ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΣΥΣΤΗΜΑ ΠΑΡΑΚΟΛΟΥΘΗΣΗΣ ΤΩΝ ΕΞΥΠΗΡΕΤΗΤΩΝ ΚΑΙ ΤΩΝ ΔΙΚΤΥΑΚΩΝ ΣΥΣΚΕΥΩΝ

ZABBIX

Του φοιτητή
Δημήτριου Κώττα
Αρ. Μητρώου: 154480

Επιβλέπων
Αντώνης Σιδηρόπουλος
Αναπληρωτής Καθηγητής

Ημερομηνία 12/5/2024

Τίτλος Π.Ε. Σύστηματα παρακολούθησης των εξυπηρετητών και των δικτυακών συσκευών
Κωδικός Π.Ε. 24184
Ονοματεπώνυμο φοιτητή/των Κώττας Δημήτριος
Ονοματεπώνυμο εισηγητή¨: Σιδηρόπουλος Αντώνης
Ημερομηνία ανάληψης Π.Ε. 30/03/2024
Ημερομηνία περάτωσης Π.Ε. 12/5/2024

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως πτυχιακή εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Πρόλογος

Τα υπολογιστικά και δικτυακά συστήματα που απαρτίζουν τις υποδομές κάθε οργανισμού είναι ένα αναπόσπαστο κομμάτι καθώς και πολύ σημαντικό για την σωστή λειτουργία των υπηρεσιών και άλλων λειτουργιών. Με όλο και περισσότερο να αυξάνονται αυτές οι ανάγκες για την αύξηση των δικτυακών υποδομών αλλά και για τους Servers, είναι όλο και ποιο δύσκολο να ελέγχεται η ορθή λειτουργίας τους. Αυτό το πρόβλημα έρχεται να λύση αυτή η πτυχιακή για τις υποδομές του πανεπιστημίου όπου θα ειδοποιεί αν κάτι δεν λειτουργεί σωστά αλλά θα παρέχει και δεδομένα για τον εκτενή έλεγχο των συστημάτων.

Περίληψη

Ο σκοπός της παρούσας πτυχιακής εργασίας είναι η εγκατάσταση και διαμόρφωση του Zabbix για την υλοποίηση συστήματος παρακολούθησης των δικτυακών συσκευών καθώς και τον Servers του πανεπιστημίου. Επιπλέον πραγματοποιείται εγκατάσταση του λογισμικού Mattermost το οποίο θα βοηθήσει στην συγκροτημένη συλλογή των alerts που θα παράγει το Zabbix. Στο θεωρητικό μέρος, πρόκειται να παρουσιαστούν οι βασικές έννοιες της παρακολούθησης δικτύου καθώς και αρκετές έννοιες που εφαρμόστηκαν για την υλοποίηση του συστήματος παρακολούθησης στο πρακτικό κομμάτι. Επίσης παρουσιάζονται τεχνολογίες που χρησιμοποιούνται γενικά στην δικτυακή παρακολούθηση αλλά και αναλύονται αυτές που χρησιμοποιούνται στη συγκεκριμένη υλοποίηση. Επίσης παρουσιάζονται γενικές πληροφορίες και πλεονεκτήματα του Zabbix και Mattermost. Τέλος φαίνονται τα βήματα που ακολουθήθηκαν στο πρακτικό κομμάτι για την υλοποίηση του συστήματος παρακολούθησης καθώς και παραδείγματα του πρακτικού κομματιού.

«Monitoring system of servers and network devices»

Kottas Dimitrios

Abstract

The purpose of this thesis is the installation and configuration of Zabbix for the implementation of a monitoring system for the network devices and the servers of the university. In addition you are installing the Mattermost software which will help in the structured collection of alerts that Zabbix will generate. In the theoretical part, the basic concepts of network monitoring as well as several concepts applied to implement the monitoring system in the practical part are to be presented. Also, technologies that are generally used in network monitoring are presented and those used in the specific implementation are analyzed. General information and advantages of Zabbix and Mattermost are also presented. Finally, the steps followed in the practical part for the implementation of the monitoring system are shown as well as examples of the practical part.

Ευχαριστίες

Θα ήθελα να εκφράσω τις θερμότερες μου ευχαριστίες σε όλους εκείνους που με βοήθησαν να ολοκληρώσω τη Πτυχιακή μου Εργασία καθώς και σε όλους τους φίλους μου που ήταν κοντά μου και με στηρίζαν ψυχολογικά σε όλη την διάρκεια.

Επίσης, θα ήθελα να ευχαριστήσω τον επιβλέποντα της εργασίας μου, καθηγητή κ. Αντώνη Σιδηρόπουλο για την εμπιστοσύνη που μου έδειξε καθώς και για την ομαλή συνεργασία που υπήρξε.

Τέλος, ένα μεγάλο και εγκάρδιο ευχαριστώ στην οικογένεια μου, η οποία στήριξε τις σπουδές μου με διάφορους τρόπους, φροντίζοντας για την καλύτερη δυνατή μόρφωση μου.

Πίνακας περιεχομένων

Πρόλογος iii
Περίληψη iv
Abstract vi
Ευχαριστίες vii
Κατάλογος Εικόνων xi
Συντομογραφίες xii
Κεφάλαιο 1ο: Τεχνολογικό Πλαίσιο 13
1.1 Εισαγωγή 13
1.2 Υλικό 13
1.2.1 Servers 13
1.2.2 Εικονικές Μηχανές (VMs) 14
1.2.3 Switches 14
1.3 Λογισμικό 15
1.3.1 Linux 15
1.3.2 Proxmox 15
1.3.3 Apache 16
1.4 Δικτύωση 16
1.4.1 Domain 16
1.4.2 SSL Certificate 17
1.5 Πρωτόκολλα διαχείρισης και ελέγχου δικτύου 17
1.5.1 Πρωτόκολλο SNMP 17
1.5.2 Πρωτόκολλο ICMP 18
1.6 Επίλογος 18
Κεφάλαιο 2ο: Τι είναι το Monitoring και η ανάγκες του 19
2.1 Εισαγωγή στη δικτυακή παρακολούθηση 19
2.2 Τα είδη του Monitoring 19
2.2.1 Δικτυακό Monitoring 19
2.2.2 Ανάλυση διαδρομής 19
2.2.3 Παρακολούθηση ιστότοπου 20
2.2.4 Συμπέρασμα 20
2.3 Τα προνόμια του Monitoring για το τμήμα IT 20
Κεφάλαιο 3ο: Εναλλακτικοί τρόποι του δικτυακού Monitoring 21
3.1 Εισαγωγή 21

3.2 Διαφορές ανοιχτού και ιδιόκτητου κώδικα 21
3.3 Agent-based και Agentless monitoring 21
3.3.1 Εισαγωγή 21
3.3.2 Agent-based monitoring 21
3.3.3 Agentless monitoring 22
3.3.4 Συμπεράσματα 23
3.4 Ανακάλυψη χαμηλού επιπέδου 23
3.5 Αυτόματη ανακάλυψη 24
3.6 Λογική Ομαδοποίηση 24
Κεφάλαιο 4ο: Zabbix 25
4.1 Επισκόπηση του Zabbix 25
4.2 Πλεονεκτήματα του Zabbix 25
4.3 Εγκατάσταση του Zabbix 26
4.3.1 Διεργασία πριν την εγκατάσταση του Zabbix 26
4.3.2 Εγκατάσταση Zabbix Server 26
4.4 Δημιουργία Host 28
4.4.1 Δημιουργία Host με SNMPv2 28
4.4.2 Δημιουργία Host με Zabbix Agent 29
4.4.3 Zabbix templates 30
4.4.4 Items 31
4.4.5 Triggers 32
4.5 Agentless και Agent client 33
4.5.1 Agentless SNMP client 33
4.5.2 Agent based Zabbix client 34
4.6 Σύστημα ειδοποίησης 35
4.6.1 Σύστημα ειδοποίησης μέσω Email 35
4.6.2 Σύστημα ειδοποίησης στο Mattermost 38
4.7 Zabbix apache 40
Κεφάλαιο 5ο: Mattermost 42
5.1 Επισκόπηση του Mattermost 42
5.2 Τρόπος λειτουργίας του Mattermost 42
5.3 Πλεονεκτήματα του Mattermsot 42
5.4 Εγκατάσταση του Mattermost 43
5.5 Mattermost bot για τις Zabbix ειδοποιήσεις 44
5.6 Mattermost apache 46

Κεφάλαιο 6ο: Συμπεράσματα 48
Βιβλιογραφία.....49

Κατάλογος Εικόνων

Εικόνα 1. *HTTPS site* 17
Εικόνα 2. Διαμόρφωση αρχείου Zabbix_server.conf 27
Εικόνα 3. Zabbix Main Page 28
Εικόνα 4. SNMP host creation 29
Εικόνα 5. Agent host creation 30
Εικόνα 6. Storage Usage Pie 31
Εικόνα 7. Ram Usage Graph 31
Εικόνα 8 items 32
Εικόνα 9 Trigger status 32
Εικόνα 10 Storage Triggers 32
Εικόνα 11 Network Triggers 33
Εικόνα 12 dashboard triggers 33
Εικόνα 13 Μορφοποίηση SNMP.conf 34
Εικόνα 14. Αλλαγές στο αρχείο Zabbix_agentd.conf 35
Εικόνα 15. Υλοποίηση Email συστήματος 36
Εικόνα 16. Email και σοβαρότητα ειδοποιήσεων 36
Εικόνα 17. Action παραμετροποίηση 37
Εικόνα 18. Ειδοποίηση προβλήματος 38
Εικόνα 19. Ειδοποίηση επίλυσης προβλήματος 38
Εικόνα 20. Mattermost webhook 39
Εικόνα 21. Mattermost κανάλι και σοβαρότητα ειδοπίησης 40
Εικόνα 22. Redirect http σε https 40
Εικόνα 23. HTTPS ρυθμίσεις 41
Εικόνα 24. Mattermost.service αρχείο 44
Εικόνα 25. Δημιουργία Bot 45
Εικόνα 26. Mattermost κανάλι με τις ειδοποιήσεις 46
Εικόνα 27. http to https redirect 46
Εικόνα 28. Https ρυθμίσεις και proxy 47

Συντομογραφίες

ΔΙΠΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
Π.Ε.	Πτυχιακή Εργασία
DNS	Domain Name Server
IT	Information Technology
LLD	Low Level Discovery
VM	Virtual Machine
MIB	Management Information Base
SMI	Structure of Management Information
ICMP	Internet Control Message Protocol
DDoS	distributed denial-of-service

Κεφάλαιο 1ο: Τεχνολογικό Πλαίσιο

1.1 Εισαγωγή

Στο πρώτο κεφάλαιο θα εξεταστεί το τεχνολογικό περιβάλλον στο οποίο θα αναπτυχθεί το σύστημα monitoring για το Πανεπιστήμιο. Αρχικά θα υπάρξει μια εισαγωγή στο υλικό κομμάτι όπου θα αναλυθεί η σημασία των Servers στο περιβάλλον του δικτύου, θα εξεταστεί ο κόσμος των VMs εξηγώντας τι είναι και γιατί είναι καθοριστικές για την υλοποίηση του Zabbix και του Mattermost καθώς και θα υπάρξει επεξήγηση ως το προς τι είναι η συσκευές δικτύου switch. Έπειτα θα παρουσιαστεί το λογισμικό κομμάτι όπου θα αναλυθεί το Linux ως λειτουργικού συστήματος για servers, το Proxmox, που είναι ένα εργαλείο που παρέχει ενοποιημένη διαχείριση των virtual machines (VMs) και containers, εξετάζοντας πώς λειτουργεί και τα πλεονεκτήματά του σε σχέση με άλλες λύσεις καθώς και ο Apache και η λειτουργία του. Στη συνέχεια, θα εξεταστεί ο ρόλος των domains στη συνδεσιμότητα προς τα domains που δημιουργήθηκαν για τις εφαρμογές όπως και η ασφάλεια που προσφέρουν τα SSL certificates κατά τη σύνδεση μέσω HTTPS. Τέλος, θα επεξηγηθούν τα πρωτόκολλα SNMP και ICMP τα οποία βοηθούν στην διαχείριση και στον έλεγχο του δικτύου.

