηκαν στο κεφάλαιο 4.4 έτσι ώστε αν θέλουμε να ανατρέξουμε στον κώδικα να είναι πιο εύχρηστο για τον αναγνώστη.

### ***hosts.conf***

Το αρχείο αυτό αναλύθηκε για τον τρόπο διαμόρφωσης του στο κεφάλαιο 4.4 στην σελίδα 60. Στην κάτωθι εικόνα φαίνεται τι αλλαγές δημιουργεί στο περιβάλλον μας ως προς την χρήση του.

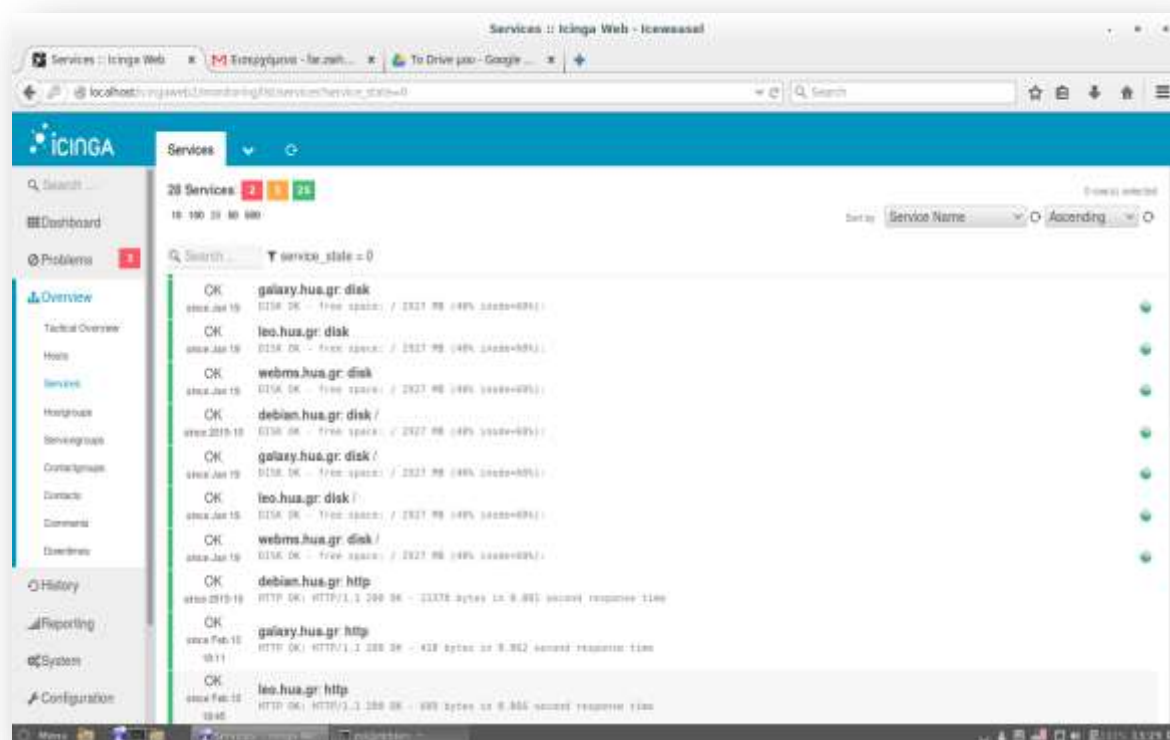


**Εικόνα 13: host.conf σε συνδυασμό με το web interface**

Στην εικόνα αυτή βλέπουμε με την διαδικασία που αναφέρθηκε στο προηγούμενο κεφάλαιο πως έχουμε ορατές πέντε υπηρεσίες/μηχανήματα προς παρακολούθηση μας. Βέβαια, θα ήθελα να επισημάνω πως για την δική μας διευκόλυνση ακλουθήσαμε το configuration που αναφέρεται στο προηγούμενο κεφάλαιο ωστόσο προτιμήσαμε να δημιουργήσουμε νέα αρχεία με την ονοματολογία *galaxy.conf*, *leo.conf*, *webms.conf* και *Omirou\_core.conf* για την διευκόλυνση μας. Το configuration παρέμεινε όπως αναφέρθηκε απλώς πραγματοποιήθηκαν τροποποιήσεις ανάλογα με το κάθε μηχάνημα.

### ***services.conf***

Στην εικόνα που ακολουθεί βλέπουμε πως μέσα από το αρχείο *services.conf* γίνονται ορατές σε εμάς υπηρεσίες που υποστηρίζει το κάθε μηχάνημα. Το πώς αυτές καταχωρηθήκαν μπορούμε να ανατρέξουμε στο προηγούμενο κεφάλαιο 4.4 σελίδα 61 για να το διευκρινίσουμε. Επιγραμματικά, οι υπηρεσίες αυτές είναι: *ping4*, *ping6*, *ssh*, *disk*, *load*, *procs* και *swap*. Κάποιες από αυτές φαίνονται και στην εικόνα που ακολουθεί:



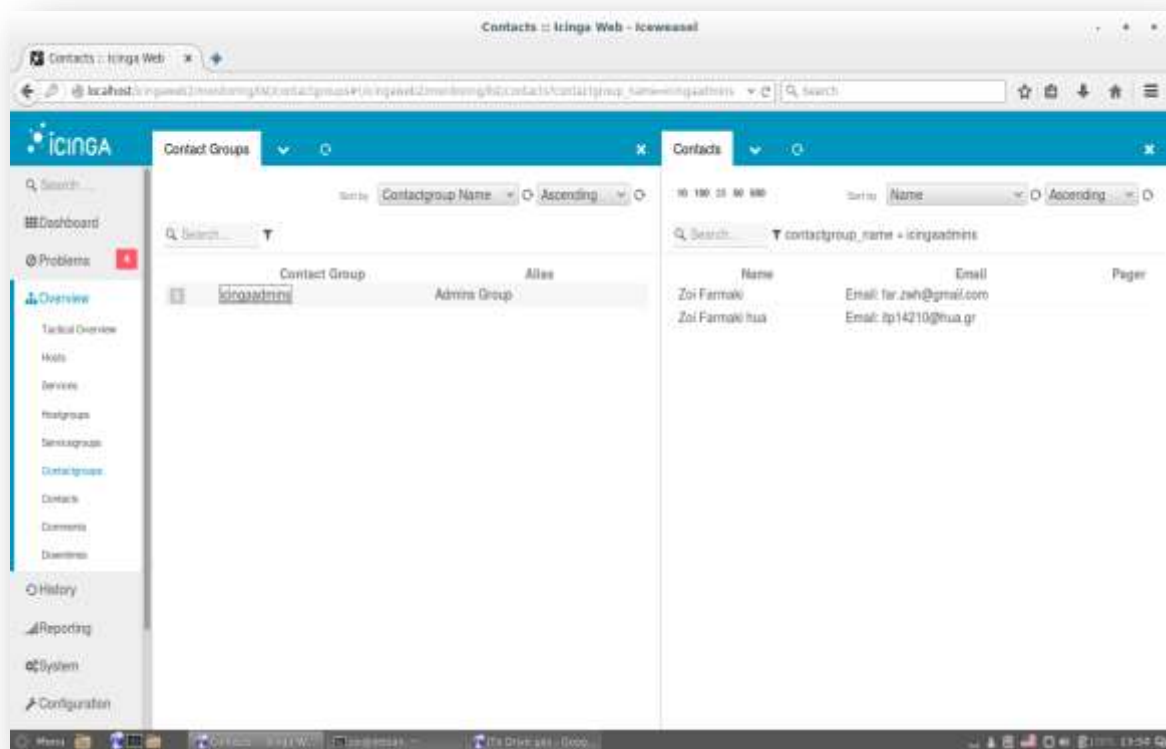
**Εικόνα 14: services.conf σε συνδυασμό με το web interface**

### *users.conf*

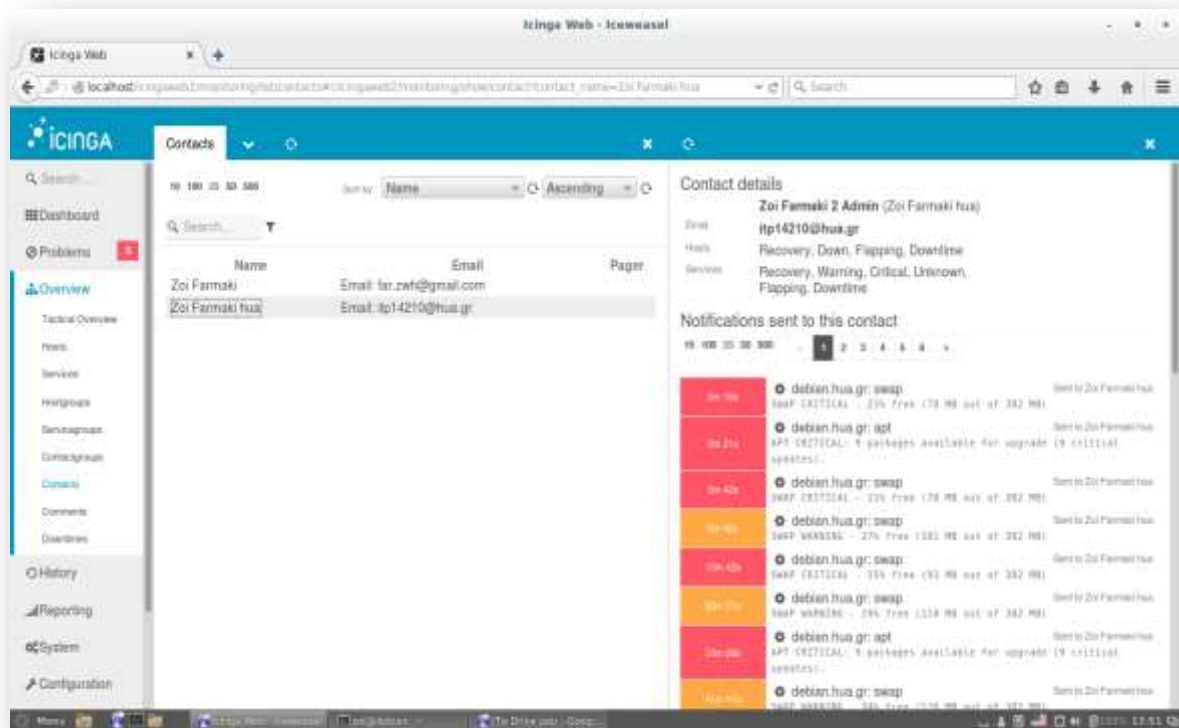
Στο συγκεκριμένο αρχείο που μπορούμε να ανατρέξουμε στην σελίδα 63 για να θυμηθούμε το configuration του, ουσιαστικά ορίζουμε ποιος θέλουμε να είναι ο διαχειριστής η οι διαχειριστές του συγκεκριμένου συστήματος διαχείρισης δικτύου και έπειτα σε ποιο ηλεκτρονικό ταχυδρομείο /email θέλουμε να λαμβάνουμε ειδοποιήσεις για τυχόν προβλήματα η απώλειες. Η εικόνα 32 ακριβώς από κάτω μας δείχνει ότι στο web interface στο πεδίο *contactgroups* που έχουμε ορίσει δύο διαχειριστές (ουσιαστικά ένας είναι απλώς θέλω να δείξω ότι υπάρχει δυνατότητα για να οριστούνε περισσότεροι από έναν διαχειριστή) με όνομα Zoi Farmaki και email επικοινωνίας [far.zwh@gmail.com](mailto:far.zwh@gmail.com) και έναν άλλον διαχειριστή με όνομα Zoi Farmaki hua με email επικοινωνίας [itp14210@hua.gr](mailto:itp14210@hua.gr). Στο σημείο αυτό θα μας βοηθούσε να ξανά ανατρέξουμε το κεφάλαιο 4.2 στην σελίδα 55 για την καλύτερη κατανόηση.

Στην εικόνα 33 μας δείχνει κάτι αντίστοιχο με όσα προαναφέρθηκαν απλώς πιο αναλυτικά: Δηλαδή όπως βλέπουμε, μας δείχνει τους διαχειριστές του συστήματος μας και κάνοντας click σε έναν από τους αυτούς έχουμε την δυνατότητα εικόνας σχετικά με τα προβλήματα που υπάρχουν στο σύστημα μας και σε ποιο email η emails σταλθήκαν ειδοποιήσεις όσο αναφορά τις απώλειες που είχαμε.



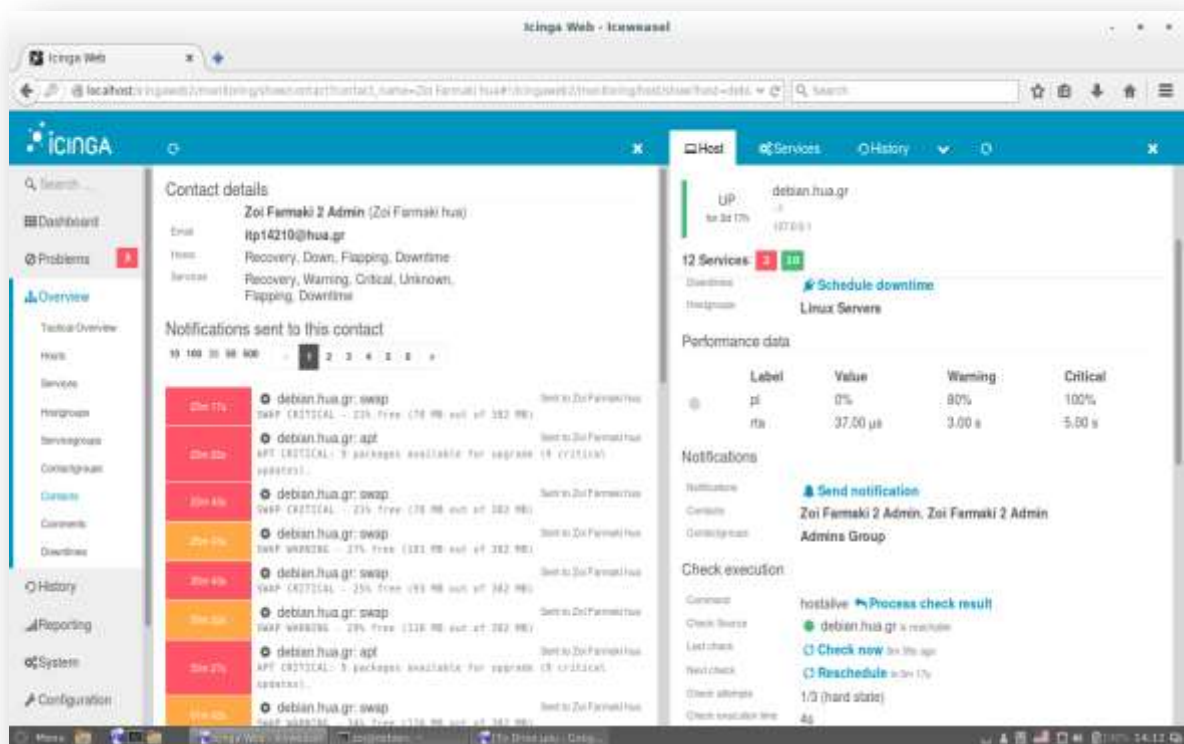


Εικόνα 15: users.conf



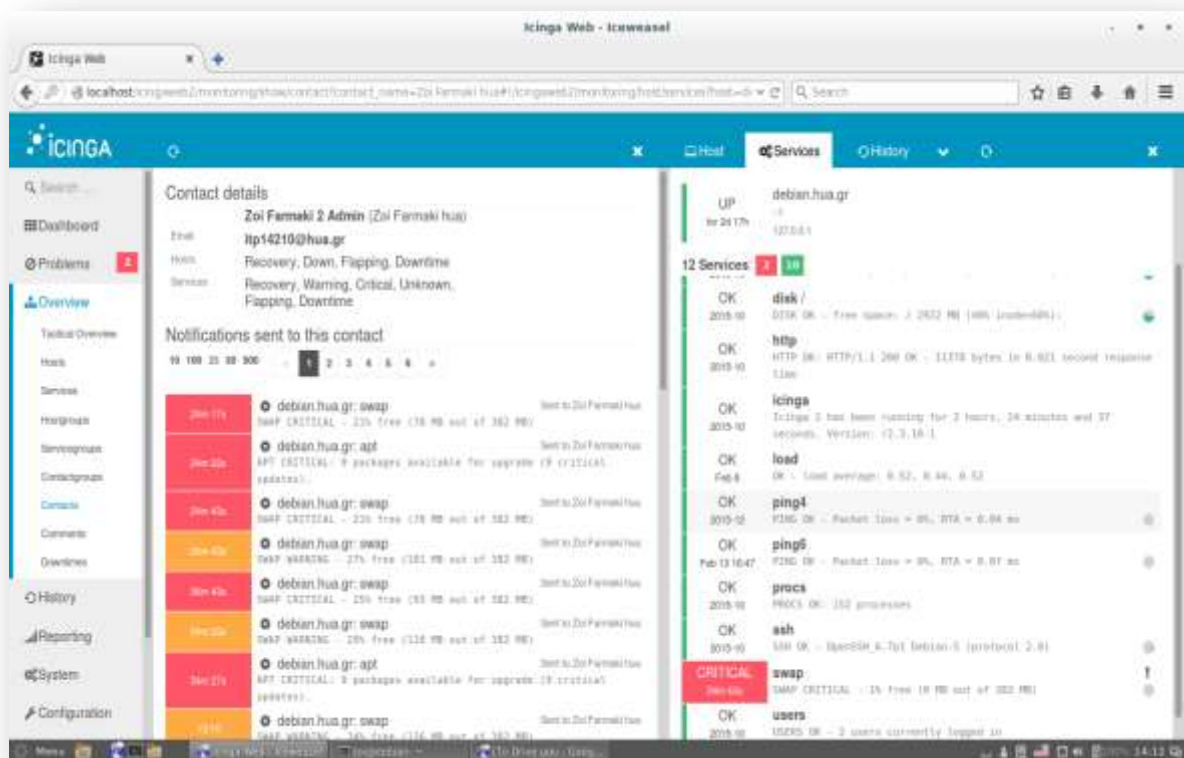
Εικόνα 16: users.conf (ii)

Έπειτα μας δίνεται η δυνατότητα όπως φαίνεται στην εικόνα 34 κάνοντας click σε κάποιο πρόβλημα που εμφανίζεται να βρισκόμαστε στο πεδίο *Host* βλέπουμε αν το μηχάνημα μας είναι σε λειτουργία, το προεπιλεγμένο στην συγκεκριμένη εικόνα μας έχει την ένδειξη «UP» οπότε λειτουργεί κανονικά. Ακριβώς από κάτω μπορούμε να δούμε ποια από τα services του λειτουργούν η ποια έχουν πρόβλημα, μπορούμε να στείλουμε κάποια επιπρόσθετη ειδοποίηση κατά την βούληση μας σε έναν από τους διαχειριστές, μπορούμε να αφήσουμε κάποιο σχόλιο για κάποιο πρόβλημα έτσι ώστε να μην μας ξανά σταλθεί ειδοποίηση για το συγκεκριμένο πρόβλημα.



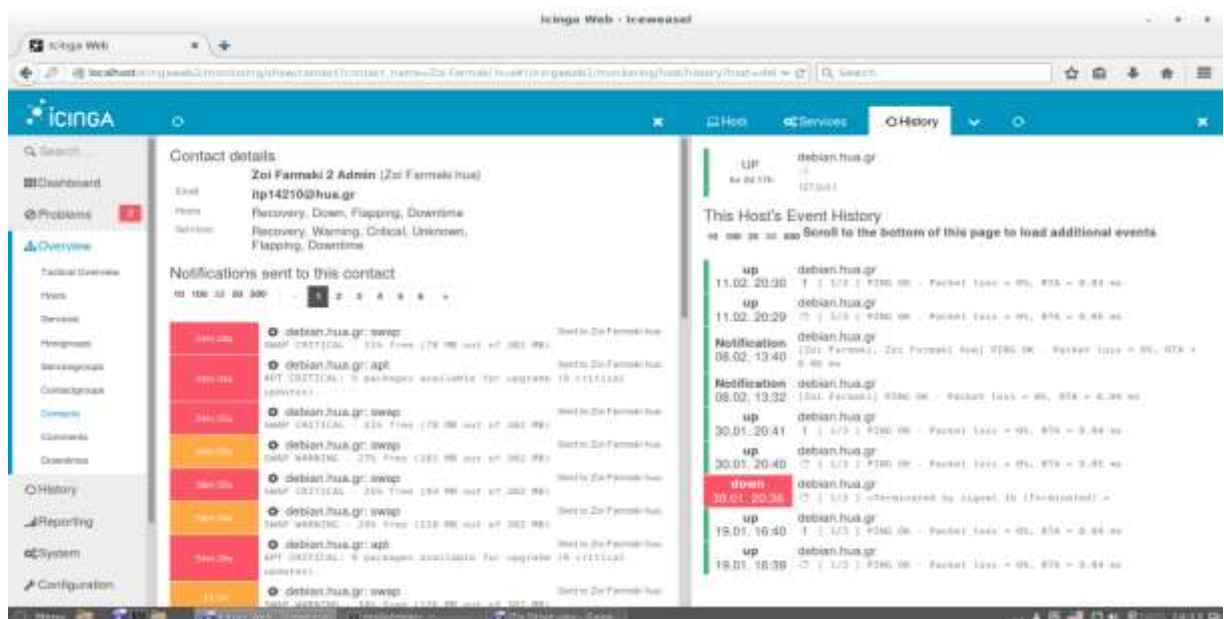
Εικόνα 17: users.conf (iii)

Στην ίδια καρτέλα *contacts* κάνοντας click στη ακριβώς διπλά καρτέλα στο services μας δείχνει συνοπτικά όλες τις υπηρεσίες που λειτουργούν η όχι και κάνοντας click σε μια οποιαδήποτε έχουμε προσβασιμότητα σε ότι ακριβώς αναφέρθηκε ακριβώς από πάνω.



Εικόνα 18: users.conf (iv)

Όπως δείχνει και ο τίτλος στην εικόνα που φαίνεται “This Host’s Event History” πρόκειται για ιστορική αναδρομή σχετικά με την κατάσταση τους συστήματος μας από την πρώτη μέρα υλοποίησης του έως την μέρα που το τρέχουμε.



Εικόνα 19: users.conf (v)

**Παράδειγμα μορφής αποστολής ειδοποίησης προς τον διαχειριστή:**



**Εικόνα 20: Παράδειγμα προβλήματος σε κάποιον εξυπηρετητή**

Εδώ διακρίνουμε πως στον server του πανεπιστημίου galaxy.hua η υπηρεσία “PING” βρίσκεται σε κατάσταση “CRITICAL” δηλαδή η υπηρεσία είναι σε κρίσιμη κατάσταση. Η ειδοποίηση που λάβαμε για αυτό το πρόβλημα καθώς και λίγα δευτερόλεπτα αργότερα για την ανάκτηση του προβλήματος είναι η εξής:



**Εικόνα 21: Εισερχόμενη ειδοποιητήρια αλληλογραφία**

Και εν συνεχεία οι πληροφορίες που μας δίνονται σχετικά με το πρόβλημα είναι της μορφής:



**Εικόνα 22: Τύπος ειδοποίησης προβλήματος**

Ενώ για την ανάκτηση του προβλήματος στην εξής μορφή:



### Εικόνα 23:Τυπος ειδοποίησης ανάκτησης του προβλήματος

Στο σημείο αυτό θα ήθελα να αναφέρω πως το παράδειγμα αυτό γίνεται κατανοητό ότι αναφέρθηκε την χρονική στιγμή που γράφτηκε αυτή η διπλωματική εργασία. Κατά τη διάρκεια ροής της λειτουργίας αυτού του συστήματος αντιλαμβανόμαστε ότι υπάρχουν διαφορετικές ενημερώσεις με διαφορετικά η ακόμη και ίδιου τύπου προβλήματος. Απλώς μπορεί να είναι για άλλο μηχάνημα για κάποια άλλη υπηρεσία η ακόμη και την ίδια.

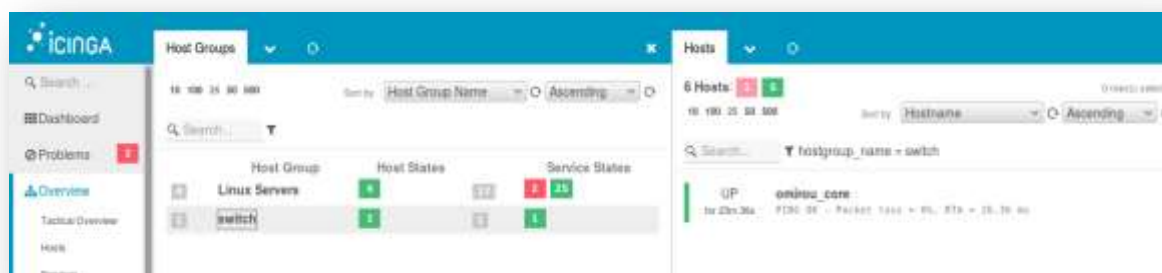
#### ***groups.conf***

Το configuration του συγκεκριμένου αρχείου αναλύεται στην σελίδα 64 και αυτό που μας παρέχει ως προς το περιβάλλον μας είναι να έχουμε οπτική ομαδοποίηση ως προς τα services μας και τα hosts μας έτσι ώστε να είναι πιο εύκολη η παρακολούθησής τους. Τι σημαίνει αυτό:

Έχουμε δημιουργήσει δυο Host Group το ένα ονομάζεται Linux Server και περιέχει όλους τους servers και το άλλο ονομάζεται switch και περιέχει όλα τα switches. Εικόνα 24-25

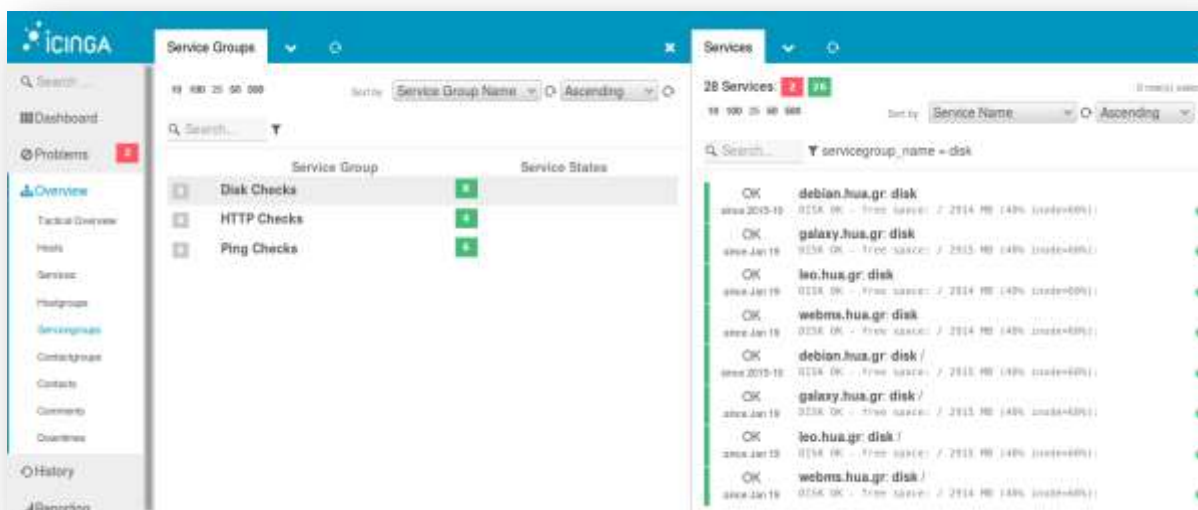


**Εικόνα 24: Ομαδοποίηση των servers**



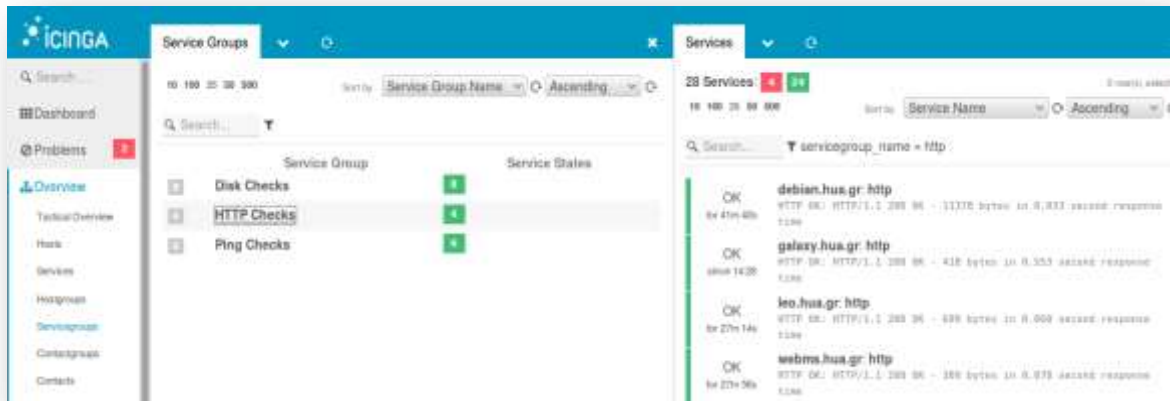
**Εικόνα 25: Ομαδοποίηση των switch**

Δημιουργήσαμε Service Group για κάθε μια υποστηριζόμενη υπηρεσία για κάθε Server. Απεικονίζονται ξεχωριστά κάθε μια στις τρεις εικόνες που ακολουθούν.