1.2 Υλικό

1.2.1 Servers

Ένας server λειτουργεί ως γέφυρα μεταξύ υπολογιστών, παρέχοντας πόρους, υπηρεσίες και δεδομένα σε άλλες συσκευές που αποκαλούνται πελάτες. Είναι ο κεντρικός πυλώνας ενός δικτύου, διαθέτοντας ποικίλες λειτουργίες όπως οι web servers, mail servers και virtual servers. Ένας server μπορεί να παρέχει και να διαχειρίζεται πόρους ταυτόχρονα, λειτουργώντας ως server και client ανάλογα με την ανάγκη. [1].

Οι πρώτοι servers ήταν συστήματα όπως οι μεγάλοι υπολογιστές (mainframe) ή οι μικρότεροι υπολογιστές (minicomputers). Τα minicomputers προορίζονταν αρχικά για μικρότερες εφαρμογές σε σχέση με τους mainframe, αλλά στη συνέχεια έγιναν μεγαλύτεροι από τους επιτραπέζιους υπολογιστές. Αυτή η αλλαγή στο μέγεθος κατέστησε αστείο τον όρο "microcomputer". Αρχικά, αυτοί οι servers συνδέονταν με τερματικά που δεν εκτελούσαν υπολογισμούς, γνωστά ως "απλά τερματικά", που απλά δέχονταν εισαγωγή δεδομένων και εμφάνιζαν τα αποτελέσματα. Οι πραγματικοί υπολογισμοί γινόντουσαν στον server. Στη συνέχεια, οι servers συνδέθηκαν με client υπολογιστές μέσω δικτύου, χρησιμοποιώντας το μοντέλο client-server, όπου και οι δύο διαθέτουν υπολογιστική ισχύ, αλλά ορισμένες εργασίες εκτελούνται από τον server. Στα προηγούμενα μοντέλα υπολογιστών, όπως το μοντέλο mainframe-terminal, το mainframe διαδραμάτιζε τον ρόλο του server, ακόμη κι αν δεν αναφερόταν με αυτή την ονομασία [1].

Όσο η τεχνολογία προχωρά, έτσι εξελίσσεται και η έννοια του server. Στις αρχές, ένας server ήταν ένας υπολογιστής που παρείχε πόρους ή υπηρεσίες σε άλλους υπολογιστές, γνωστούς ως πελάτες, μέσω ενός δικτύου. Αυτή η δομή άλλαξε σταδιακά με την εξέλιξη της τεχνολογίας. Σήμερα, ένας server μπορεί να είναι απλώς ένα πρόγραμμα λογισμικού που εκτελείται σε έναν ή περισσότερους φυσικούς ή εικονικούς υπολογιστές. Οι virtual servers είναι ένα παράδειγμα αυτής της εξέλιξης. Αρχικά, χρησιμοποιήθηκαν για να αυξηθεί ο αριθμός των λειτουργιών που μπορούσε να φιλοξενήσει ένας μόνος φυσικός server. Σήμερα, οι virtual servers συχνά λειτουργούν σε υλικό που βρίσκεται στο Διαδίκτυο, σε μια διάταξη που ονομάζεται cloud computing. Αυτή η εξέλιξη έχει αλλάξει τον τρόπο με τον οποίο εκτελούνται οι

εργασίες στο διαδίκτυο. Πλέον, οι servers μπορεί να εκτελούν μια εργασία ή πολλαπλές εργασίες, ανάλογα με την ανάγκη και την αρχιτεκτονική του συστήματος [1].

1.2.2 Εικονικές Μηχανές (VMs)

Μια εικονική μηχανή (VM) είναι ένα εργαλείο υπολογιστικού πόρου που αξιοποιεί λογισμικό αντί για τον φυσικό υπολογιστή για την εκτέλεση προγραμμάτων και την εγκατάσταση εφαρμογών. Το VM μπορεί να εκτελείται σε έναν κεντρικό υπολογιστή, γνωστό ως "host", και επιτρέπει τη λειτουργία πολλών εικονικών μηχανών παράλληλα. Κάθε εικονική μηχανή λειτουργεί ανεξάρτητα, έχοντας το δικό της λειτουργικό σύστημα, ακόμα και όταν τρέχει στον ίδιο host. Αυτό επιτρέπει την εκτέλεση διαφορετικών λειτουργικών συστημάτων ή τη δοκιμή εφαρμογών σε απομονωμένα περιβάλλοντα, γνωστά ως "sandboxed" περιβάλλοντα, ενώ επιτρέπει επίσης την αποτελεσματική χρήση των πόρων υπολογιστή. Η τεχνολογία των εικονικών μηχανών έχει επιτρέψει στις επιχειρήσεις να εξοικονομήσουν πόρους και να αυξήσουν την αποδοτικότητα των υπολογιστικών τους συστημάτων. Οι εικονικές μηχανές έχουν ιστορικά χρησιμοποιηθεί για την εικονικοποίηση των servers, επιτρέποντας στις ομάδες πληροφορικής να βελτιώσουν την απόδοση τους. Επιπλέον, οι εικονικές μηχανές μπορούν να εκτελέσουν εργασίες που θεωρούνται ριψοκίνδυνες για το περιβάλλον του κεντρικού υπολογιστή, όπως η πρόσβαση σε δεδομένα που μπορεί να έχουν μολυσματικό λογισμικό ή η εκτέλεση δοκιμών λειτουργικών συστημάτων. Λόγω του ότι η εικονική μηχανή είναι διαχωρισμένη από το υπόλοιπο σύστημα, το λογισμικό που τρέχει σε αυτήν δεν μπορεί να επηρεάσει τον κεντρικό υπολογιστή [2].

Οι εικονικές μηχανές είναι εύκολο να διαχειριστούν και να συντηρηθούν και προσφέρουν αρκετά πλεονεκτήματα σε σχέση με τις φυσικές μηχανές [2]:

- Οι Εικονικές Μηχανές παρέχουν τη δυνατότητα εκτέλεσης πολλαπλών περιβαλλόντων λειτουργικών συστημάτων σε έναν μόνο φυσικό υπολογιστή, εξοικονομώντας φυσικό χώρο, χρόνο και κόστος διαχείρισης.
- Οι εικονικές μηχανές υποστηρίζουν παλαιότερες εφαρμογές, μειώνοντας το κόστος μετάβασης σε ένα νέο λειτουργικό σύστημα. Για παράδειγμα, μια εικονική μηχανή Linux μπορεί να λειτουργεί ως φιλοξενούμενο λειτουργικό σύστημα σε έναν κεντρικό διακομιστή που εκτελεί ένα διαφορετικό λειτουργικό σύστημα, όπως τα Windows.
- Οι εικονικές μηχανές παρέχουν ολοκληρωμένες επιλογές ανάκτησης μετά από καταστροφές.

Η εικονική μηχανή έχει πολλά πλεονεκτήματα, αλλά όπως και με κάθε τεχνολογία, υπάρχουν και ορισμένα πιθανά μειονεκτήματα. Ανάμεσα σε αυτά μπορούν να είναι [2]:

- Απόδοση: Η εκτέλεση εικονικών μηχανών μπορεί να οδηγήσει σε μειωμένη απόδοση σε σύγκριση με τις φυσικές μηχανές, καθώς υπάρχει μια επιπλέον στρώση επεξεργασίας που απαιτείται για τη διαχείριση των εικονικών περιβαλλόντων.
- Κόστος: Η χρήση εικονικών μηχανών μπορεί να συνεπάγεται υψηλότερο κόστος σε σύγκριση με τη χρήση φυσικών μηχανών, καθώς απαιτείται επιπλέον υλικό και λογισμικό για την εγκατάσταση και τη συντήρηση του εικονικού περιβάλλοντος.

1.2.3 Switches

Το switch είναι είδος συσκευών οι οποίες συνδέουν άλλες συσκευές σε ένα δίκτυο υπολογιστών χρησιμοποιώντας τη μεταγωγή πακέτων για τη λήψη και προώθηση των δεδομένων στη που είναι να σταλούν τα πακέτα.

Είναι μια συσκευή που έχει πολλές θύρες (πχ 24 θύρες) και χρησιμοποιεί τις διευθύνσεις MAC (επίπεδο 2) του μοντέλου OSI για την αποστολή των πακέτων στον σωστό προορισμό. Ορισμένα switch μπορούν να λειτουργίσων και στο επίπεδο δικτύου (επίπεδο 3 του μοντέλου OSI) ενσωματώνοντας επιπλέον

λειτουργίες δρομολόγησης. Τέτοια switches είναι συνήθως αρκετά ποιο ακριβά από τα απλά του επιπέδου 2 και επίσης είναι γνωστά ως switches επιπέδου 3 ή switch πολλαπλών επιπέδων [3].

Σε αντίθεση με τα repeater hubs (παλιές συσκευές που αντικαταστάθηκαν πλέον από τα switches), τα οποία μεταδίδουν τα ίδια δεδομένα από κάθε θύρα και αφήνουν τις συσκευές να διαλέξουν τα δεδομένα που απευθύνονται σε αυτές, ένας switch μαθαίνει τις διευθύνσεις Ethernet των συνδεδεμένων συσκευών και στη συνέχεια προωθεί τα δεδομένα μόνο στη θύρα που είναι συνδεδεμένη με τη συσκευή στην οποία απευθύνονται [3].