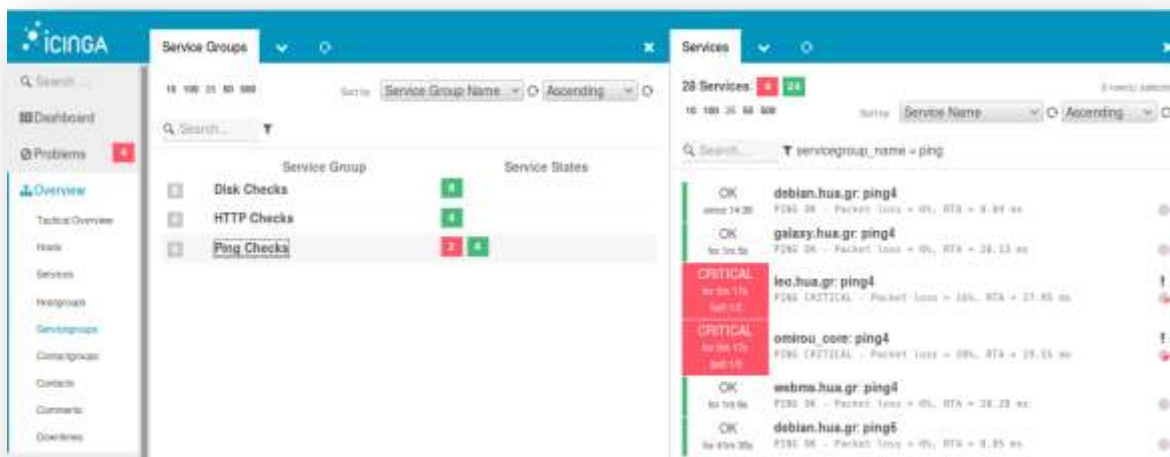


**Εικόνα 26: Ομαδοποιημένος έλεγχος μόνο για Dick Checks**





Εικόνα 27: Ομαδοποιημένος έλεγχος μόνο για HTTP Checks



Εικόνα 28: Ομαδοποιημένος έλεγχος μόνο για Ping Checks

### *templates.conf*

Το configuration του αρχείου αυτού βρίσκεται στην σελίδα 65. Όπως έχει αναφερθεί στο κεφάλαιο 4.4 ουσιαστικά μέσα από αυτό το αρχείο μας δίνεται η δυνατότητα αλλαγής χρόνου αποστολής ειδοποιήσεων στο προσωπικό μας email.

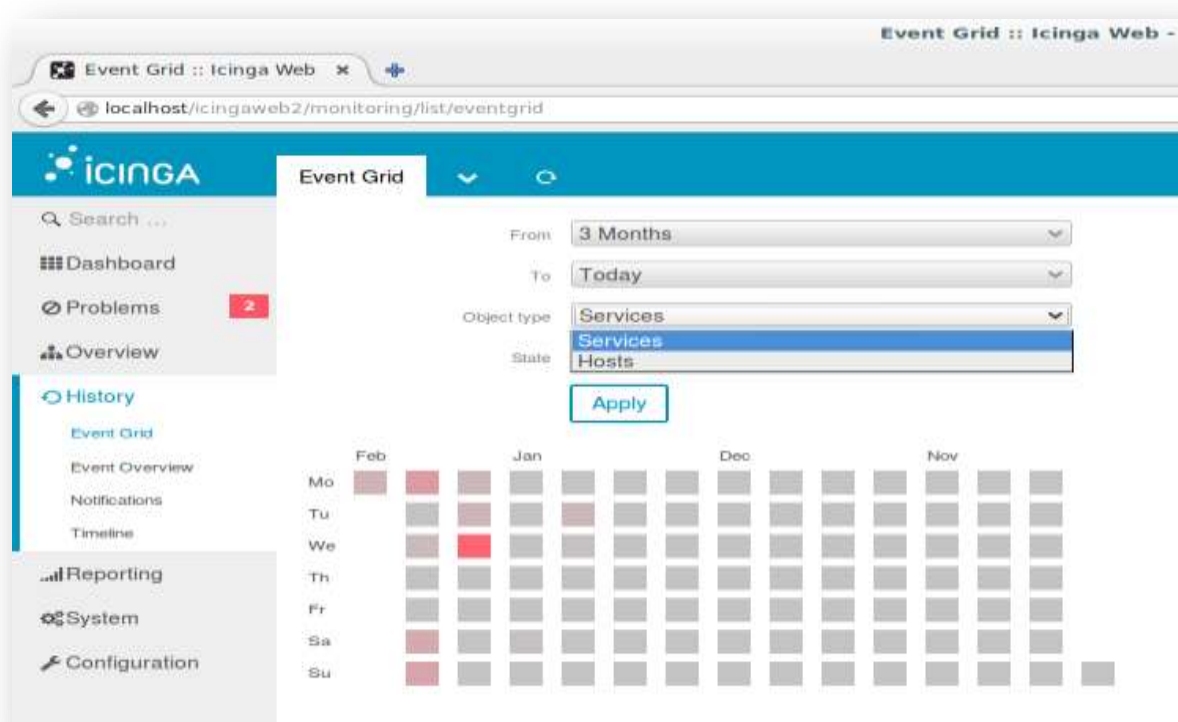
Τα υπόλοιπα αρχεία που απομείνανε και στα οποία δεν έγινε αναφορά με παραδείγματα είναι γιατί δεν επιφέρουν ως προς το δίκτυο διαχείρισης μας/ το περιβάλλον μας κάποιες οπτικές αλλαγές παρά μόνο λειτουργικές. Συνοπτικά, τα αρχεία αυτά είναι notifications.conf, downtimes.conf, timeperiods.conf, satellite.conf και apt.conf.

## 5.3 Εύχρηστες υπηρεσίες πληροφόρησης μέσα από το web interface

Εκτός από τις υπηρεσίες που αναφέρθηκαν στο προηγούμενο κεφάλαιο, οι οποίες διαμορφώνονται μέσα από τον κατάλογο `conf.d` υπάρχουν και κάποιες άλλες υπηρεσίες οι οποίες δεν διαμορφώνονται μέσα από αυτόν τον κατάλογο αλλά μας παρέχουν πολλές σημαντικές και εύχρηστες πληροφορίες.

### 5.3.1 Η καρτέλα History

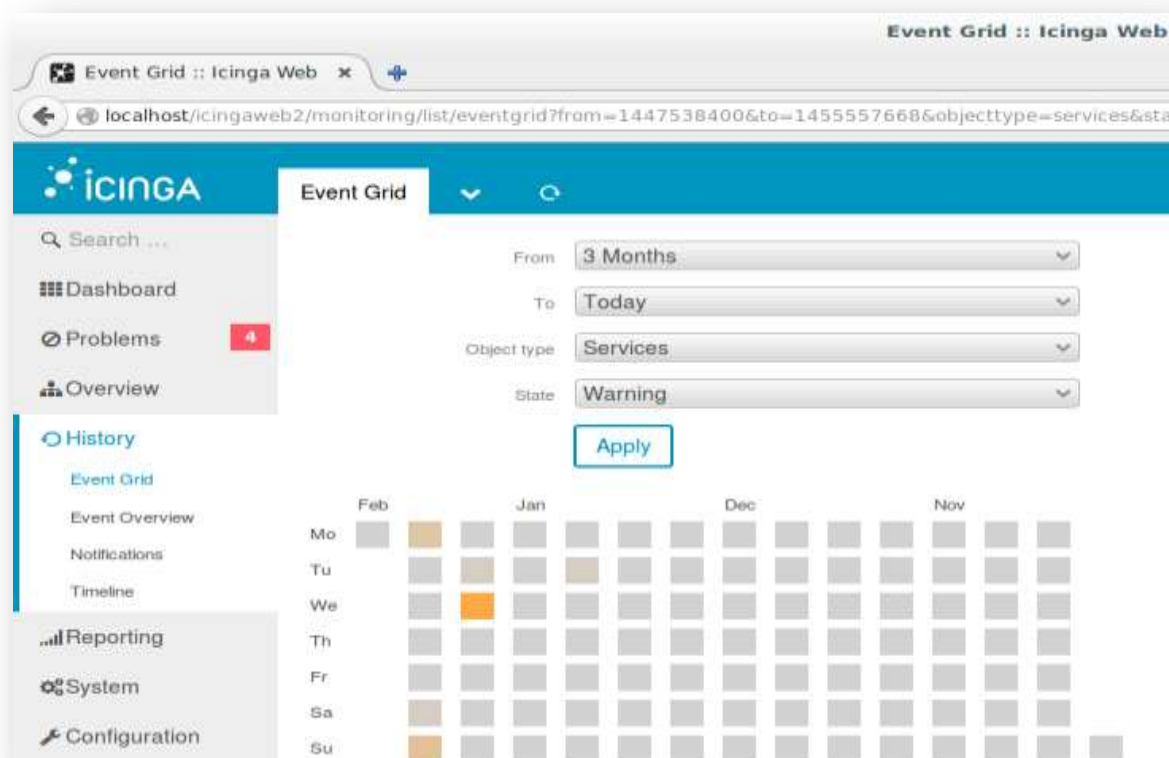
#### *Event Grid*



**Εικόνα 29: History/ Event Grid**

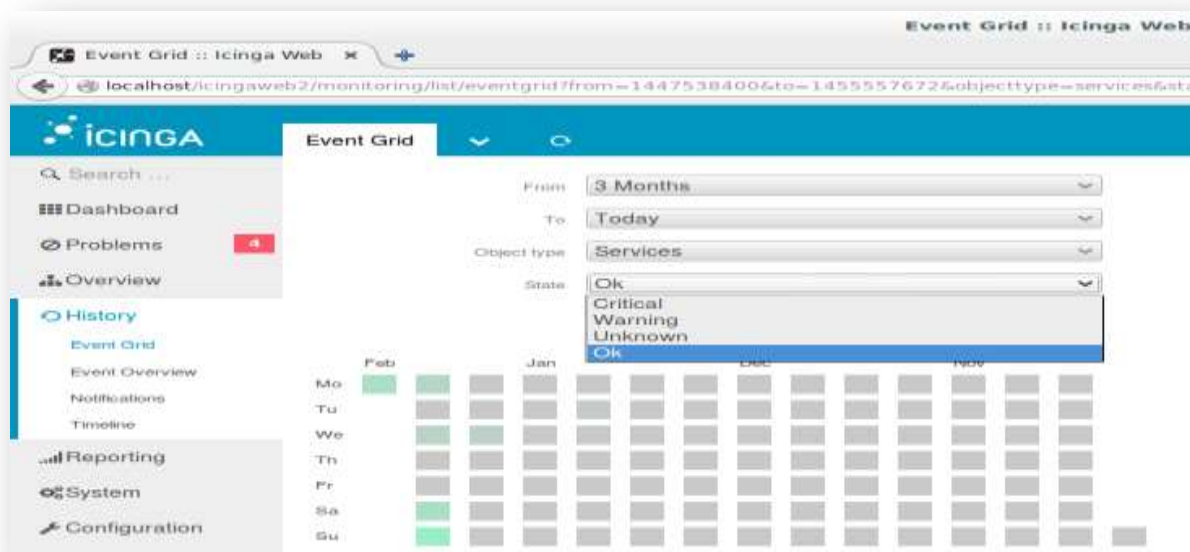
Στο συγκεκριμένη καρτέλα έχουμε την δυνατότητα να δούμε μέσα από το πλέγμα που απεικονίζεται τις υπηρεσίες μας *Services/Hosts* και σε τι κατάσταση βρίσκονται. Επίσης μπορούμε να ορίσουμε από πότε μέχρι πότε θέλουμε να ελέγξουμε κάτι τέτοιο και σε ποια κατάσταση. Στο συγκεκριμένο παράδειγμα ελέγχουμε για *Services* και επιλεγούμε να ελέγξουμε πότε ήταν σε κατάσταση κρίσιμη/ *Critical* και αυτό φαίνεται με κόκκινο χρώμα.





Εικόνα 30: History/ Event Grid (ii)

Ότι ίσχυε και προηγουμένως με την διαφορά ότι εδώ επιλέξαμε κατάσταση *Warning* και η οποία φαίνεται στο πλέγμα μας με κίτρινο χρώμα.



Εικόνα 31: History/ Event Grid (iii)

Το ίδιο και εδώ με την διαφορά ότι εδώ επιλέξαμε κατάσταση *Ok* η οποία φαίνεται στο πλέγμα μας με πράσινο χρώμα.