1.3 Λογισμικό

1.3.1 Linux

Το Linux αναπτύχθηκε ως ένα εναλλακτικό λειτουργικό σύστημα ανοιχτού κώδικα, το οποίο δημιουργήθηκε από τον Linus Torvalds το 1991. Από τότε, έχει μεταμορφωθεί σε ένα ισχυρό και ευέλικτο λειτουργικό σύστημα που χρησιμοποιείται παγκοσμίως. Ένα από τα βασικά χαρακτηριστικά του Linux είναι η ποικιλία των διανομών του. Οι διάφορες οργανώσεις και κοινότητες αναπτύσσουν τις δικές τους εκδόσεις, γνωστές ως διανομές ή distros. Ορισμένες δημοφιλείς διανομές Linux περιλαμβάνουν το Ubuntu, το Fedora, το Debian και το CentOS. Παρακάτω αναφέρονται κάποιοι από τους λόγους για τους οποίους επιλέγεται το Linux ως λειτουργικό σύστημα: [4]:

- Ελεύθερος Κώδικας: Ο κώδικας του Linux είναι ανοιχτός και προσβάσιμος σε όλους, επιτρέποντας στους χρήστες να προσαρμόσουν το λειτουργικό σύστημα στις ανάγκες τους.
- Ευελιξία: Το Linux είναι ευέλικτο και προσαρμόσιμο σε διάφορες πλατφόρμες και συσκευές, από φορητούς υπολογιστές και κινητά τηλέφωνα έως και servers.
- Ασφάλεια: Λόγω της ανοιχτής φύσης του κώδικα, το Linux επωφελείται από τη συνεχή εποπτεία και διόρθωση προβλημάτων ασφάλειας από την κοινότητα.
- Οικονομία: Το Linux είναι δωρεάν για λήψη και χρήση, μειώνοντας το κόστος απόκτησης και συντήρησης λογισμικού.
- Κοινότητα: Η ενεργή κοινότητα πίσω από το Linux παρέχει υποστήριξη, επίλυση προβλημάτων και συνεχή βελτίωση του συστήματος.

1.3.2 Proxmox

Το Proxmox Virtual Environment (Proxmox VE ή PVE) είναι ένα λογισμικό ανοικτού κώδικα για τη διαχείριση της εικονικοποίησης. Είναι ένας bare-metal hypervisor Type-1 που μπορεί να τρέξει λειτουργικά συστήματα, συμπεριλαμβανομένων των Linux και Windows, σε υλικό x64. Το Proxmox VE περιλαμβάνει μια διαδικτυακή κονσόλα και εργαλεία γραμμής εντολών. Παρέχεται επίσης ένα REST API για εργαλεία τρίτων κατασκευαστών. Υποστηρίζονται δύο τύποι εικονικοποίησης: container-based με το LXC (ξεκινώντας από την έκδοση 4.0 αντικαθιστώντας το OpenVZ που χρησιμοποιείται στις εκδόσεις έως και 3.4) και πλήρης εικονικοποίηση, με το KVM που περιλαμβάνει μια διεπαφή διαχείρισης μέσω διαδικτύου [5] [6] Επίσης, το Proxmox VE διαθέτει ευέλικτο μοντέλο αποθήκευσης. Τα images των εικονικών μηχανών μπορούν να αποθηκεύονται σε τοπικό ή κοινόχρηστο αποθηκευτικό χώρο, όπως NFS ή SAN (π.χ. χρησιμοποιώντας iSCSI ή FC). Είναι επίσης δυνατή η χρήση του DRBD για επισκέπτες KVM. [7].

Παρακάτω παρατίθενται κάποια από τα πλεονεκτήματα του Proxmox:

- Ευελιξία: Η δυνατότητα υποστήριξης τόσο VMs όσο και containers παρέχει μεγάλη ευελιξία στην ανάπτυξη και τη διαχείριση εφαρμογών, επιτρέποντας την εκτέλεση διαφορετικών τύπων εφαρμογών σε ένα ενιαίο περιβάλλον.

- Ενοποιημένος Έλεγχος: Η ενοποιημένη διαχείριση όλων των εργαλείων σε ένα σύστημα απλοποιεί τη διαχείριση και τη συντήρηση του IT περιβάλλοντος, εξασφαλίζοντας ομαλή λειτουργία και αποτελεσματική διαχείριση.
- Απόδοση και Αξιοπιστία: Η χρήση του KVM και του LXC εξασφαλίζει υψηλή απόδοση και αξιοπιστία στις εικονικές μηχανές και τα containers, επιτρέποντας την αποτελεσματική λειτουργία των εφαρμογών και των υπηρεσιών.

1.3.3 Apache

Ο Apache HTTP Server, ευρέως γνωστός απλά ως Apache, αποτελεί έναν εξυπηρετητή του παγκόσμιου ιστού (web server). Κάθε φορά που ένας χρήστης επισκέπτεται μια ιστοσελίδα, το πρόγραμμα πλοήγησης (browser) αλληλεπιδρά με έναν τέτοιο server μέσω του πρωτοκόλλου HTTP. Ο server αυτός παράγει τις ιστοσελίδες και τις αποστέλλει στο πρόγραμμα πλοήγησης, επιτρέποντας στους χρήστες να πλοηγούνται και να αλληλεπιδρούν με το περιεχόμενο του διαδικτύου. Ο Apache είναι ένας από τους δημοφιλέστερους εξυπηρετητές ιστού, εν μέρει γιατί λειτουργεί σε διάφορες πλατφόρμες όπως τα Windows, το Linux, το Unix και το Mac OS X. Κυκλοφόρησε υπό την άδεια λογισμικού Apache και είναι λογισμικό ανοιχτού κώδικα. Συντηρείται από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation) [8].

Ο Apache χρησιμοποιείται και σε τοπικά δίκτυα σαν διακομιστής συνεργαζόμενος με συστήματα διαχείρισης Βάσης Δεδομένων π.χ. Oracle, MySQL.

Η πρώτη του έκδοση, γνωστή ως NCSA HTTPd, δημιουργήθηκε από τον Robert McCool και κυκλοφόρησε το 1993. Θεωρείται ότι έπαιξε σημαντικό ρόλο στην αρχική επέκταση του παγκόσμιου ιστού. Ήταν η πρώτη βιώσιμη εναλλακτική επιλογή που παρουσιάστηκε απέναντι στον εξυπηρετητή http της εταιρείας Netscape και από τότε έχει εξελιχθεί στο σημείο να ανταγωνίζεται άλλους εξυπηρετητές βασισμένους στο Unix σε λειτουργικότητα και απόδοση. Από το 1996 ήταν από τους πιο δημοφιλείς όμως από τον Μάρτιο του 2006 έχει μειωθεί το ποσοστό της εγκατάστασής του κυρίως από τον Microsoft Internet Information Services και την πλατφόρμα .NET. Τον Οκτώβριο του 2007 το μερίδιο του ήταν 47.73% από όλους τους ιστοτόπους. Τον Μάρτιο του 2017, το 49,48% του συνόλου των καταχωρημένων Ελληνικών τομέων χρησιμοποιούσαν τον Apache [8].

1.4 Δικτύωση

1.4.1 Domain

Το Domain είναι μια συμβολοσειρά κειμένου (π.χ thesis.com) που αντιστοιχεί σε μια διεύθυνση IP, η οποία χρησιμοποιείται για την πρόσβαση σε έναν ιστότοπο από το πρόγραμμα περιήγησης. Απλούστερα, το domain name είναι αυτό που πληκτρολογεί ο χρήστης στη γραμμή διευθύνσεων ενός προγράμματος περιήγησης για να επισκεφτεί έναν συγκεκριμένο ιστότοπο. Για παράδειγμα, το domain name του youtube είναι "youtube.com" [9].

Λόγο του ότι ο πραγματικός τρόπος με τον οποίο συνδέονται σε έναν ιστότοπο είναι η IP (π.χ. 192.0.2.2), χρησιμοποιείτε το domain name για ευκολία. Με αυτόν τον τρόπο οι χρήστες μπορούν να το θυμούνται αλλά να είναι και ποιο εύκολο κατά την πληκτρολόγηση τους ο προορισμός όπως το "youtube.com" ή το "Instagram.com". Αυτή η διαδικασία, όπου οι ανθρώπινοι χρήστες εισάγουν ονόματα τομέα και μεταφράζονται σε διευθύνσεις IP, είναι γνωστή ως αναζήτηση DNS [9].

1.4.2 SSL Certificate

Τα πιστοποιητικά SSL είναι βασικά για την ασφάλεια των ιστότοπων. Επιτρέπουν τη χρήση του HTTPS, που προσφέρει ασφαλή σύνδεση μεταξύ του περιηγητή του χρήστη και του διακομιστή. Κατά την επίσκεψη ενός χρήστη σε μια ιστοσελίδα, ο browser ανατρέχει στο πιστοποιητικό για επαλήθευση της ταυτότητας του ιστότοπου και λήψη του δημόσιου κλειδιού του διακομιστή. Το ιδιωτικό κλειδί παραμένει ασφαλές και δεν αποκαλύπτεται. Με αυτόν τον τρόπο, οι πληροφορίες των χρηστών παραμένουν ιδιωτικές και οι επικοινωνίες τους ασφαλείς, ενισχύοντας την εμπιστοσύνη στον ιστότοπο και προστατεύοντας τα δεδομένα από ανεπιθύμητη παρέμβαση [10].

Εικόνα 1. *HTTPS site*

Η κρυπτογράφηση SSL/TLS είναι δυνατή χάρη στην ασφαλή σύνδεση μεταξύ δημόσιου και ιδιωτικού κλειδιού, τα οποία παρέχονται από τα πιστοποιητικά SSL. Οι πελάτες, όπως οι περιηγητές ιστού, λαμβάνουν το δημόσιο κλειδί από το πιστοποιητικό SSL του διακομιστή, το οποίο απαιτείται για τη δημιουργία μιας ασφαλούς σύνδεσης TLS.

Τα πιστοποιητικά SSL επιβεβαιώνουν την αυθεντικότητα του διακομιστή στον οποίο συνδέεται ο πελάτης, εξασφαλίζοντας ότι αυτός είναι πράγματι ο σωστός διακομιστής για τον συγκεκριμένο τομέα. Αυτή η διαδικασία βοηθά στην αποφυγή παραποίησης του domain και άλλων μορφών επιθέσεων.

HTTPS: Το HTTPS είναι ζωτικής σημασίας για τις επιχειρήσεις, απαιτώντας ένα πιστοποιητικό SSL για να λειτουργήσει. Αποτελεί την ασφαλή μορφή του HTTP, καθώς οι ιστότοποι που χρησιμοποιούν HTTPS κρυπτογραφούν την κυκλοφορία τους με SSL/TLS [10].

Εκτός από την προστασία των δεδομένων των χρηστών κατά τη μεταφορά, το HTTPS δίνει στους ιστότοπους αξιοπιστία στα μάτια των χρηστών. Πολλοί χρήστες δεν διακρίνουν τη διαφορά μεταξύ μιας διεύθυνσης ιστότοπου που ξεκινάει με http:// και μιας που ξεκινάει με https://, αλλά τα περισσότερα προγράμματα περιήγησης επισημαίνουν τους ιστότοπους HTTP ως "μη ασφαλείς". Αυτό προωθεί τη μετάβαση σε HTTPS και αυξάνει την ασφάλεια.