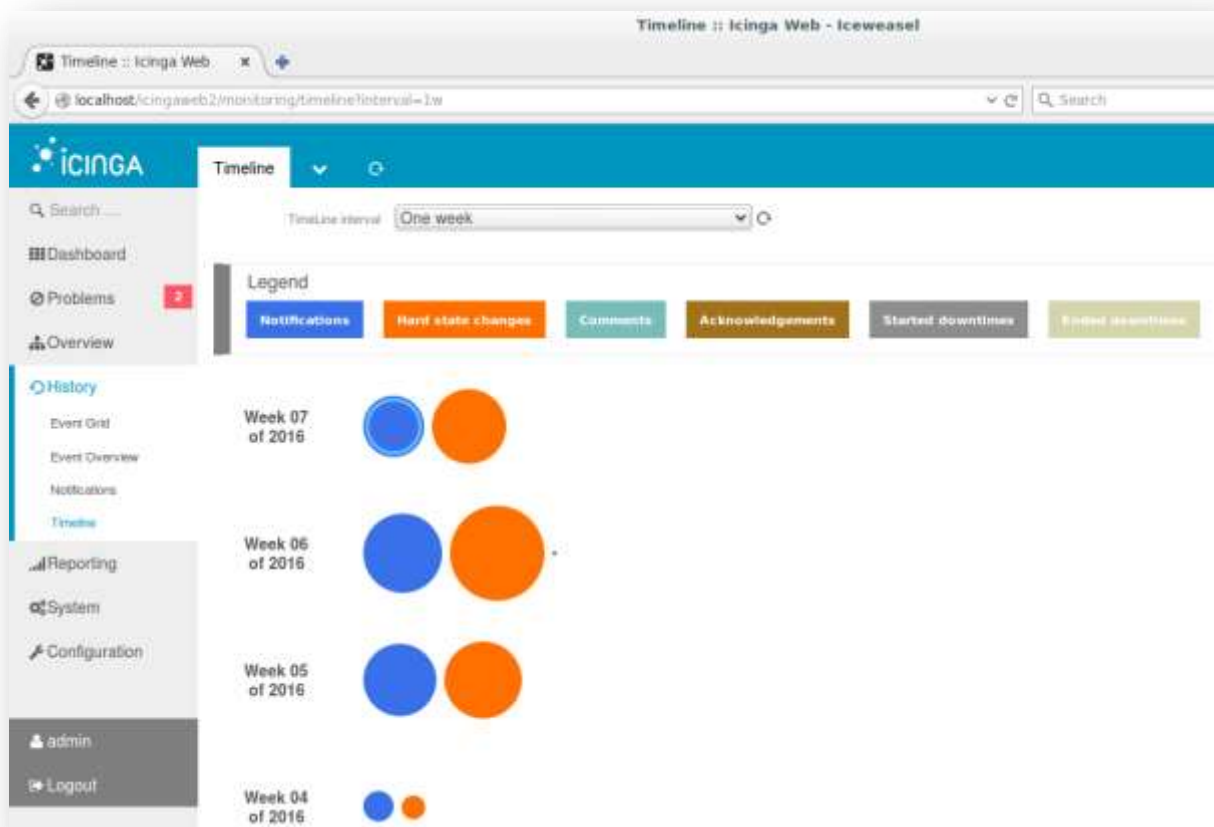
### ***Event Overview***

Είναι ακριβώς όπως την καρτέλα Overview κεφάλαιο 5.2 με την διαφοροποίηση ότι εκεί τις ενημερώσεις τις βγάζει για κάθε υπηρεσία ξεχωριστά και ομαδοποιημένες ενώ εδώ ανά χρονικά και ανάμικτα.

### ***Notifications***

Πάλι όπως στο κεφάλαιο 5.2 όπου αναφέρθηκε, μας ενημερώνει για τα μηχανήματα μας ως προς τις υπηρεσίες τους και σχετικά με τις ειδοποιήσεις που στάλθηκαν σε ποιον διαχειριστή, για το τύπο προβλήματος, την επίλυση του η όχι και τον αριθμό ειδοποιήσεων.

### ***Timeline***



**Εικόνα 32: History/ Timeline**

Στο συγκεκριμένο πεδίο είναι η συνοπτική εικόνα αναφορικά με τον διαχειριστή του συγκεκριμένου συστήματος δικτύου και το ίδιο το σύστημα. Για παράδειγμα, η εικόνα 49 μας δείχνει πως για την έβδομη εβδομάδα του 2016, δηλαδή την τρίτη εβδομάδα του Φεβρουαρίου τι όγκος ειδοποιήσεων υπήρχαν σχετικά με απώλειες η ανακτήσεις που υπήρχαν την χρονική διάρκεια εκείνη. Ωστόσο, μας δίνεται η ευκολία να κάνουμε click σε ένα κύκλο από τους υπάρχων και έπειτα ανοίγει δεξιά ένα νέο frame που δείχνει που σταλθήκαν ειδοποιήσεις και για ποιον λόγο, αν κάνουμε click στο πορτοκαλί κύκλο βλέπουμε την κατάσταση των υπηρεσιών μας εκείνη την χρονική στιγμή που επιλέξαμε. Επίσης, εδώ διακρίνεται κ ένας μικρός κύκλος στην έκτη εβδομάδα που αν κάνουμε click πάνω του μας δείχνει για ποια υπηρεσία είχαμε κάνει σχόλιο και τι ακριβώς σημειώσαμε.

### 5.3.2 Η καρτέλα Reporting



**Εικόνα 33: Reporting - Alert Summary**

Μπορούμε να δούμε την πορεία του συστήματος. Υπάρχει δυνατότητα απεικόνισης σε γραφική παράσταση από μια εβδομάδα, έναν μήνα η ένα χρόνο. Στο πρώτο διάγραμμα στην κάθετη στήλη διακρίνουμε το πλήθος των ειδοποιήσεων που στάλθηκαν με μπλε χρώμα και στην οριζόντια την ημερομηνία αυτού με κόκκινο χρώμα, ατέλειες που υπήρχανε ανά ημέρα.

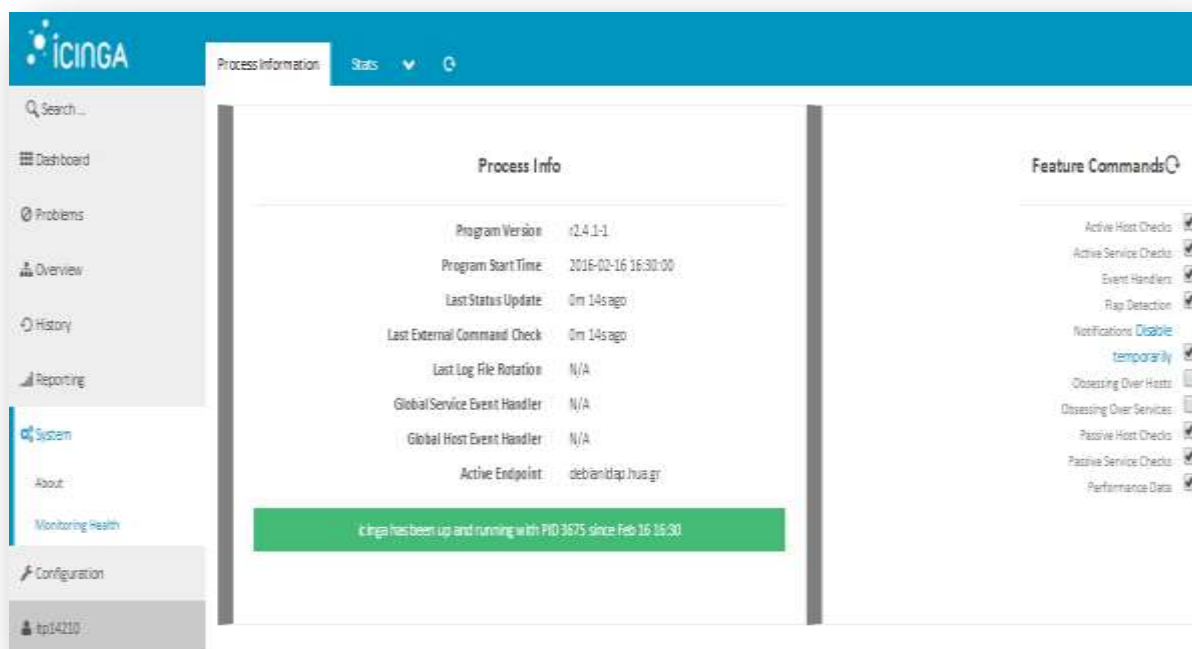
Στο δεύτερο σχήμα μπορούμε να δούμε το μέσο χρόνο ειδοποιήσεων που στάλθηκαν οι ειδοποιήσεις. Στο συγκεκριμένο παράδειγμα μας πληροφορεί το σύστημα μας ότι υπολογίζοντας 14,84 ειδοποιήσεις ανά ώρα οι 8 είναι την τελευταία ώρα ενώ για το τελευταίο 24 όρο πάνω από 19,64% ο μέσος Όρος.

### 5.3.3 Η καρτέλα System

#### About

Γενικότερες πληροφορίες ως προς το σύστημα την έκδοση την οποία λειτουργεί , δυνατότητα πρόσβασης σε chat με την κοινότητα του icinga web 2 σχετικά με απορίες η οτιδήποτε άλλο

#### Monitoring Health



Εικόνα 34: System/ Monitoring Health

Πληροφορίες σχετικά με το σύστημα ως προς την λειτουργία από την ημερομηνία εγκατάστασης, τότε έγινε η τελευταία ενημέρωση και γενικότερα μας εμφανίζει στην μπάρα κάτω που φαίνεται με πράσινο χρώμα αν όλα λειτουργούν σωστά η όχι. Αν υπάρχει κάποιο πρόβλημα στο σύστημα σχετικά με κάποια λειτουργία μας την εμφανίζει σε εκείνο το σημείο με κόκκινο χρώμα και αναφέρει το πρόβλημα.

## 5.3.4 Η καρτέλα Configuration

### Application

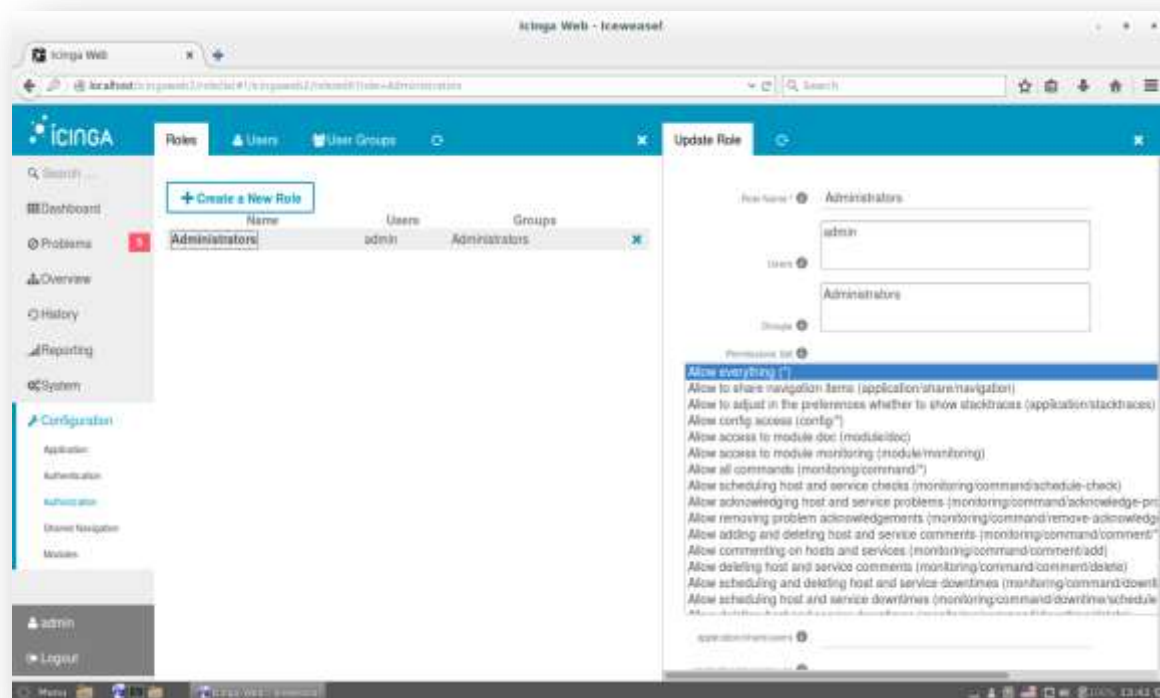
Μας ενημερώνει σχετικά με την διαδικασία διαμόρφωσης που ακολουθήσαμε στην εγκατάσταση και μας δίνει το δικαίωμα να τα τροποποιήσουμε αν θέλουμε και εφόσον χρειάζεται. Μπορούμε να ανατρέξουμε στο παράρτημα Α, στις εικόνες 13, 16 και 21 να θυμηθούμε τι ακριβώς εννοούμε.

### Authentication

Επαλήθευση των στοιχείων που πραγματοποιήθηκαν κατά την εγκατάσταση του συστήματος αυτό με δυνατότητα της αλλαγή τους αν και εφόσον χρειάζεται. Στο παράρτημα Α, εικόνα 14 μπορούμε να θυμηθούμε ακριβώς ποια διαμόρφωση επιλέξαμε.

### Authorization

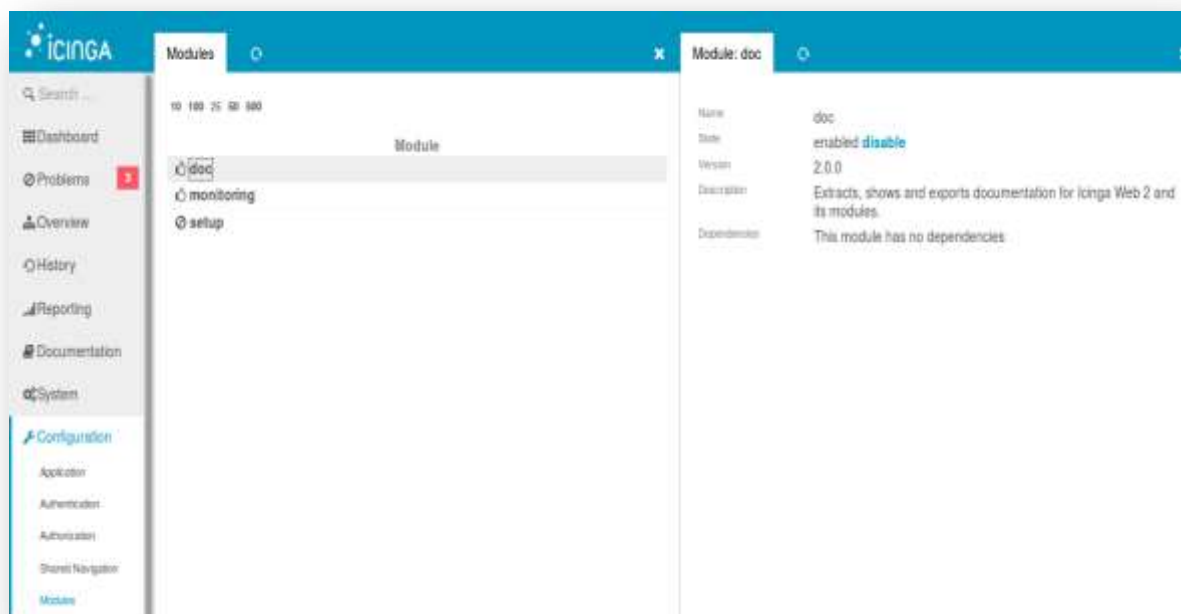
Εδώ μπορούμε να δούμε ποιος είναι ο διαχειριστής του συστήματος, να προσθέσουμε κάποιον άλλον και επίσης να ορίσουμε τα δικαιώματα του ως προς το σύστημα διαχείρισης αυτό. Στην εικόνα βλέπουμε πως υλοποιείται κάτι τέτοιο πρακτικά:



Εικόνα 35: Configuration/ Authorization

## Modules

Η υπηρεσία αυτή μας δίνει την δυνατότητα να δούμε ποια “modules” έχουμε εγκαταστήσει για το σύστημα μας, μας επιτρέπει να δούμε πληροφορίες σχετικά με αυτά ωστόσο και να επιλέξουμε ποια από αυτά θέλουμε να τα βλέπουμε στο κύριο menu του προγράμματος η να τα βλέπουμε μόνο όταν επιθυμούμε. Για παράδειγμα, για να δούμε τι ακριβώς εννοούμε ως ανατρέξουμε στην εικόνα 29 στο κεφάλαιο 5.1. Στο σημείο εκείνο βλέπουμε πως στο αριστερό frame δεν υπάρχει η υπηρεσία υποστηρίξεις σχετικά με το “documentation”. Ας δούμε την κάτω εικόνα να δούμε την διαφορά που μας επιτρέπει να πραγματοποιήσουμε το πεδίο των Modules.



**Εικόνα 36: Configuration/ Modules**

Διακρίνουμε λοιπόν πως η υπηρεσία του doc είναι ενεργοποιημένη. Προηγουμένως είχε το εικονίδιο που απεικονίζεται στην υπηρεσίας setup. Κάνοντας click στο enable στο δεξί frame εκεί που μας δίνονται πληροφορίες σχετικά με τι υπηρεσία πρόκειται ενεργοποιείται η υπηρεσία κ πλέον την βλέπουμε και στο αριστερό frame ότι έχει προστεθεί κάτω από την καρτέλα “Reporting” πλέον εμφανίζεται και το “documentation” του icinga web2.

## 6. Συμπεράσματα

Η δημιουργία ενός λογισμικού ανοιχτού κώδικα για την τεχνική υποστήριξη στο Χαροκόπειο Πανεπιστήμιο ήταν πλέον αναγκαία για την παρακολούθηση των συστημάτων του Πανεπιστημίου απομακρυσμένα. Η δημιουργία ενός τέτοιου συστήματος διαχείρισης δικτύου από την αρχή θα ήταν πολύ χρονοβόρα και σίγουρα όχι αναγκαία διότι στις μέρες μας η ποικιλία εφαρμογών που διατίθενται στο διαδίκτυο είναι αρκετές και καλύπτουν πλήρως τις απαιτήσεις μας. Έτσι, επιλέχθηκε το κατάλληλο λογισμικό το οποίο ήταν πιο κοντά στις ανάγκες του Πανεπιστημίου και με τις κατάλληλες παραμετροποιήσεις για τις απαιτήσεις μας ήταν η βέλτιστη λύση.

Επιλέγοντας ένα λογισμικό που είναι “έτοιμο” (icinga web 2) υπάρχει το πλεονέκτημα αναβαθμίσεων και προσθήκη νέων δυνατοτήτων. Σε περίπτωση δημιουργίας μιας εφαρμογής από την αρχή με την επίτευξη πολύ καλών χαρακτηριστικών και απόδοσης, θα πρέπει να γινόταν συνεχώς προσθήκες και αναβαθμίσεις από κάποιον ειδικό (φυσικό πρόσωπο). Στην περίπτωση όμως την δική μας του “έτοιμου” λογισμικού δεν απαιτεί πολύ χρόνο και κάποια τεχνογνωσία συντήρησης και αναβάθμισης διότι γίνεται από άλλον φορέα και όχι από εμάς.

Λαμβάνοντας υπόψη ότι στις μέρες μας οι εφαρμογές ανοιχτού κώδικα έχουν φανεί αξιόπιστες και ανταγωνίζονται σε μεγάλο βαθμό, μπορεί και σε ίδιο, τις εφαρμογές που απαιτείται καταβολή χρηματικού ποσού για την απόκτηση τους η επιλογή ανοιχτού κώδικα λογισμικού ήταν η καλύτερη δυνατή. Συνεπώς, υπάρχει εξοικονόμηση χρημάτων για το Πανεπιστήμιο και όσον αναφορά την απόδοση είναι ανάλογη αν όχι ίδια με κάποιο εμπορικό εργαλείο.

Η διαδικασία εγκατάστασης της εφαρμογής ήταν σχετικά εύκολη, αφού υπάρχει καθοδήγηση βήμα προς βήμα από το επίσημο εγχειρίδιο χρήσης αλλά και ποικίλες διαδικτυακές κοινότητες στις οποίες οι χρήστες παραθέτουν απαντήσεις σε διάφορα προβλήματα που παρουσιάστηκαν κατά την διαδικασία εγκατάστασης. Κάτι αντίστοιχο υπάρχει και για τα επόμενα στάδια που είναι για τις παραμετροποιήσεις, σχετικές οδηγίες και καθοδηγήσεις.

Η συγκεκριμένη διπλωματική εργασία δημιούργησε τις κατάλληλες προϋποθέσεις για μια άρτια συνεργασία μεταξύ των διαχειριστών (στην παρούσα τον Δρ, Βασίλειο Δαλάκα μέλος του Κέντρου Πληροφορικής και Δικτύων). Το σύστημα διαχείρισης δικτύου αυτό έχει πολύ φιλικό και κατανοητό περιβάλλον και δίνει την δυνατότητα απομακρυσμένου ελέγχου των μηχανημάτων του Πανεπιστημίου τόσο της κατάστασης αυτών όσο και της ενημέρωσης με email στον προσωπικό λογαριασμό του διαχειριστή-ες την ίδια στιγμή της απώλειας. Κάτι

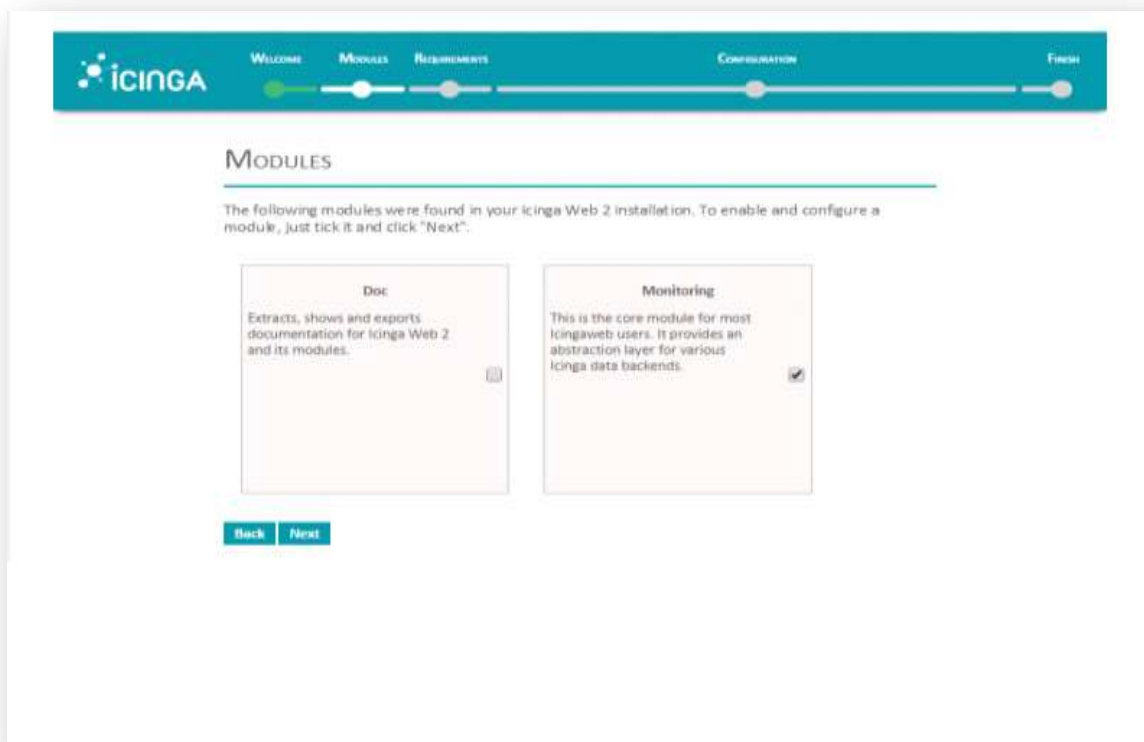
τέτοιο είναι πολύ εξυπηρετικό για την άμεση επίλυση του προβλήματος για την σωστή αποδοτικότητα του συστήματος.



## Παράρτημα Α

Στο σημείο αυτό φαίνεται αναλυτικά η διαδικασία εγκατάστασης η οποία αναφέρθηκε στο κεφάλαιο 3.2.1. Επιλέξαμε να ακολουθηθεί η διαδικασία στο παράρτημα αυτό διότι περιέχει όλα τα βήματα που πραγματοποιήθηκαν έτσι ώστε να φτάσουμε στην πλήρη εγκατάσταση, με εικόνες κάτι το οποίο θα γέμιζε το κύριος σκέλος της συγγραφής αυτής και δε το θέλαμε για την πιο εύχρηστη ευανάγνωστη.

Στο δεύτερο βήμα επιλέγουμε σε ποιο “Module” θέλουμε να συνεχίσουμε την εγκατάσταση. Επιλέξαμε το δεύτερο όπως φαίνεται διότι ήταν αυτό που θέλαμε “Monitoring” κάτι που ζητάνε και οι περισσότεροι χρήστες του icinga.



**Εικόνα 37: Επιλογή μοντέλου διαμόρφωσης**

Επιλέγουμε, με ποιον τρόπο θα μας αναγνωρίζει το icinga web 2 όταν προσπαθούμε να κάνουμε εισαγωγή.

Authentication

Please choose how you want to authenticate when accessing Icinga Web 2. Configuring backend specific details follows in a later step.

Authentication Type ? Database

Back Next

**Εικόνα 38: Τύπος αυθεντικοποίησης**

Ρυθμίζουμε έτσι ώστε οι προτιμήσεις του χρήστη να αποθηκεύονται στην ίδια βάση δεδομένων που επιλέξαμε και για τον τύπο αυθεντικοποίησης.

Preferences

- Please choose how Icinga Web 2 should store user preferences.
- Note that choosing "Database" causes Icinga Web 2 to use the same database as for authentication.

User Preference Storage Type Database

Back Next

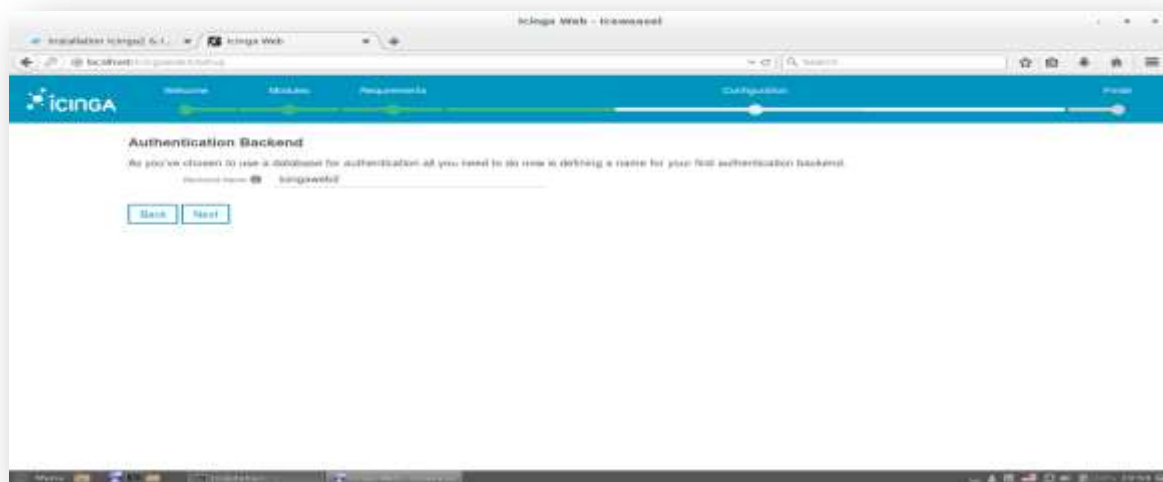
**Εικόνα 39: Προτίμηση αποθήκευσης στην database**

Έπειτα διαμορφώνουμε την βάση δεδομένων μας. Πρόκειται για την MySQL η οποία λειτουργεί στην πόρτα 3306 και ορίζουμε όνομα χρήστη και κωδικό πρόσβασης.