1.5 Πρωτόκολλα διαχείρισης και ελέγχου δικτύου

1.5.1 Πρωτόκολλο SNMP

Το SNMP (Simple Network Management Protocol) αποτελεί ένα ευρέως χρησιμοποιούμενο πρωτόκολλο στα συστήματα διαχείρισης δικτύων. Βασίζεται στο πρωτόκολλο UDP, το οποίο δεν απαιτεί να διατηρείτε η σύνδεση μεταξύ του πράκτορα (agent) και του διαχειριστή (manager). Καθώς το Διαδίκτυο εξελίσσεται, το SNMP έχει γίνει η κύρια τεχνολογία διαχείρισης δικτύων. Ένα σύστημα διαχείρισης δικτύων βασισμένο στο SNMP παρέχει απλή δομή και ισχυρή επεκτασιμότητα. Συνήθως αποτελείται από τέσσερα βασικά στοιχεία: τον πράκτορα (agent), τον διαχειριστή (manager), το MIB (Βάση Πληροφοριών Διαχείρισης) και το SMI (Δομή Πληροφοριών Διαχείρισης). Ο διαχειριστής αποστέλλει αιτήσεις στον agent για να λάβει ή να ορίσει πληροφορίες στο MIB και να λαμβάνει traps από αυτόν. Ο πράκτορας απαντά στις αιτήσεις και στέλνει τα αποτελέσματα πίσω στο διαχειριστή. Το MIB χρησιμοποιείται για την οργάνωση των πληροφοριών διαχείρισης. Υπάρχουν πέντε βασικοί τύποι δεδομένων πρωτοκόλλου SNMP [11]:

- GetRequest: Όταν ο manager ζητάει συγκεκριμένες τιμές από τον agent.
- GetNextRequest: Όταν ο manager ζητάει από τον agent τις επόμενες μεταβλητές του δέντρου MIB.
- SetRequest: Με αυτό το request αλλάζει μια η περισσότερες τιμές στον agent.
- GetResponse: ο manager λαμβάνει απαντήσεις από τον agent με τις τιμές των ζητούμενων παραμέτρων.
- Trap: Με την λειτουργία trap ο agent ειδοποιεί τον manager ασύγχρονα όταν συμβεί μια σημαντική αλλαγή σε ένα σύστημα ή μια συσκευή δικτύου.

1.5.2 Πρωτόκολλο ICMP

Το ICMP (Internet Control Message Protocol) αποτελεί ένα πρωτόκολλο στο επίπεδο δικτύου το οποίο χρησιμοποιείται ευρέως για την ανταλλαγή μηνυμάτων διαχείρισης και διάγνωσης σε ένα δίκτυο. Κυρίως, χρησιμοποιείται για να επιβεβαιώσει εάν τα δεδομένα φθάνουν εγκαίρως στον προορισμό τους. Συνήθως εφαρμόζεται σε συσκευές δικτύου όπως routers, firewalls καθώς και Server. Πέραν της αναφοράς σφαλμάτων, το ICMP μπορεί να χρησιμοποιηθεί και σε επιθέσεις κατανεμημένης άρνησης υπηρεσιών (DDoS) [12].

Κύριος στόχος του ICMP είναι να ανιχνεύσει και να αναφέρει σφάλματα κατά τη μετάδοση δεδομένων στο δίκτυο. Όταν δύο συσκευές επικοινωνούν μέσω του Διαδικτύου και παρουσιάζεται κάποιο πρόβλημα, όπως η απώλεια δεδομένων, το ICMP δημιουργεί μηνύματα λάθους για να ειδοποιήσει την αρχική συσκευή αποστολής. Για παράδειγμα, εάν ένα πακέτο δεδομένων είναι υπερβολικά μεγάλο για έναν router, αυτός ο router θα απορρίψει το πακέτο και θα αποστέλλει ένα μήνυμα ICMP πίσω από εκεί που στάλθηκε το πακέτο [12].

Μια επιπλέον χρήση του πρωτοκόλλου ICMP είναι η δυνατότητα εκτέλεσης διαγνωστικών ελέγχων δικτύου. Τα ευρέως διαδεδομένα εργαλεία traceroute και ping αξιοποιούν το πρωτόκολλο ICMP για τη λειτουργία τους. Το εργαλείο traceroute χρησιμοποιείται για να ανιχνεύσει την διαδρομή δρομολόγησης μεταξύ δύο συσκευών στο Διαδίκτυο. Κάθε σύνδεση μεταξύ δύο συσκευών συμβολίζεται ως "hop" και το traceroute αναφέρει επίσης τον χρόνο που απαιτείται για κάθε hop κατά μήκος της διαδρομής. Αυτές οι πληροφορίες μπορούν να είναι χρήσιμες για να γίνει αντιληπτό σε ποια hops υπάρχει καθυστέρηση η ακόμη να χάνεται η επικοινωνία [12].

1.6 Επίλογος

Σε αυτό το κεφάλαιο έγινε παρουσίαση και περιγραφή των τεχνολογιών που έχουν συνεισφέρει πριν την υλοποίηση των Mattermost και Zabbix εφαρμογών καθώς και επεξήγηση αυτών ώστε να μπορεί να γίνει μελλοντική αναφορά πάνω σε αυτές τις τεχνολογίες.

Κεφάλαιο 2ο: Τι είναι το Monitoring και η ανάγκες του

2.1 Εισαγωγή στη δικτυακή παρακολούθηση

Οι δικτυακές συσκευές που συνδέονται στο διαδίκτυο πρέπει να παρακολουθούνται σε τακτά χρονικά διαστήματα σε πραγματικό χρόνο. Η παρακολούθηση του δικτύου αφορά την συλλογή δεδομένων ώστε να παρέχει στατιστικά σε πραγματικό χρόνο όπως και τις επιδόσεις του δικτύου. Όταν συμβεί δυσλειτουργία στο δίκτυο, ο διαχειριστής του δικτύου πρέπει να ενημερωθεί. Το δίκτυο πρέπει να διασφαλίζεται με την ειδοποίηση για τα πιθανά ζητήματα πριν αυτά τα προβλήματα μεγαλώσουν. Τρόποι όπως SMS και E-mails μπορούν να χρησιμοποιηθούν ώστε να ειδοποιήσουν τον διαχειριστή του δικτύου για το σχετικό πρόβλημα στο δίκτυο. Ο όρος "Network Monitoring" χρησιμοποιείται για να περιγράψει ένα σύστημα που βοηθά στη συνεχή παρακολούθηση του δικτύου και διαπιστώνει εάν υπάρχει κάποιο πρόβλημα, καθυστέρηση του συστήματος ή βλάβη και ειδοποιεί αμέσως τον διαχειριστή του δικτύου μέσω E-mail, SMS ή οποιουδήποτε άλλου τρόπου σε περίπτωση κάποιου προβλήματος. Η παρακολούθηση του δικτύου διαπιστώνεται ότι δεν έχει καμία χρησιμότητα εάν δεν παρακολουθούνται τα σωστά πράγματα. Οι συνήθεις τομείς που παρακολουθούνται είναι η χρήση του εύρους ζώνης, η απόδοση του Server και η απόδοση της εφαρμογής. Η παρακολούθηση των Servers απαρτίζεται από την παρακολούθηση του λειτουργικού συστήματος, την παρακολούθηση των υλικών καθώς και των εφαρμογών που «τρέχουν» στον Server. Οι βασικές μετρήσεις που παρακολουθούνται είναι ο χρόνος συστήματος CPU, ο χρόνος αναμονής CPU, η χρησιμοποιούμενη μνήμη, η ελεύθερη μνήμη, η χρήση του δίσκου, τα collisions του δικτύου, ο ρυθμός μετάδοσης προσαρμογέα κ.λπ. Οι προαναφερθέντες είναι οι ποιο βασικές μετρήσεις σε Server, router και switches καθώς και αρκετές άλλες μετρήσεις αναλόγως με τις ανάγκες του κάθε συστήματος.

2.2 Τα είδη του Monitoring

Τα εργαλεία παρακολούθησης δικτύου παρέχουν μια ευρεία γκάμα δυνατοτήτων σάρωσης και ανάλυσης για διάφορους τύπους συσκευών και υπηρεσιών στο δίκτυο. Αυτό επιτυγχάνεται μέσω της χρήσης διαφορετικών πρωτοκόλλων που λειτουργούν σε διαφορετικά επίπεδα του OSI (Open Systems Interconnection).

2.2.1 Δικτυακό Monitoring

Για την γενικότερη αξιολόγησή της απόδοσης του δικτύου χρησιμοποιείται η παρακολούθηση του. Αυτό περιλαμβάνει τη μέτρηση της ποσότητας των μεταδιδόμενων και λαμβανόμενων πακέτων, καθώς και των καθυστερήσεων των δικτυακών συσκευών και των καθυστερήσεων διάδοσης κατά την διάρκεια της διαδρομής. Αυτό μας επιτρέπει να εκτιμήσουμε την απώλεια πακέτων, το εύρος ζώνης και τις καθυστερήσεις [13].

2.2.2 Ανάλυση διαδρομής

Ένα άλλο σημαντικό μέρος του Monitoring είναι η ανάλυση διαδρομής (route analytics). Αυτή η διαδικασία περιλαμβάνει ένα σύνολο εργαλείων, τεχνικών και αλγορίθμων που επιτρέπουν την παρακολούθηση της δρομολόγησης στο εσωτερικό ενός δικτύου, λειτουργώντας στο επίπεδο του δικτύου. Το Routing analytics παρακολουθεί παθητικά τα πρωτόκολλα δρομολόγησης OSPF, IS-IS, EIGRP και BGP, λαμβάνοντας κάθε μήνυμα ενημέρωσης από όλους τους δρομολογητές. Επιπλέον, χρησιμοποιεί τον αλγόριθμο Dijkstra για τον υπολογισμό του χάρτη τοπολογίας του δικτύου,

συμπεριλαμβανομένων όλων των διαδρομών. Τέλος, καταγράφει ένα πλήρες ιστορικό συμβάντων δρομολόγησης που μπορεί να χρησιμοποιηθεί για την αντιμετώπιση προβλημάτων στο μέλλον. Η ανάλυση διαδρομής συμβάλλει στην αύξηση της ταχύτητας και της αποδοτικότητας του δικτύου, μειώνοντας το κόστος και αυξάνοντας την παραγωγικότητα των εργαζομένων [13].

2.2.3 Παρακολούθηση ιστότοπου

Η παρακολούθηση του ιστότοπου προσφέρει λεπτομερή παρακολούθηση της κατάστασης του server. Καταγράφει τη διαθεσιμότητα του, τις επιδόσεις του, τη συνδεσιμότητά του, το χρόνο διαθεσιμότητας, τις εγγραφές DNS, το εύρος ζώνης και ακόμη και τους πόρους υλικού. Υπάρχουν δύο κύριοι τύποι παρακολούθησης ιστοτόπων: εσωτερική και εξωτερική. Η εσωτερική παρακολούθηση (εντός του firewall) εστιάζει στον εντοπισμό ζητημάτων που σχετίζονται με την εσωτερική υποδομή ή τον σχεδιασμό εφαρμογών. Η εξωτερική παρακολούθηση (εκτός του εταιρικού τείχους προστασίας) εστιάζεται στην παρακολούθηση από άκρο σε άκρο. Η παρακολούθηση ιστοτόπων καλύπτει τα πρωτόκολλα του Διαδικτύου, συμπεριλαμβανομένων των SSH, TELNET, POP3, DNS, SSL, UDP, HTTP, HTTPS, FTP, SNMP, STPM, και TCP [14].

2.2.4 Συμπέρασμα

Υπάρχουν διάφοροι τύποι παρακολούθησης, καθένας από τους οποίους έχει σχεδιαστεί για να εκτελεί συγκεκριμένους τύπους ελέγχων. Ο τρόπος λειτουργίας τους είναι εντελώς διαφορετικός και ο κάθε τύπος εφαρμόζεται ανάλογα με τις ανάγκες που υπάρχουν.

2.3 Τα προνόμια του Monitoring για το τμήμα IT

Με τη χρήση των εργαλείων παρακολούθησης δικτύου οι διαχειριστές δικτύων αποδεσμεύονται από την παρακολούθηση των δικτυακών συστημάτων και μπορούν να επικεντρωθούν σε πιο σημαντικά καθήκοντα. Θα έχουν πλήρη εικόνα σχετικά με το τι συμβαίνει στο δίκτυο τους. Επιπλέον, η αντιμετώπιση των βλαβών ρουτίνας θα μπορούσε να αυτοματοποιηθούν με τη βοήθεια ειδικών script. Δεν υπάρχει αμφιβολία ότι ορισμένες δυσλειτουργίες θα πρέπει ακόμη να επιλύονται χειροκίνητα. Ωστόσο, αυτό θα είναι σημαντικά ευκολότερο λόγω της λεπτομερούς διάγνωσης του προβλήματος.

Κεφάλαιο 3ο: Εναλλακτικοί τρόποι του δικτυακού Monitoring

3.1 Εισαγωγή

Κατά την επιλογή του εργαλείου παρακολούθησης δικτύου ο διαχειριστής δικτύου αντιμετωπίζει μια μεγάλη πρόκληση. Τη στιγμή αυτή, υπάρχουν πάνω από 60 διαθέσιμα εργαλεία από διαφορετικούς προμηθευτές στην αγορά. Τα περισσότερα από αυτά παρέχουν ευρεία γκάμα δυνατοτήτων. Παρ' όλα αυτά, υπάρχουν ορισμένες διαφορές που μπορεί να έχουν σημαντικό αντίκτυπο στο δίκτυο καθώς και στις απαιτήσεις του οργανισμού. Για αυτό τον λόγο, προκειμένου να επιλεχθεί η καλύτερη επιλογή για το δίκτυο μιας εταιρείας, θα πρέπει να αναλυθούν ορισμένα κρίσιμα χαρακτηριστικά.

3.2 Διαφορές ανοιχτού και ιδιόκτητου κώδικα

Στην αγορά υπάρχουν δύο τύποι εφαρμογών παρακολούθησης. Ο πρώτος τύπος είναι εργαλεία Monitoring ανοικτού κώδικα όπου οι προγραμματιστές τρίτων επιτρέπεται να κάνουν αλλαγές σε επίπεδο κώδικα. Ο δεύτερος τύπος είναι τα ιδιόκτητα εργαλεία Monitoring. Η άδεια χρήσης περιορίζει την πραγματοποίηση κάθε είδους τροποποίησης σε επίπεδο κώδικα. Λόγο του ότι οι πωλητές υπηρεσιών δικτυακού monitoring προσπαθούν να συμπεριλάβουν τα πιο σημαντικά χαρακτηριστικά για το monitoring, είναι σχεδόν αδύνατο να δημιουργηθεί μια λύση που θα είναι η βέλτιστη για κάθε δίκτυο. Διαφορετικά δίκτυα έχουν διαφορετικές ανάγκες. Για αυτό τον λόγο το δικτυακό monitoring ανοικτού κώδικα προσφέρει το πλεονέκτημα ότι εάν κάποιο χαρακτηριστικό που είναι σχετικό με το δικτυακό monitoring δεν περιλαμβάνεται στην έκδοση, τότε μπορεί να δημιουργηθεί από τον διαχειριστή του δικτύου ή να ληφθεί από την εκάστοτε κοινότητα.

Επιπλέον, οι εταιρείες που υιοθετούν προϊόντα ανοιχτού κώδικα επωφελούνται από το φαινόμενο του crowd sourcing. Αυτό συμβαίνει διότι όταν ανεξάρτητοι προγραμματιστές αναπτύσσουν πρόσθετες λειτουργίες, οι πιο δημοφιλείς από αυτές μπορούν να ενσωματωθούν στην επόμενη έκδοση του προϊόντος. Αυτή η διαδικασία παρέχει επιπλέον ευελιξία και παρέχει στην εταιρεία πολύτιμες πληροφορίες σχετικά με τις τάσεις στον τομέα της παρακολούθησης δικτύου Τέλος, τα εργαλεία δικτυακού monitoring ανοικτού κώδικα βοηθούν στην επέκταση ορισμένων χαρακτηριστικών που πιθανός να μην παρέχονται από τους προμηθευτές της εφαρμογής.

3.3 Agent-based και Agentless monitoring

3.3.1 Εισαγωγή

Μια άλλη σημαντική απόφαση για τους διαχειριστές συστημάτων είναι η επιλογή μεταξύ παρακολούθησης δικτύου που βασίζεται σε agents (agent-based) και παρακολούθησης χωρίς agents (agentless). Δεν υπάρχει μια μοναδική βέλτιστη λύση για όλα τα δίκτυα, καθώς αυτή η απόφαση εξαρτάται από το επιθυμητό επίπεδο παρακολούθησης και από τον τύπο των συσκευών που θα παρακολουθούνται στο δίκτυο. Για αυτόν τον λόγο θα εξεταστούν οι βασικές διαφορές μεταξύ τους καθώς και πότε είναι προτιμότερη η επιλογή του κάθε agent [15].

3.3.2 Agent-based monitoring

Η παρακολούθηση του δικτύου με agent-based προσέγγιση είναι μια μέθοδος παρακολούθησης συσκευών και συστημάτων δικτύου, όπου εγκαθίσταται agent λογισμικό σε κάθε συσκευή ή σύστημα που πρέπει να παρακολουθηθεί. Αυτοί οι agents συλλέγουν δεδομένα σχετικά με τη συσκευή ή το σύστημα, όπως μετρήσεις επιδόσεων και αρχεία καταγραφής συστήματος, και στη συνέχεια

μεταβιβάζουν αυτά τα δεδομένα σε μια κεντρική πλατφόρμα παρακολούθησης. Αυτό επιτρέπει την παρακολούθηση και ανάλυση ολόκληρου του δικτύου σε πραγματικό χρόνο, καθιστώντας δυνατή την ταχεία αναγνώριση και αντιμετώπιση προβλημάτων κατά την εμφάνισή τους [15].

3.3.2.1 Σημαντικά οφέλη

Ένα από τα κύρια πλεονεκτήματα της παρακολούθησης δικτύου με agent-based agent είναι ότι παρέχει λεπτομερείς πληροφορίες για μεμονωμένες συσκευές και συστήματα. Αυτή η προσέγγιση επιτρέπει την λεπτομερή παρακολούθηση και την αποτελεσματική αντιμετώπιση προβλημάτων, διευκολύνοντας τον γρήγορο εντοπισμό των αιτίων οποιουδήποτε ζητήματος. Ένα άλλο σημαντικό πλεονέκτημα της παρακολούθησης δικτύου με agent είναι η προληπτική φύση της. Επιπλέον, η παρακολούθηση με agent είναι πιο ασφαλής, καθώς τα δεδομένα συλλέγονται τοπικά και στη συνέχεια αποστέλλονται στην κεντρική πλατφόρμα παρακολούθησης, μειώνοντας τον κίνδυνο παραβίασης δεδομένων και μη εξουσιοδοτημένης πρόσβασης. Η παρακολούθηση με agent είναι ένα ισχυρό εργαλείο για τη διαχείριση και τη διατήρηση της απόδοσης και της ασφάλειας ενός δικτύου. Με την αναλυτική, προληπτική και προσαρμόσιμη προσέγγισή της, καθιστά δυνατή την ταχεία αναγνώριση και αντιμετώπιση προβλημάτων, την αποτροπή διαταραχών και διακοπών λειτουργίας και τη διασφάλιση της αποδοτικής και αποτελεσματικής λειτουργίας του δικτύου [15].

3.3.2.2 Τα αρνητικά του Agent-based monitoring

Για το monitoring με agent-based agent πρέπει να εγκαθίσταται agent και να συντηρείται. Για μία συσκευή, αυτό είναι εύκολο. Σε μεγάλο αριθμό αυτό μπορεί να δημιουργήσει σημαντικό βάρος για τούς διαχειριστές του δικτύου Επιπλέον υπάρχει περιορισμένη υποστήριξη συσκευών. Το σύνολο των συσκευών στις οποίες μπορεί να εγκατασταθεί ένας συγκεκριμένος monitoring agent είναι μικρότερος από το σύνολο των συσκευών που υποστηρίζουν τυποποιημένα πρωτόκολλα. Τέλος, υπάρχει περισσότερη κατανάλωση πόρων στη συσκευή. Η εκτέλεση ενός agent-based agent σε μια συσκευή μπορεί να αυξήσει αισθητά την κατανάλωση πόρων [16]].

3.3.3 Agentless monitoring

Με το Agentless monitoring, το κύριο χαρακτηριστικό του είναι ότι δεν χρειάζεται να γίνει εγκατάσταση κάποιου λογισμικού στην συσκευή ώστε να μπορέσει να λαμβάνει δεδομένα από αυτή. Αντ' αυτού, βασίζεται σε ενσωματωμένα πρωτόκολλα όπως το SNMP ή το ICMP για τη συλλογή δεδομένων και την παρακολούθηση της συσκευής ή του συστήματος [15].

3.3.3.1 Σημαντικά οφέλη

Ένα από τα κύρια πλεονεκτήματα του δικτυακού παρακολούθησης χωρίς πράκτορες είναι η απλότητα και η ευκολία ανάπτυξής του. Από τη στιγμή που δεν απαιτείται η εγκατάσταση του agent σε κάθε συσκευή, απαλλάσσει τους διαχειριστές του δικτύου από τη συντήρηση, ενημέρωση και διαχείριση αυτών των agents. Αυτό οδηγεί σε σημαντική εξοικονόμηση χρόνου και προσπάθειας. Ένα άλλο κύριο πλεονέκτημα της παρακολούθησης δικτύου χωρίς πράκτορες είναι η μικρότερη απαίτηση σε πόρους. Δεν απαιτείται επιπλέον λογισμικό για την εκτέλεσή του στις συσκευές, βοηθώντας έτσι στη διατήρηση της απόδοσης τους και στη μείωση του συνολικού φόρτου στο δίκτυο. Επιπλέον, μπορεί να παρακολουθήσει μεγάλο αριθμό συσκευών και συστημάτων με ελάχιστη επιβάρυνση στους πόρους. Τέλος, η παρακολούθηση χωρίς agent είναι πιο ασφαλής, καθώς δεν απαιτεί το άνοιγμα σύνδεσης με τις συσκευές για την παρακολούθησή τους, μειώνοντας τις πιθανές ευπάθειες της συσκευής [15].