The screenshot shows the Icinga Web 2 installation wizard in a web browser. The browser tab is titled 'Icinga Web - Icinga2 & Icinga Web'. The address bar shows 'localhost:icingaweb2/setup'. The wizard has a progress bar at the top with five steps: Welcome, Module, Requirements, Configuration, and Finish. The 'Configuration' step is currently active. The main heading is 'Database Resource'. Below it, there is a note: '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.' The form contains the following fields: 'Resource Name' (text input, value: 'icingaweb\_db'), 'Database Type' (dropdown menu, value: 'MySQL'), 'Host' (text input, value: 'localhost'), 'Port' (text input, value: '3306'), 'Database Name' (text input, value: 'icingaweb2'), 'Username' (text input, value: 'root'), 'Password' (password input, masked with dots), 'Character Set' (text input, empty), and 'Persistent' (checkbox, checked). At the bottom of the form are three buttons: 'Back', 'Next', and 'Validate Configuration'. A small note at the bottom of the form states: 'Required fields are marked with \* and must be filled in to complete the form.'

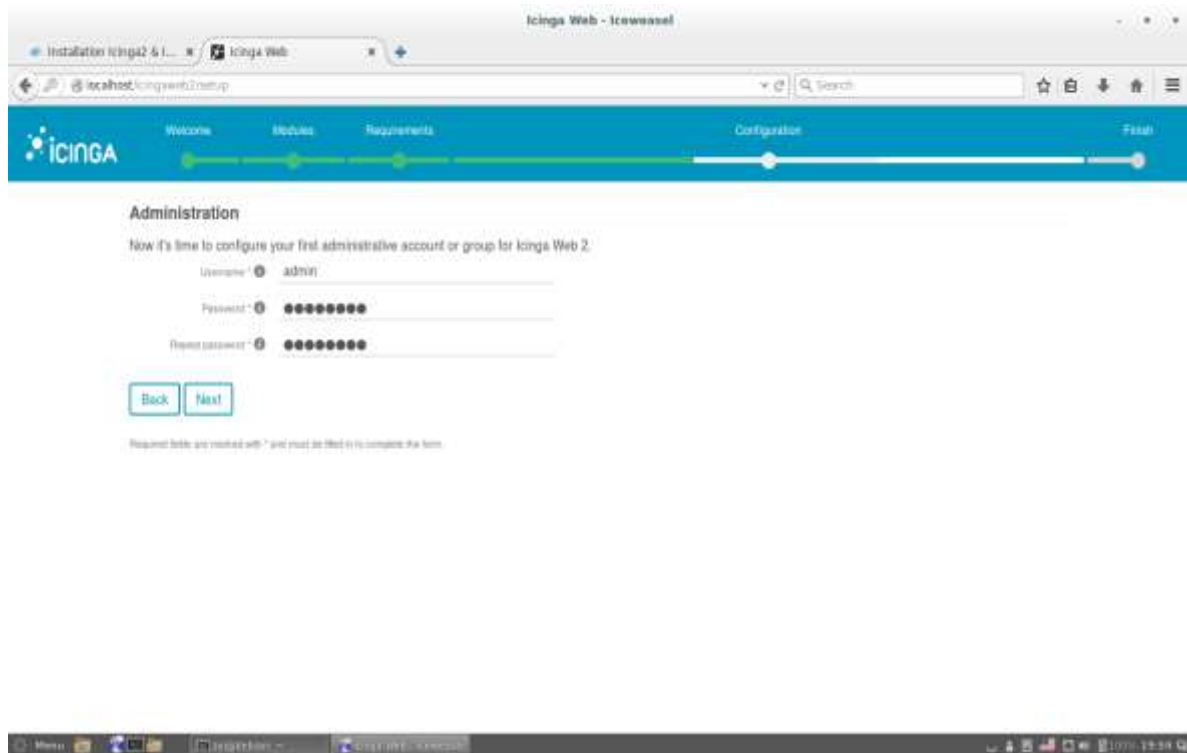
**Εικόνα 40: Διαμόρφωση Βάσης δεδομένων για τους χρήστες**

Επιβεβαιώνουμε το Backend name το οποίο εμφανίζεται, αν θέλουμε μπορούμε να το αλλάξουμε ωστόσο εμείς το αφήσαμε όπως δόθηκε και το οποίο είναι icingaweb2.



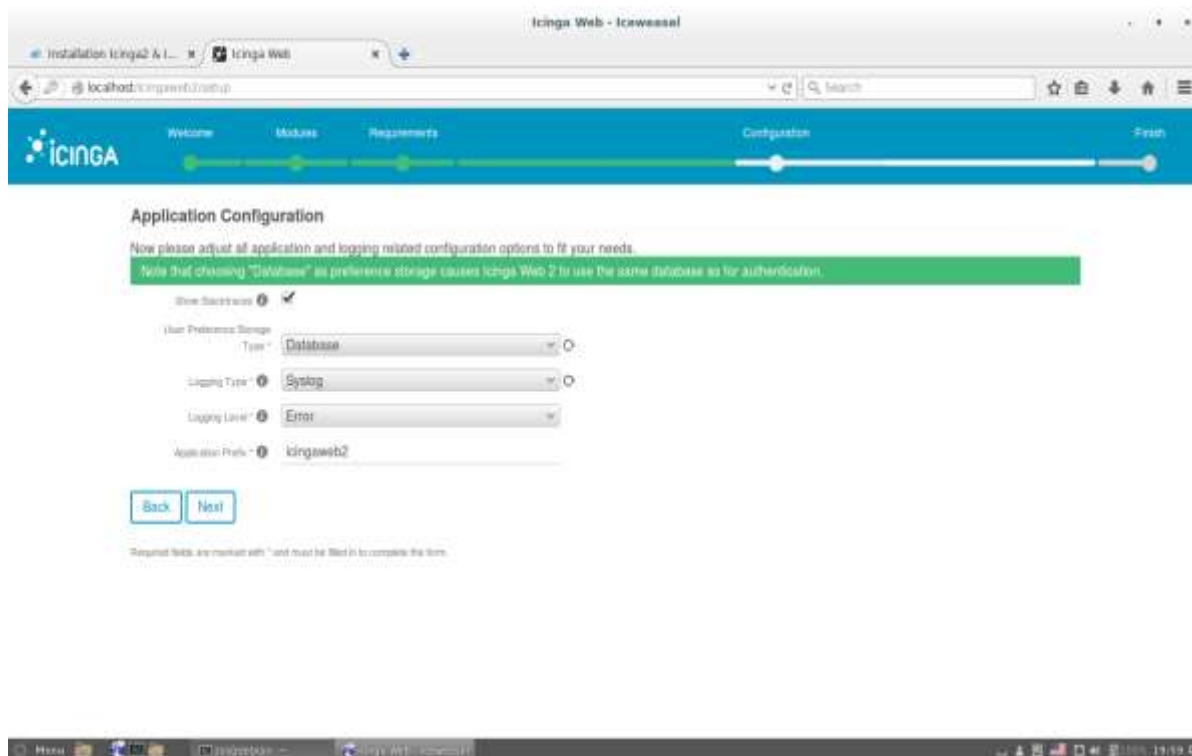
**Εικόνα 41: Βήμα Αυθεντικοποίησης/επιβεβαίωσης**

Ορίζουμε το username σε admin και δίνουμε και μυστικό κωδικό πρόσβασης.



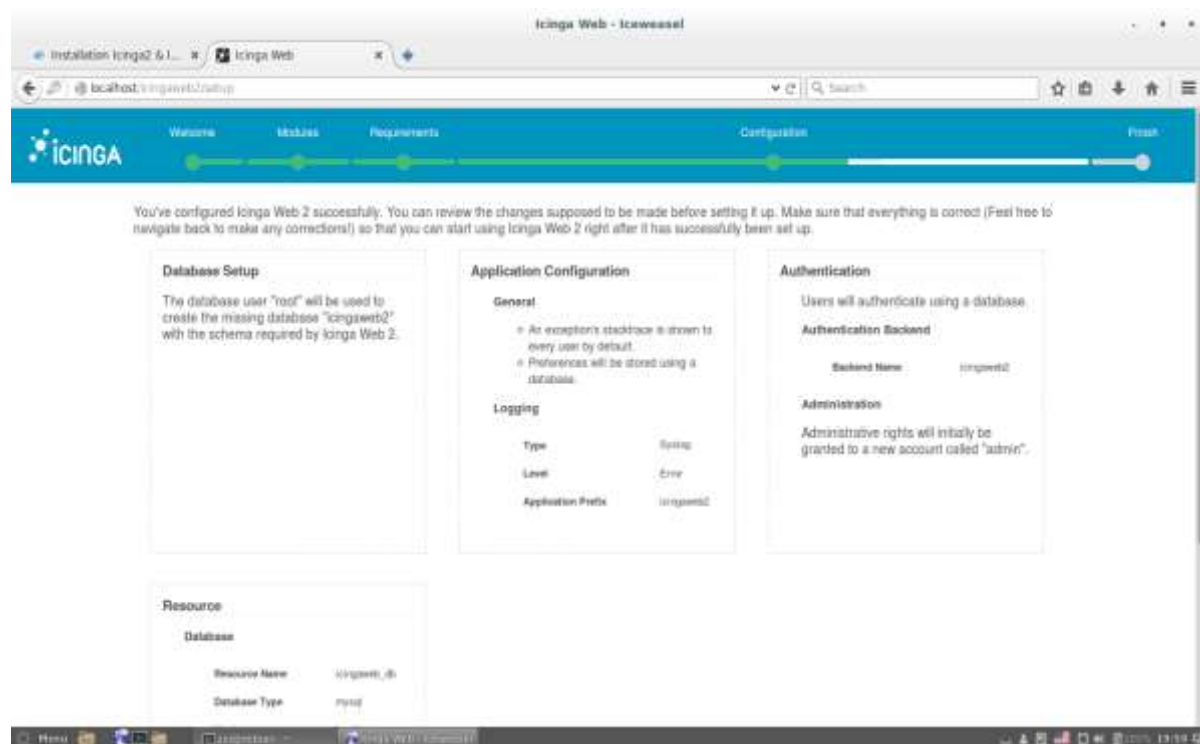
**Εικόνα 42: Δημιουργία του προσωπικού μου λογαριασμού**

Ρυθμίζουμε τους παραμέτρους εφαρμογής. Όπως διακρίνεται επιλέξαμε τύπο αποθήκευσης για τις προτιμήσεις του χρήστη σε βάση δεδομένων και αυτό γιατί έτσι ορίστηκε εξ' αρχής.

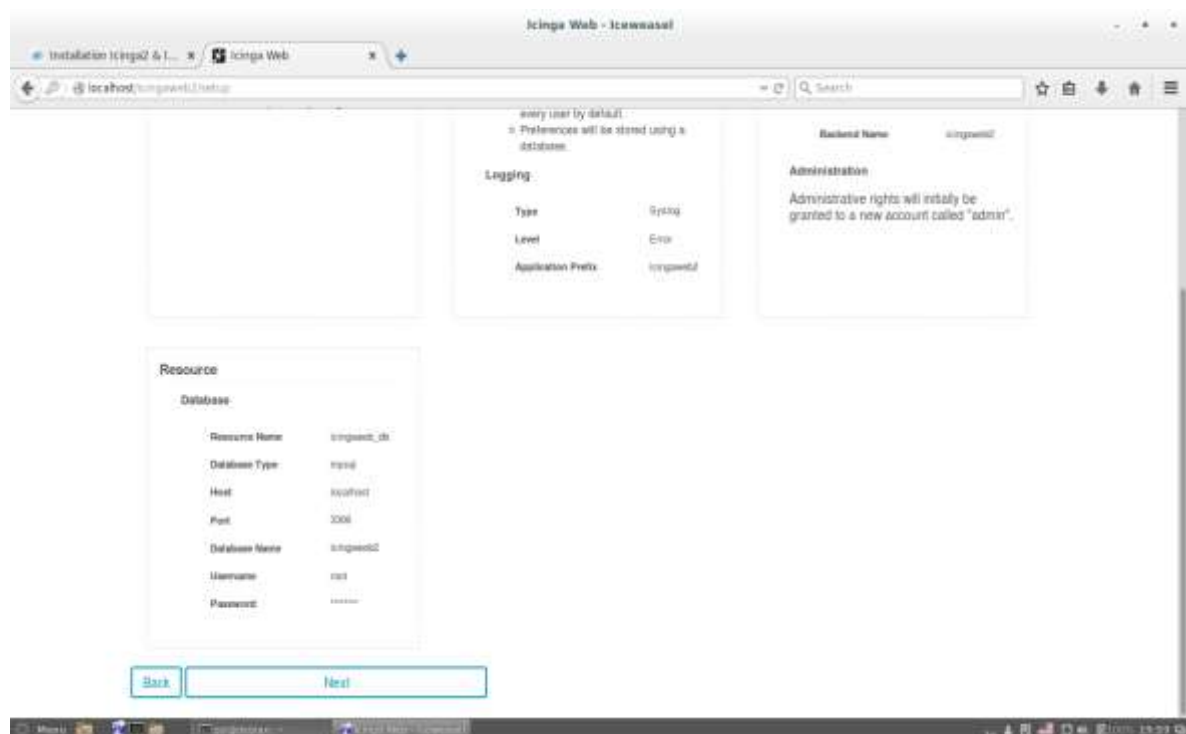


**Εικόνα 43: Ρυθμίσεις παραμέτρων εφαρμογής**

Έπειτα μετά από αυτές τις ρυθμίσεις μας εμφανίζεις στις εικόνες 17-18 που ακολουθούνε τι συνολικά έχουμε επιλέξει για την διαμόρφωση του συστήματος αυτού και μετά επιλέγουμε next για να συνεχίσουμε στις επόμενες ρυθμίσεις.