3.3.3.2 Τα αρνητικά του Agentless monitoring

Ένα αρνητικό του agentless monitoring είναι ότι υπάρχει περιορισμός στα δεδομένα που μπορεί να συλλέξει, αναλόγως και την συσκευή. Πολλές φορές χρειάζεται η συλλογή κάποιον δεδομένων που δεν υποστηρίζεται από το agentless monitoring. Συνήθως χρησιμοποιείτε για την συλλογή συνηθισμένων δεδομένων [15].

Επιπλέον η συλλογή δεδομένων από μια συσκευή monitoring απαιτεί δικτυακή σύνδεση. Εάν η σύνδεση χαθεί, τα εργαλεία agentless monitoring μπορούν να αναφέρουν μια συσκευή ως "εκτός λειτουργίας" και ίσως ακόμη και να καταγράψουν αρχεία καταγραφής που να εξηγούν τα πράγματα μετά το συμβάν. Η ύπαρξη ενός agent στη συσκευή αυτή, επιτρέπει να παρακολουθεί και να αποθηκεύει μετρήσεις ανεξάρτητα από το αν υπάρχει συνδεσιμότητα με το δίκτυο [16].

3.3.4 Συμπεράσματα

Για το IT, το ερώτημα καταλήγει στο ποια είναι η πιο αποτελεσματική μέθοδος για την επίτευξη των επιχειρηματικών στόχων κάθε εταιρίας. Η απάντηση θα είναι διαφορετική για τον καθένα, αλλά το σκεπτικό "agentless εξ ορισμού" είναι ένα χρήσιμο πλαίσιο για την αιτιολόγηση της πρόσθετης εργασίας που συνεπάγεται μια προσέγγιση βασισμένη σε agent-based. Κατά τα άλλα, η παρακολούθηση με βάση τους agent απαιτεί περισσότερη εργασία από το IT. Σημαίνει επίσης ότι οι συσκευές θα εκτελούν ένα πρόσθετο λογισμικό που καταναλώνει κάποιους πόρους και μπορεί να απαιτεί επιδιόρθωση. Αλλά ένας agent μπορεί να παρέχει συγκεκριμένα δεδομένα μιας συσκευής και να προσφέρει λειτουργίες που δεν προσφέρουν τα τυπικά πρωτόκολλα. Οι agent βοηθούν επίσης στην αντιμετώπιση προβλημάτων όταν η συνδεσιμότητα είναι περιορισμένη. Τέλος, η παρακολούθηση με agentless agent και η παρακολούθηση με agent-based agent δεν χρειάζεται να είναι αμοιβαία αποκλεισμένες. Η χρήση agentless agent για το μεγαλύτερο μέρος του δικτυακού συστήματος και η εγκατάσταση των agent-based agent σε κρίσιμα συστήματα ανάλογα με τις ανάγκες είναι συχνά η πιο πρακτική προσέγγιση [16].

3.4 Ανακάλυψη χαμηλού επιπέδου

Η ανακάλυψη χαμηλού επιπέδου (low-level discovery) χρησιμοποιείται για την παρακολούθηση συστημάτων αρχείων και διασυνδέσεων δικτύου χωρίς να χρειάζεται να δημιουργείται και να προστίθεται χειροκίνητα κάθε στοιχείο. Η ανακάλυψη χαμηλού επιπέδου δίνει την δυνατότητα να προστίθενται και αφαιρούνται αυτόματα στοιχεία. Δημιουργεί επίσης αυτόματα triggers και γραφήματα για συστήματα αρχείων, διασυνδέσεις δικτύου και πίνακες SNMP [17].

Πριν από την ευρεία χρήση της ανακάλυψης χαμηλού επιπέδου, χρησιμοποιούνταν τα Templates. Ωστόσο, η δημιουργία ενός template είναι μια απαιτητική διαδικασία. Κάθε template απαιτεί τη χειροκίνητη δημιουργία του trigger για κάθε θύρα ή λογικό δίσκο. Παράλληλα, πρέπει να καθοριστεί αναλυτικά το περιεχόμενο του κάθε trigger. Για παράδειγμα, αν ένα δίκτυο διαθέτει ένα switch με 24 θύρες, το template θα πρέπει να δημιουργηθεί χρησιμοποιώντας triggers SNMPv2 για την παρακολούθηση της κατάστασης αυτών των θυρών. Για την κάθε θύρα χρειάζεται να δημιουργηθούν τα εκάστοτε triggers και τα αντίστοιχα γραφήματα το οποίο είναι πολύ χρονοβόρα διαδικασία και για τις 24 θύρες του switch.

Αντί να δημιουργούνται χειροκίνητα templates για κάθε θύρα, η ανακάλυψη χαμηλού επιπέδου επιτρέπει τη δημιουργία πρωτοτύπων των στοιχείων και triggers μόνο μία φορά. Μετά από αυτό η ανακάλυψη χαμηλού επιπέδου ανακαλύπτει αυτόματα τις θύρες, το σύστημα αρχείων και τους πίνακες

SNMP. Θα μπορούσαν να δημιουργηθούν πολλά διαφορετικά triggers που να ανταποκρίνονται σε διαφορετικές ανάγκες [17].

Τέλος, η ανακάλυψη χαμηλού επιπέδου χρησιμοποιείται για την παρακολούθηση του φόρτου εργασίας των πυρήνων της CPU και των φυσικών δίσκων. Γενικά, η ανακάλυψη χαμηλού επιπέδου συμβάλλει στη βελτιστοποίηση της παρακολούθησης του δικτύου και παρέχει ειδοποιήσεις και γραφήματα για την κατάσταση των θυρών, των λογικών δίσκων και άλλων παραμέτρων.

3.5 Αυτόματη ανακάλυψη

Μία από τις προκλήσεις που αντιμετωπίζει ένας διαχειριστής δικτύου είναι να διατηρεί το σύστημα διαχείρισης δικτύου ενημερωμένο με όλες τις δυναμικές αλλαγές. Σε πολλά περιβάλλοντα δικτύων, προστίθενται νέες συσκευές σε εβδομαδιαία ή ακόμη και καθημερινή βάση. Για να παρακολουθείται συνεχώς το μεταβαλλόμενο περιβάλλον χρησιμοποιείται η αυτόματη ανακάλυψη. Η αυτόματη ανακάλυψη παρέχει την δυνατότητα της αυτοματοποίησης της λειτουργίας προσθήκης νέας συσκευής στο monitoring σύστημα. Επίσης παρέχει την ανίχνευση δικτυακών διεπαφών, θυρών και συστημάτων αρχείων [17].

Η αυτόματη ανακάλυψη θα μπορούσε να χρησιμοποιηθεί για να εντοπιστεί η τρέχουσα κατάσταση του δικτύου όπως ποιες συσκευές και υπηρεσίες βρίσκονται επί του παρόντος στο δίκτυο. Επιπλέον, συμβάλλει σε θέματα ασφάλειας, επαληθεύοντας ποιες θύρες είναι ενεργοποιημένες. Παρόλο που η αυτόματη ανακάλυψη παίζει καθοριστικό ρόλο στην παρακολούθηση του δικτύου, ορισμένα διαθέσιμα εργαλεία στην αγορά δεν προσφέρουν αυτή τη δυνατότητα.

3.6 Λογική Ομαδοποίηση

Σε εκτεταμένα δίκτυα τα οποία έχουν πολλές συσκευές, η παρακολούθηση και η επίλυση προβλημάτων μπορεί να είναι ιδιαίτερα δύσκολη. Η λογική ομαδοποίηση επιτρέπει τη συγκέντρωση συσκευών του ίδιου τύπου, διευκολύνοντας σημαντικά την παρακολούθηση μεγάλων δικτύων. Με αυτή τη μέθοδο, συσκευές παρόμοιου είδους δικτύου ομαδοποιούνται, και για κάθε ομάδα μπορούν να καθοριστούν συγκεκριμένοι παράμετροι παρακολούθησης και ενέργειες σε περίπτωση αποτυχίας. Επίσης, η λογική ομαδοποίηση επιτρέπει την εφαρμογή κοινών ρυθμίσεων σε όλα τα μέλη της ομάδας. Για μεγάλα δίκτυα, είναι δυνατό να δημιουργηθούν ένθετες ομάδες, δηλαδή ομάδες εντός άλλων ομάδων. Αυτό καθιστά τη διαχείριση των συσκευών δικτύου σε ένα εκτεταμένο δίκτυο πολύ πιο εύκολη.

Κεφάλαιο 4ο: Zabbix

4.1 Επισκόπηση του Zabbix

Το Zabbix, που δημιουργήθηκε από τον Alexei Vladishev και τώρα υποστηρίζεται ενεργά από την εταιρεία Zabbix SIA, είναι μια κατανεμημένη λύση παρακολούθησης ανοιχτού κώδικα για επιχειρήσεις. Το λογισμικό αυτό παρακολουθεί μια πληθώρα παραμέτρων δικτύου, καθώς και την κατάσταση και την ακεραιότητα των διακομιστών. Το Zabbix διαθέτει έναν ευέλικτο μηχανισμό ειδοποιήσεων, επιτρέποντας στους χρήστες να ρυθμίζουν ειδοποιήσεις μέσω ηλεκτρονικού ταχυδρομείου για σχεδόν οποιοδήποτε συμβάν, διευκολύνοντας την άμεση ανταπόκριση σε προβλήματα. Επιπλέον, παρέχει εξαιρετικές δυνατότητες αναφοράς και οπτικοποίησης δεδομένων με βάση τα συλλεγμένα δεδομένα. [17].

Το Zabbix υποστηρίζει σαν λειτουργίες λήψεις δεδομένων τόσο το polling όσο και το trapping. Όλες οι αναφορές και τα στατιστικά του Zabbix, όπως και οι ρυθμίσεις, είναι προσβάσιμα μέσω ενός web-based περιβάλλοντος. Αυτό επιτρέπει την παρακολούθηση της κατάστασης του δικτύου και των διακομιστών από οπουδήποτε. Με τη σωστή ρύθμιση, το Zabbix μπορεί να είναι ουσιαστικό εργαλείο για την παρακολούθηση της IT υποδομής, ανεξάρτητα από το μέγεθος του οργανισμού, από μικρές επιχειρήσεις με λίγους διακομιστές έως μεγάλες εταιρείες με πολλούς. Επιπλέον, το Zabbix είναι δωρεάν λογισμικό, γραμμένο και διανεμόμενο υπό την έκδοση 2 της Γενικής Δημόσιας Άδειας (GPL). Αυτό σημαίνει ότι ο πηγαίος κώδικάς του είναι ελεύθερα διαθέσιμος στο κοινό [17].

4.2 Πλεονεκτήματα του Zabbix

Το Zabbix θεωρείτε ένα εργαλείο το οποίο είναι αρκετά αξιόπιστο για την παρακολούθηση του δικτύου. Η παρακολούθηση μπορεί να γίνει είτε με agent-based είτε με agentless agent καθώς και πολλές φορές γίνεται με συνδυασμό αυτών, αναλόγως με τις ανάγκες της παρακολούθησης. Όλα τα προαναφερθέντα καθιστούν το Zabbix ένα αλάνθαστο εργαλείο παρακολούθησης δικτύου που ικανοποιεί πλήρως τις απαιτήσεις οποιουδήποτε μεγέθους δικτύου. Το Zabbix είναι ένα αξιόπιστο εργαλείο παρακολούθησης δικτύων. Όταν ειδοποιεί τον χρήστη για δυσλειτουργίες, αυτό σημαίνει ότι το πρόβλημα υπάρχει με βεβαιότητα. Οι ίδιες αρχές αξιοπιστίας ισχύουν και για την ανάκτηση δεδομένων και την οπτικοποίηση. Ένα από τα κύρια πλεονεκτήματα του Zabbix είναι η επεκτασιμότητά του, καθιστώντας το κατάλληλο για δίκτυα οποιουδήποτε μεγέθους. Η αρχή της επεκτασιμότητας αφορά τόσο την απόδοση όσο και τη χρηστικότητα του frontend. Επιπλέον, οι δυνατότητες του Zabbix δεν περιορίζονται μόνο στον τομέα της πληροφορικής. [18].

Το Zabbix ως εργαλείο παρακολούθησης δικτύου θα μπορούσε να συγκριθεί με τον εγκέφαλο. Λαμβάνει τις πληροφορίες που ρέουν όπως είσοδο από αισθητήρες και ακέραιους αριθμούς δεδομένων. Τα triggers αναλύουν όλα αυτά τα δεδομένα και δημιουργούν μια έξοδο που είναι ανάλογη με τα δεδομένα που λαμβάνουν. Θα μπορούσε να είναι η διεύθυνση mac μιας συσκευής, θερμοκρασία της CPU ή ακόμη και μια ειδοποίηση ή μια εντολή για την εκκίνηση ενός scirpt [18].

4.3 Εγκατάσταση του Zabbix

4.3.1 Διεργασία πριν την εγκατάσταση του Zabbix

Η εγκατάσταση του Zabbix 6.4 έχει γίνει σε virtual machine το οποίο είναι Debian 12 και δημιουργήθηκε μέσω Proxmox. Προτού ξεκινήσει η εγκατάσταση του Zabbix, εφαρμόστηκαν οι παρακάτω εντολές:

- `sudo apt-get update`
- `sudo apt-get upgrade`
- `sudo apt-get install mariadb-server -y`

Η εντολή apt-get update ενημερώνει τη λίστα των διαθέσιμων πακέτων από τα repository που έχουν οριστεί στο σύστημα. Ουσιαστικά, ελέγχει για νέες εκδόσεις των πακέτων, αλλά δεν εγκαθιστά κάτι νέο.

Η εντολή apt-get upgrade εφαρμόζει τις διαθέσιμες αναβαθμίσεις για τα εγκατεστημένα πακέτα. Ουσιαστικά, αναβαθμίζει τα πακέτα στις τελευταίες εκδόσεις που είναι διαθέσιμες στο repository. Αυτό περιλαμβάνει τόσο τα κανονικά πακέτα λογισμικού όσο και το λειτουργικό σύστημα και άλλα βασικά πακέτα.

Η εντολή apt-get install mariadb-server -y χρησιμοποιείται για να εγκαταστήσει τον διακομιστή βάσης δεδομένων MariaDB η οποία είναι αναγκαία να υπάρχει ώστε να μπορέσει να εγκατασταθεί το Zabbix σύστημα.

4.3.2 Εγκατάσταση Zabbix Server

Για την εγκατάσταση του Zabbix Server έχουν εκτελεστεί οι παρακάτω εντολές στο command line του Debian Server. Οι εντολές αυτές μπορούν να βρεθούν στο επίσημο έγγραφο του Zabbix.

- `wget https://repo.zabbix.com/zabbix/6.4/debian/pool/main/z/zabbix-release/zabbix-release_6.4-1+debian12_all.deb`
- `dpkg -i zabbix-release_6.4-1+debian12_all.deb`
- `apt update`

Η πρώτη εντολή χρησιμοποιεί το wget για να κατεβάσει το πακέτο zabbix-release_6.4-1+debian12_all.deb από τη διεύθυνση https://repo.zabbix.com/zabbix/6.4/debian/pool/main/z/zabbix-release/. Αυτό το πακέτο είναι ένα αρχείο που περιέχει πληροφορίες σχετικά με το repository του Zabbix για την έκδοση 6.4.

Με την εντολή «dpkg -i zabbix-release_6.4-1+debian12_all.deb» χρησιμοποιείται το dpkg για να εγκαταστήσει το πακέτο zabbix-release_6.4-1+debian12_all.deb που μόλις κατεβάσαμε. Το dpkg είναι ο διαχειριστής πακέτων σε συστήματα Debian-based, όπως το Debian και το Ubuntu.

Τέλος με την εντολή apt update ενημερώνεται η λίστα των διαθέσιμων πακέτων από το repository που έχει οριστεί στο σύστημα καθώς και των νέων πακέτων του Zabbix που προστέθηκαν από την εντολή dpkg.

- `apt install zabbix-server-mysql zabbix-frontend-php zabbix-apache-conf zabbix-sql-scripts zabbix-agent`

Με την εκτέλεση αυτής της εντολής, τα πακέτα που χρειάζονται από το Zabbix θα εγκατασταθούν στο σύστημά, προετοιμάζοντας το για την εγκατάσταση και τη ρύθμιση του Zabbix.

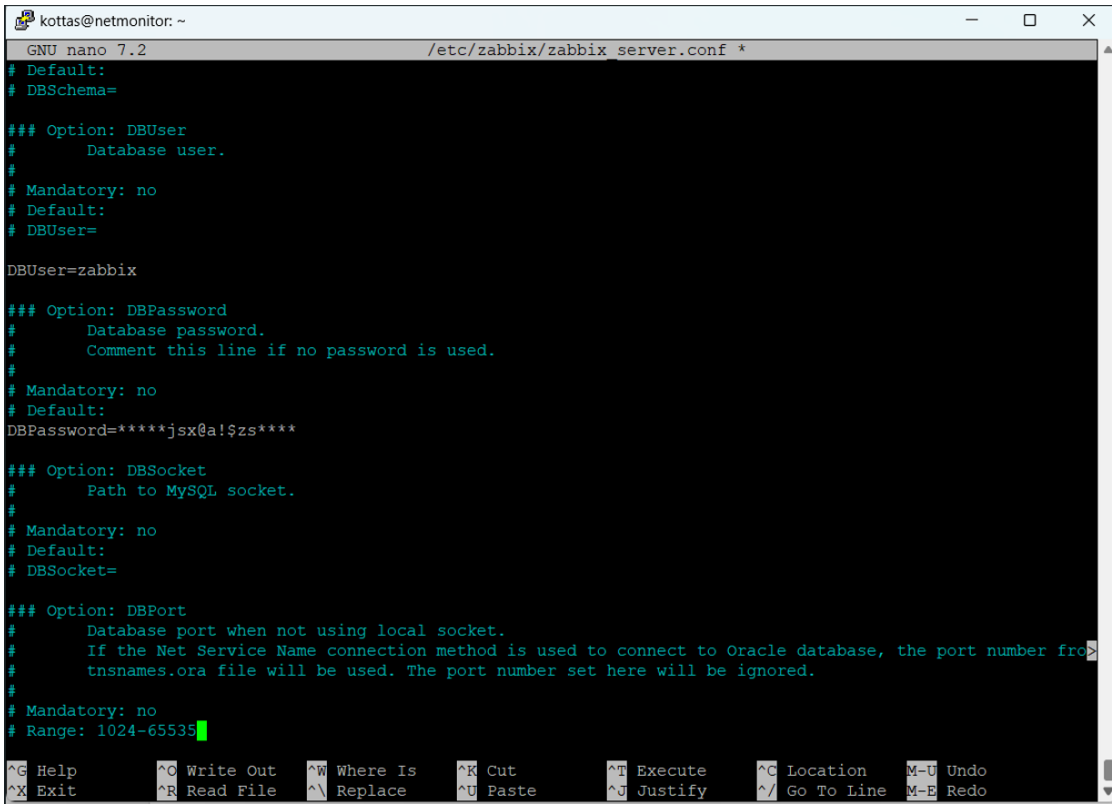
Στο επόμενο βήμα δημιουργείτε η βάση του Zabbix αλλά και χρήστης που έχει όλα τα δικαιώματα για αυτή την βάση. Συνεπώς γίνεται σύνδεση στη mariadb με την εντολή «mysql» και έπειτα εφαρμόζονται οι παρακάτω εντολές.

- `mysql> create database zabbix character set utf8mb4 collate utf8mb4_bin;`
- `mysql> create user zabbix@localhost identified by 'password';`
- `mysql> grant all privileges on zabbix.* to zabbix@localhost;`
- `mysql> quit;`

Έπειτα με την παρακάτω εντολή δημιουργείτε και διαμορφώνονται οι πίνακες και το σχήμα βάσης δεδομένων που απαιτούνται για την εγκατάσταση του Zabbix.

- `zcat /usr/share/zabbix-sql-scripts/mysql/server.sql.gz | mysql --default-character-set=utf8mb4 -uzabbix -p zabbix`

Τέλος χρειάζεται να γίνει edit το αρχείο /etc/zabbix/zabbix_server.conf με τον κωδικό της βάσης όπως φαίνεται και στην Εικόνα 2.

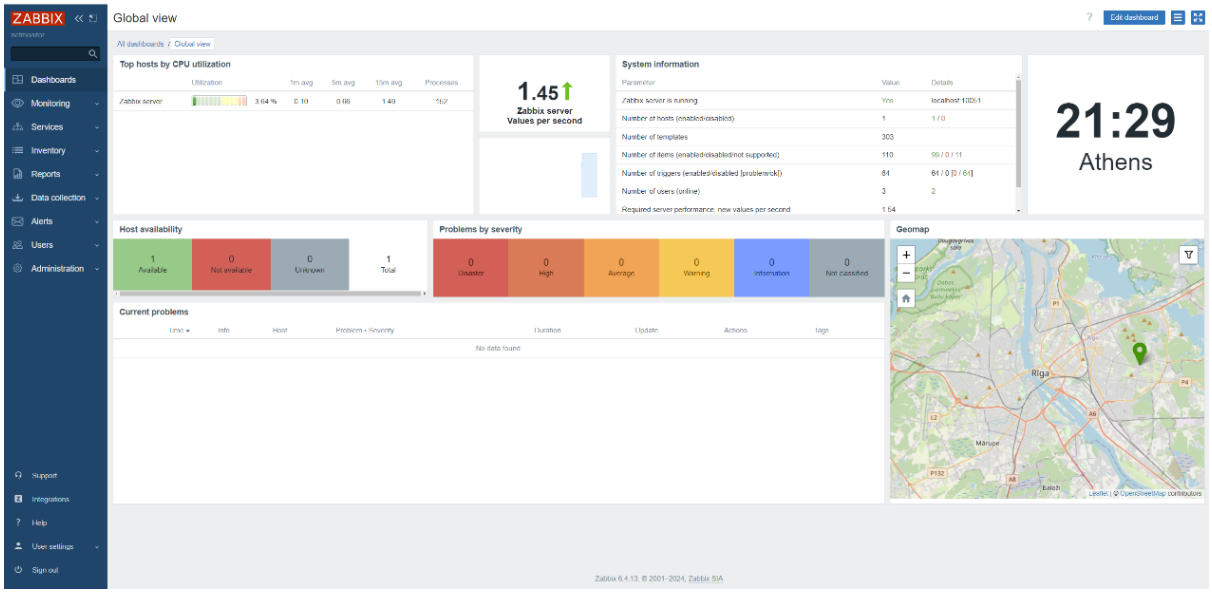


Εικόνα 2. Διαμόρφωση αρχείου Zabbix_server.conf

Κατόπιν αυτών των ενεργειών χρειάζονται restart οι υπηρεσίες ώστε να πάρουν τις αλλαγές που έγιναν, το οποίο γίνεται με την παρακάτω εντολή.