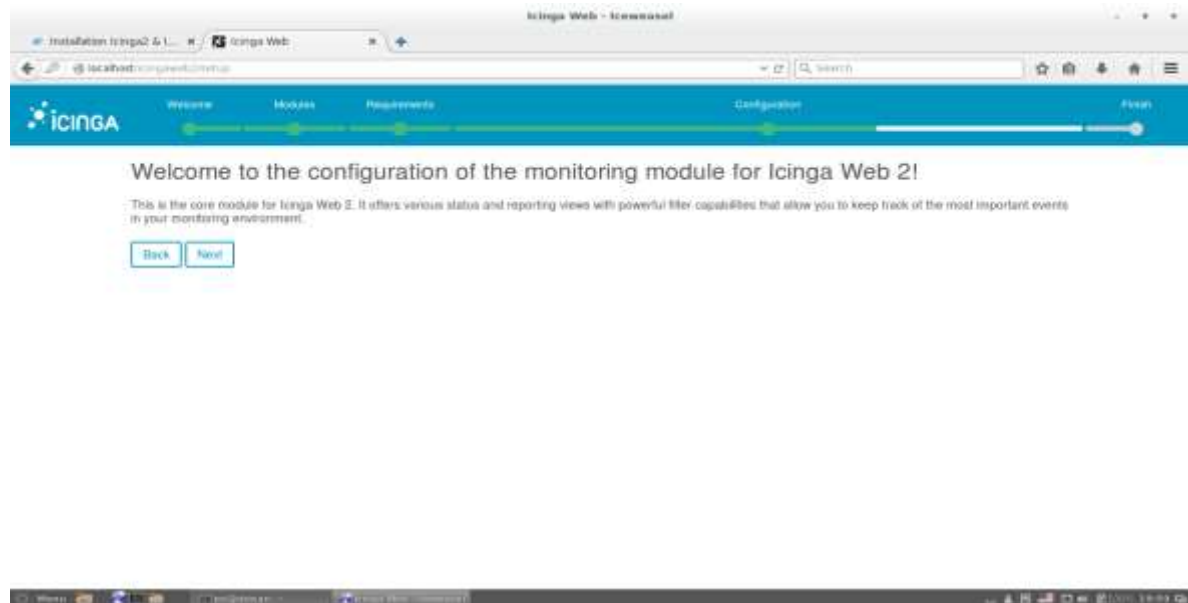


Εικόνα 44: Επαλήθευση στοιχείων διαμόρφωσης\_a



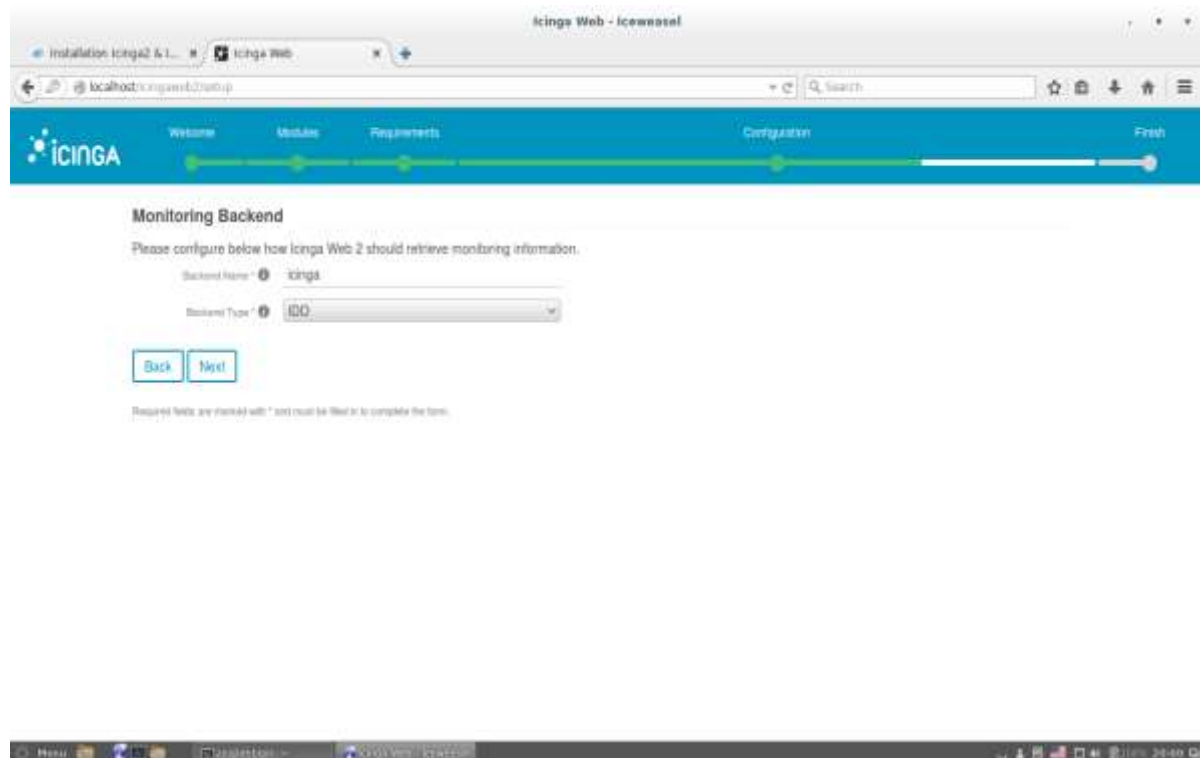
Εικόνα 45: Επαλήθευση στοιχείων διαμόρφωσης\_b

Συνεχίζουμε με τις ρυθμίσεις που χρειάζονται για την παρακολούθηση μέσω του icinga web 2



**Εικόνα 46: Ολοκλήρωση επιτυχής διαδικασίας διαμόρφωσης**

Η πρώτη μας επιλογή ως προς τις ρυθμίσεις παρακολούθησης είναι πώς το icinga web2 θα λαμβάνει πληροφορίες και επιλέγουμε IDO MySQL



**Εικόνα 47: Επιλογή βάσης δεδομένων για ανάκτηση πληροφοριών παρακολούθησης**

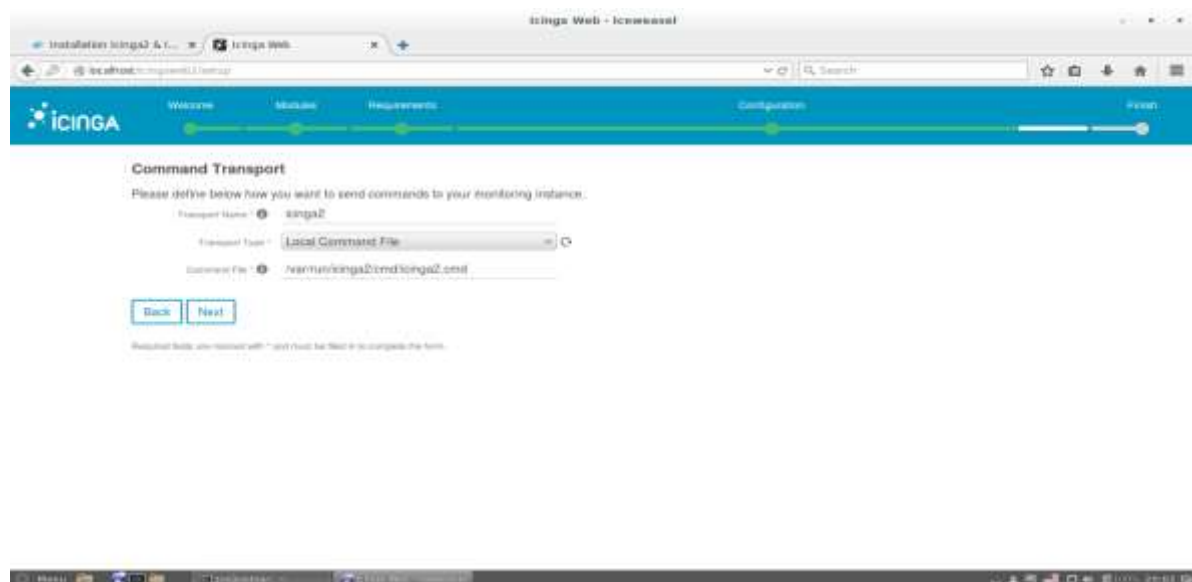
Συμπληρώστε τα παρακάτω στοιχεία για να αποκτήσουμε πρόσβαση στη βάση δεδομένων Ido του περιβάλλοντος παρακολούθησης μας

The screenshot shows the Icinga Web 2 configuration interface in a web browser. The browser's address bar shows 'localhost:5666/icinga2/setup'. The interface has a blue header with the Icinga logo and navigation tabs: Overview, Modules, Requirements, Configuration (active), and Fresh. Below the header, a progress bar indicates the current step in the configuration process. The main content area is titled 'Monitoring IDO Resource' and contains a form for configuring the IDO database connection. The form includes fields for Resource Name (icinga\_ido), Database Type (MySQL), Host (localhost), Port (3306), Database Name (icinga2idomysql), Username (icinga2-ido-mysql), Password (masked with dots), Character Set, and Persistent. At the bottom of the form are buttons for 'Back', 'Next', and 'Validate Configuration'. A note at the bottom states: 'Required fields are marked with \* and must be filled in to complete the form.'

**Εικόνα 48:** Συμπλήρωση στοιχείων διαδικασίας σύνδεσης IDO-mysql με το icinga web 2

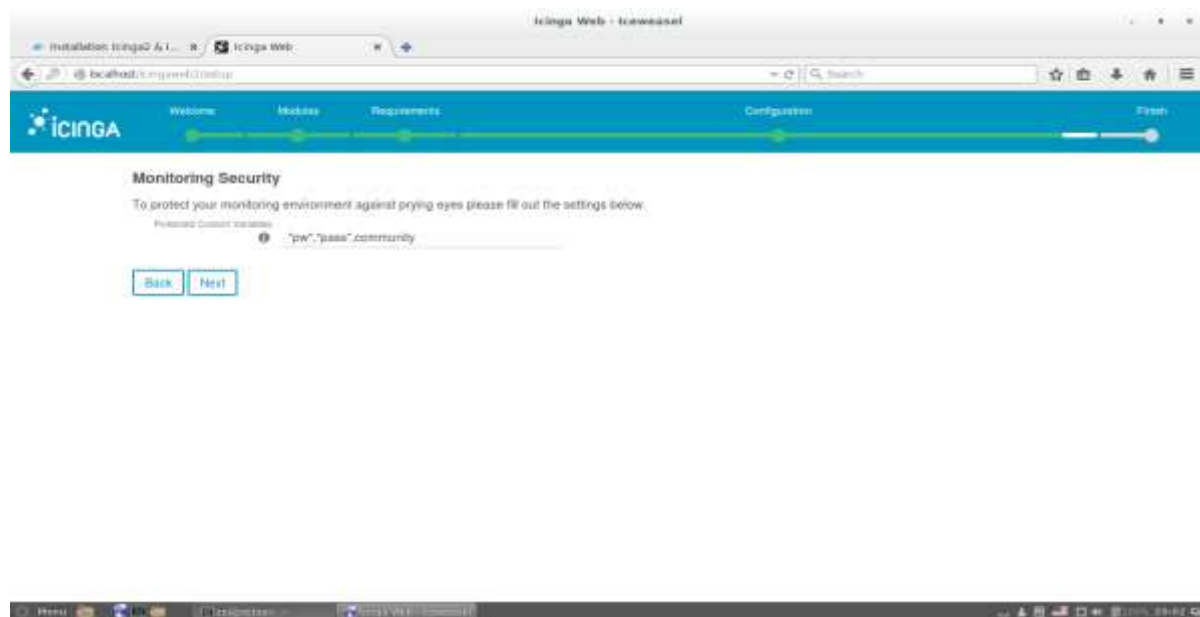
Ορίζουμε πως θέλουμε να στέλνουμε εντολές για υπόδειξη παρακολούθησης





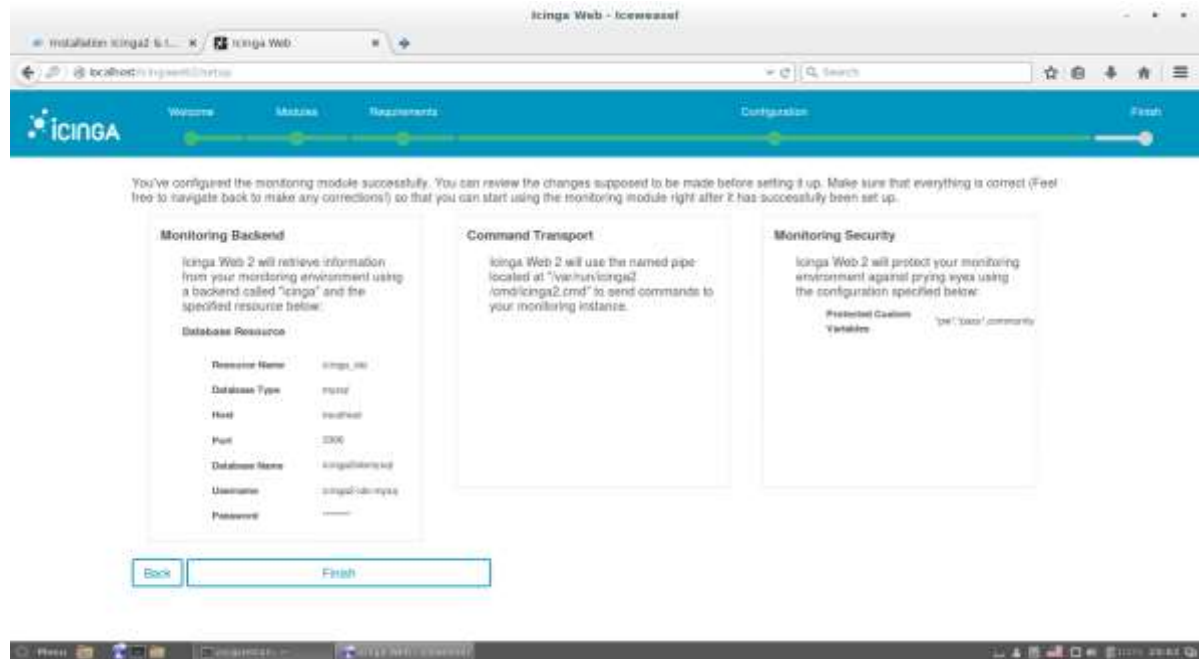
**Εικόνα 49: Τρόπος μεταφοράς εντολών**

Για την προστασία του περιβάλλοντος παρακολούθησή μας, μας καθοδηγεί να ακολουθήσουμε έχοντας αυτές τις εντολές



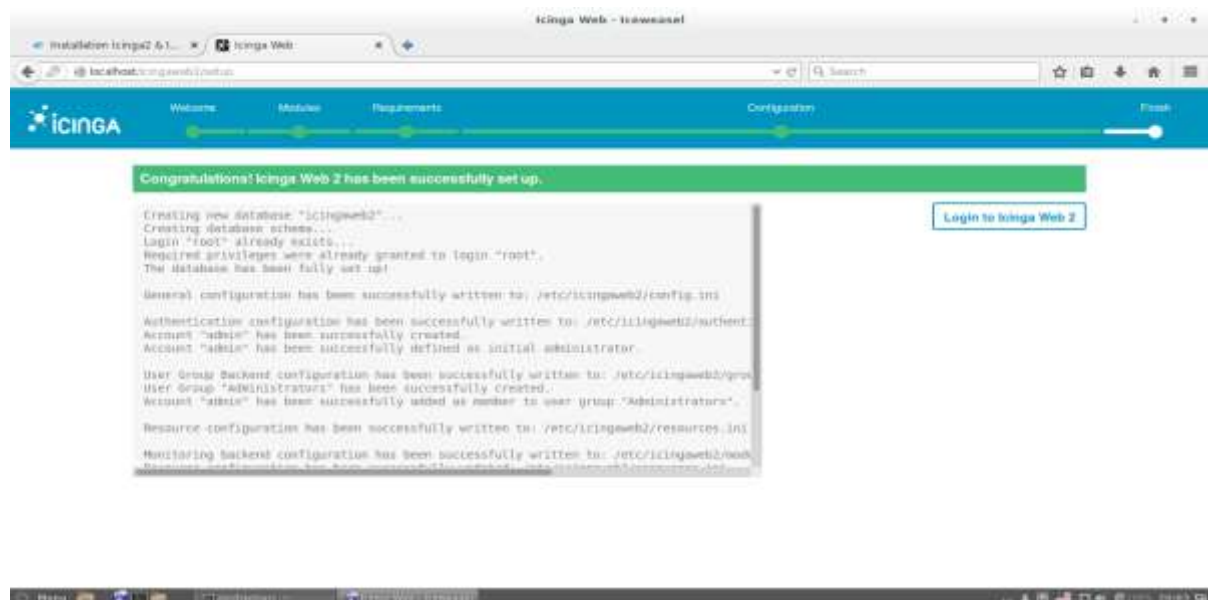
**Εικόνα 50: Προστασία του δικτύου διαχείρισης**

Πριν την ολοκλήρωση της εγκατάστασης γίνεται συνοπτικός έλεγχος στο τι έχουμε επιλέξει



**Εικόνα 51: Ανακεφαλαιοποίηση όσων προηγήθηκαν και επιβεβαίωση**

Και έπειτα έχουμε ολοκληρώσει την διαδικασία και είμαστε έτοιμοι να κάνουμε Login.



**Εικόνα 52: Επιτυχής εγκατάσταση του icinga web 2 στον υπολογιστή μου**

## Παράρτημα Β

### Εγκατάσταση Exim internet mailer

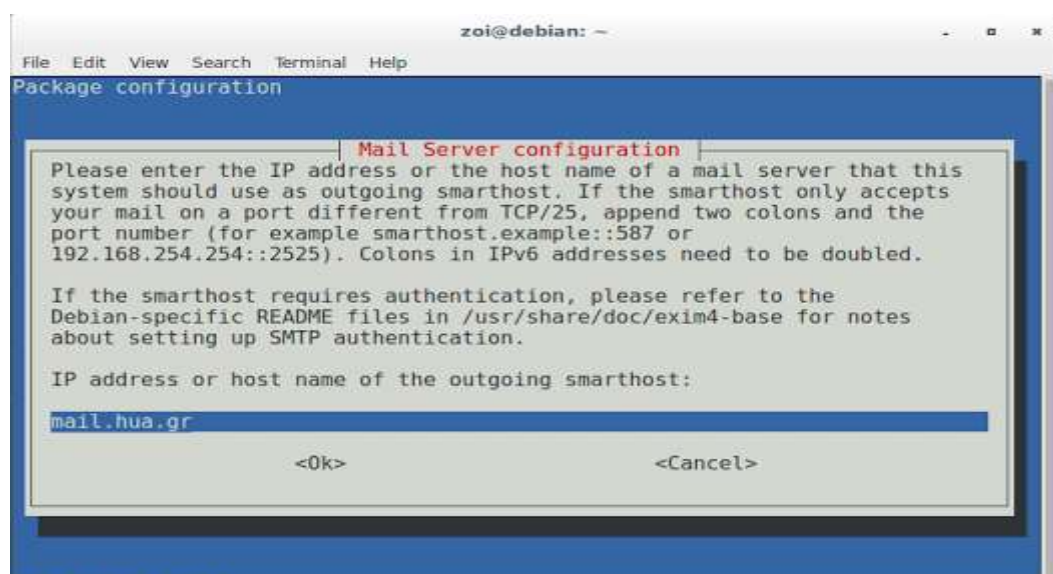
Η λύση στο πρόβλημα που αναφέρθηκε ενότητας 4.2 βρέθηκε μέσω του Exim(Exim internet mailer), δηλαδή ενός παράγοντα ο οποίος είναι υπεύθυνος για την μεταφορά μηνυμάτων, ο οποίος αναπτύχθηκε στο Πανεπιστήμιο του Cambridge.

Είναι ελεύθερα διαθέσιμο υπό τους όρους της GNU General Public Licence.χρήση σε συστήματα Unix συνδεδεμένα στο διαδίκτυο. Υπάρχει μια μεγάλη ευελιξία στον τρόπο ταχυδρόμησης που μπορεί να δρομολογηθεί, ενώ υπάρχουν εκτεταμένες ευκολίες για τον έλεγχο της εισερχόμενης αλληλογραφίας. Το Exim μπορεί να εγκατασταθεί στη θέση του sendmail, αν και η διαμόρφωση του Exim είναι αρκετά διαφορετική. Η τελευταία έκδοση και αυτή που χρησιμοποιήσαμε και εμείς είναι 4.86. [8]

Η εγκατάσταση έγινε μέσω της εντολής:

- dpkg-reconfigure exim4-config
- service icinga2 restart

Ακολουθήσαμε τα βήματα εγκατάστασης, κρατώντας τις προεπιλεγμένες ρυθμίσεις επεκτείνοντας κάπως από το αρχικό το σημείο που φαίνεται ακριβώς από κάτω:



Εικόνα 53: Διαμόρφωση Mail Server

Στο σημείο αυτό υπήρχε το mail server του debian ωστόσο έγινε η σωστή παραμετροποίηση με το mail server του Χαροκοπείου. Στο πέραν αυτών των διαδικασιών η διαδικασία αποστολής ειδοποιήσεων γινόταν επιτυχώς. [7], [8]

## Παράρτημα Γ

### Αλλαγή λογότυπου στον web interface

Ως τελευταία διαμόρφωση του συστήματος αυτού αφήσαμε την αλλαγή του λογότυπου. Φυσικά δεν θα μπορούσαμε να αρκεστούμε με την εικόνα του ICINGA γι' αυτό προσθέσαμε το λογότυπο του Χαροκοπείου Πανεπιστημίου.

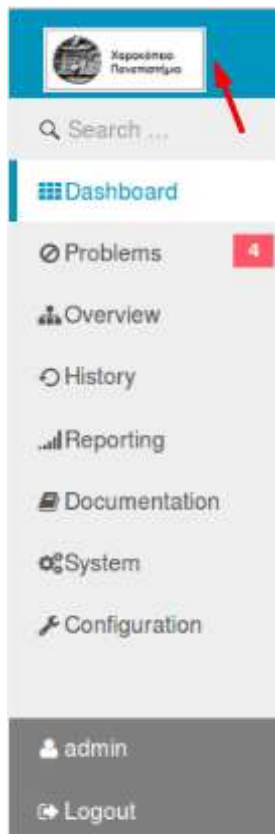
Πως έγινε αυτό; Με τα εξής βήματα:

1. Βρίσκουμε το directory: `find / -name logo_icinga-inv.png`
2. Μετονομάζουμε την φωτογραφία: `mv logo_icinga-inv.png logo_icinga-inv_original.png`
3. Μεταφέρουμε την εικόνα που θέλουμε στο ίδιο σημείο: `cp 150.png /usr/share/icingnaweb2/public/img`

150.png είναι η ονομασία που δόθηκε από εμάς ως αποθήκευση της εικόνας στον desktop του υπολογιστή μας.

4. Μετονομάσαμε την εικόνα μας στο όνομα που πρέπει: `mv 150.png logo_icinga-inv.png`

Το αποτέλεσμα:





## Βιβλιογραφία

1. Network Management System (NMS). [Ηλεκτρονικό] Σύνδεσμος:  
<https://www.techopedia.com/definition/11988/network-management-system-nms>
2. Το πρωτόκολλο SNMP. [Ηλεκτρονικό] Σύνδεσμος:  
[http://conta.uom.gr/conta/ekpaideysh/metaptyxiaka/technologies\\_diktywn/teaching\\_m/management/snmp.htm](http://conta.uom.gr/conta/ekpaideysh/metaptyxiaka/technologies_diktywn/teaching_m/management/snmp.htm)
3. **Eric Lippmann**, Icinga Web 2.0.0 – The final version is unleashed. [Ηλεκτρονικό] [Oct2,2015] Σύνδεσμος <https://www.icinga.org/2015/10/02/icinga-web-2-0-0-the-final-version-is-unleashed/>
4. Icinga2 Documentation. [Ηλεκτρονικό] Σύνδεσμος:  
<http://docs.icinga.org/icinga2/latest/doc/module/icinga2/toc>
5. Icinga2 Tutorial: Part 2 - Agent-Less Checks. [Ηλεκτρονικό] Σύνδεσμος:  
<http://zyradyl.github.io/2015/08/16/Icinga2-Tutorial-Part-2/>
6. Icinga2 Tutorial Part 4 - Expanding Checks to SNMP. [Ηλεκτρονικό] Σύνδεσμος:  
<http://zyradyl.github.io/2015/09/07/Icinga2-Tutorial-Part-4/>
7. Exim – wiki debian [Ηλεκτρονικό] Σύνδεσμος:  
<https://wiki.debian.org/Exim>
8. Exim internet mailer – Latest version: 4.86. [Ηλεκτρονικό] Σύνδεσμος:  
<http://www.exim.org/>
9. Σύγχρονα λειτουργικά συστήματα 3<sup>η</sup> Αμερικάνικη έκδοση – **Andrew S.Tanenbaum** [Βιβλίο] Επιστημονική επιμέλεια Ελληνικής έκδοσης Δημήτρης Γκιζόπουλος, Πανεπιστήμιο Πειραιώς.
10. Δίκτυα υπολογιστών 4<sup>η</sup> Αμερικάνικη έκδοση – **Andrew S. Tanenbaum** [Βιβλίο] Μετάφραση Γ. Ξυλωμένος 2008.
11. Νικόλαος Πρεβές. Ασύρματα Δίκτυα Υπολογιστών Ασφάλεια Και Απόδοση των πρωτοκόλλων TCP/IP. [Ηλεκτρονικό] [2008] Σύνδεσμος:  
[https://books.google.gr/books?id=0KiO\\_W\\_3w24C&pg=PA218&lpg=PA218&dq=%CE%B2%CE%AC%CF%83%CE%B7%CF%82+%CF%80%CE%BB%CE%B7%CF%81%CE%BF%CF%86%CE%BF%CF%81%CE%B9%CF%8E%CE%BD+%CE%B4%CE%B9%CE%B1%CF%87%CE%B5%CE%AF%CF%81%CE%B9%CF%83%CE%B7%CF%82+MIB&source=bl&ots=PeHH4Scg3G&sig=vY3ghFmgRNL0UGiwli36x6ZrrkE&hl=el&sa=X&ved=0ahUKEwjNs5G41dHKAhXKXBoKHbZFAwYQ6AEIPjAG#v=onepage&q&f=false](https://books.google.gr/books?id=0KiO_W_3w24C&pg=PA218&lpg=PA218&dq=%CE%B2%CE%AC%CF%83%CE%B7%CF%82+%CF%80%CE%BB%CE%B7%CF%81%CE%BF%CF%86%CE%BF%CF%81%CE%B9%CF%8E%CE%BD+%CE%B4%CE%B9%CE%B1%CF%87%CE%B5%CE%AF%CF%81%CE%B9%CF%83%CE%B7%CF%82+MIB&source=bl&ots=PeHH4Scg3G&sig=vY3ghFmgRNL0UGiwli36x6ZrrkE&hl=el&sa=X&ved=0ahUKEwjNs5G41dHKAhXKXBoKHbZFAwYQ6AEIPjAG#v=onepage&q&f=false)
12. About Icinga 2 [Ηλεκτρονικό] Σύνδεσμος:  
<https://docs-icinga.beutner.name/1-about/>
13. Icinga 2 – Architecture Modular and flexible [Ηλεκτρονικό] Σύνδεσμος:  
<https://www.icinga.org/icinga/icinga-2/architecture/>

14. Installion.co.uk – How to uninstall icinga2 [Ηλεκτρονικό] Σύνδεσμος:  
<http://installion.co.uk/ubuntu/wily/universe/i/icinga2/uninstall/index.html>
15. Debian Monitoring Project [Ηλεκτρονικό] Σύνδεσμος:  
<http://debmon.org/packages>
16. Icinga/ icingaweb2 [Ηλεκτρονικό] Σύνδεσμος:  
<https://github.com/Icinga/icingaweb2/blob/v2.0.0-rc1/doc/installation.md#upgrading-to-icinga-web-2-release-candidate-1>