- `systemctl restart zabbix-server zabbix-agent apache2`

Στο τελευταίο βήμα της εγκατάστασης χρειάζεται η μετάβαση στο url http://host/zabbix και έπειτα η συμπλήρωση των πεδίων που εμφανίζονται στον browser. Παρακάτω στην φωτογραφία φαίνεται το Zabbix UI κατά την πρώτη σύνδεση.



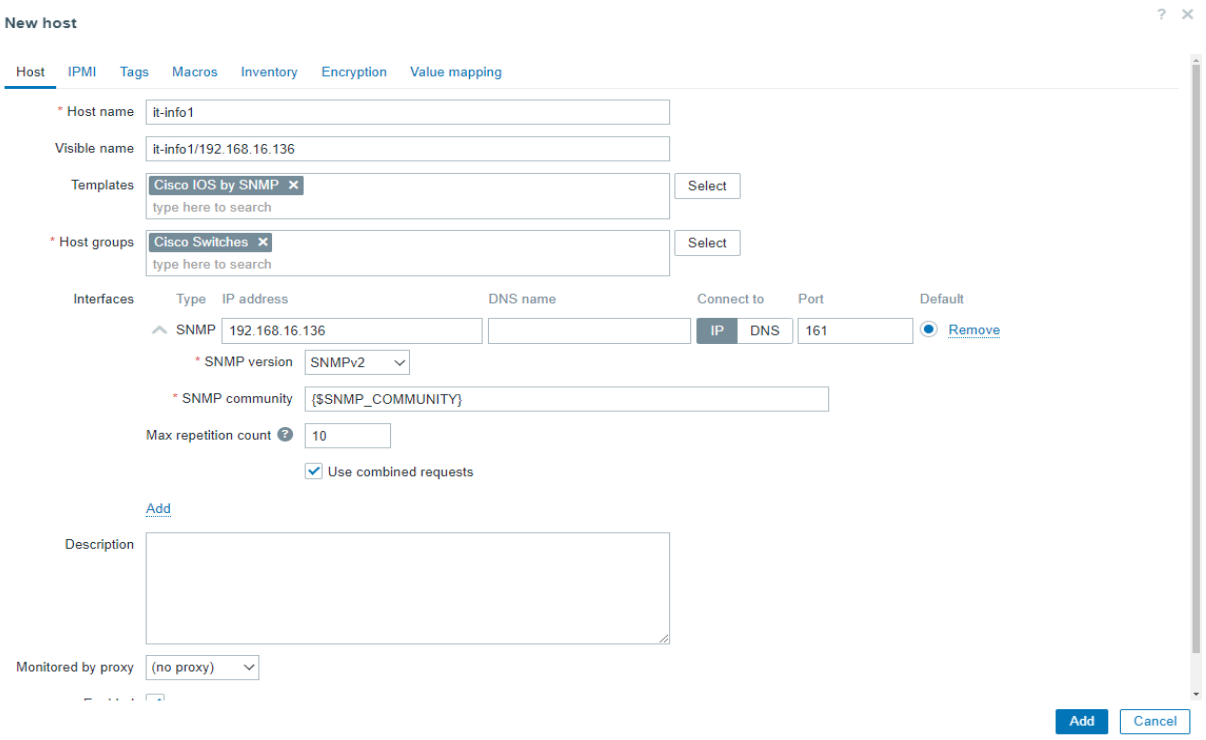
Εικόνα 3. Zabbix Main Page

4.4 Δημιουργία Host

Το πρώτο βήμα είναι η δημιουργία ενός host. Πρόκειται για μια συσκευή ή υπηρεσία δικτύου που πρόκειται να παρακολουθείται από τον Zabbix Server. Σε αυτό το κεφάλαιο θα δημιουργηθούν δύο hosts όπου στον πρώτο παράδειγμα θα παίρνει τα δεδομένα μέσω SNMPv2 ενώ στον δεύτερο παράδειγμα θα τα παίρνει τα δεδομένα μέσω Zabbix agent. Για την δημιουργία του host από το Zabbix UI χρειάζεται να γίνει μετάβαση σε “Data collection” -> “Hosts” -> “Create host”.

4.4.1 Δημιουργία Host με SNMPv2

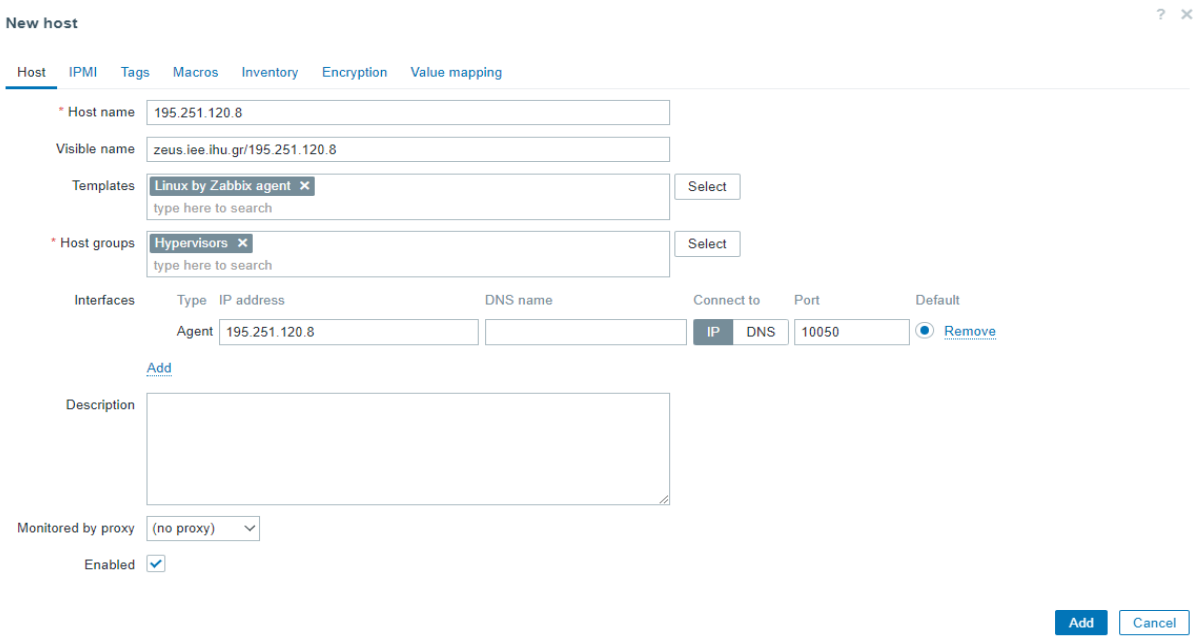
Κατά την δημιουργία του host χρειάζεται η εισαγωγή ονόματος το οποίο πρέπει να είναι αντιπροσωπευτικό καθώς βοηθάει στον εντοπισμό της συσκευής και στο να γίνεται αντιληπτό μόνο από το όνομα, από ποια συσκευή έχει δημιουργηθεί κάποιο alert. Έπειτα στην Εικόνα 4. απεικονίζονται τα βήματα της δημιουργίας ενός νέου host. Έχει δοθεί το όνομα it-info1 που αντιπροσωπεύει το switch. Σαν “Visible name” έχει οριστεί το it-info1/192.168.16.136 το οποίο θα είναι το ορατό όνομα του host καθώς βοηθάει πολλές φορές και η IP στον εντοπισμό των συσκευών. Για την αυτόματη δημιουργία των items, triggers και γραφημάτων χρησιμοποιήθηκε το Cisco SNMP template καθώς η συγκεκριμένη συσκευή είναι του κατασκευαστή της Cisco. Έπειτα χρειάζεται η εισαγωγή του host σε κάποιο host group. Στην συγκεκριμένη περίπτωση προστέθηκε στο Cisco Switches όπου είναι ομαδοποιημένα όλα τα Cisco switches που γίνονται monitoring. Τέλος στο “interfaces” χρειάζεται να επιλεχθεί ο τρόπος που θα λαμβάνονται τα δεδομένα καθώς και την IP της συσκευής. Στην συγκεκριμένη περίπτωση το πρωτόκολλο που χρησιμοποιείτε για την λήψη των δεδομένων είναι το SNMPv2 και η IP της συσκευής είναι 192.168.16.136 καθώς και το SNMP community που χρησιμοποιείτε για την ταυτοποίηση της λήψης των δεδομένων έχει οριστεί σαν μεταβλητή στην καρτέλα Macros.



Εικόνα 4. SNMP host creation

4.4.2 Δημιουργία Host με Zabbix Agent

Για την δημιουργία Host που θα λαμβάνει τα δεδομένα μέσω Zabbix Agent η διαδικασία είναι ίδια όπως προαναφέρθηκε και στο 4.4.1 με την διαφορά ότι στο κομμάτι του “Interfaces” επιλέγεται Agent όπως φαίνεται και στην Εικόνα 5. Ποιο συγκεκριμένα έχει οριστεί ως ορατό όνομα το “ zeus.iee.ihu.gr/195.251.120.8”. Για την δημιουργία των items, triggers και γραφημάτων χρησιμοποιήθηκε το “Linux by Zabbix agent” template και έχει προστεθεί στο Hypervisors group. Τέλος αφότου στο interface type έχει επιλεχθεί το «Agent», χρειάζεται να εισαχθεί η IP του host που θα λαμβάνει τα δεδομένα. Στην συγκεκριμένη περίπτωση είναι η 195.251.120.8.



Εικόνα 5. Agent host creation

4.4.3 Zabbix templates

Ένα template είναι ένα σύνολο οντοτήτων που μπορούν να εφαρμοστούν σε οποιονδήποτε host. Ένα template μπορεί να αποτελείται από items, triggers και γραφήματα. Στο Zabbix από προεπιλογή υπάρχει ένας κατάλογος template. Ωστόσο, περισσότερα template μπορούν να μεταφορτωθούν από την επίσημη ιστοσελίδα του Zabbix ή ακόμη και να δημιουργηθούν από έναν χρήστη. Υπάρχουν δύο βασικά πλεονεκτήματα που προκύπτουν από τα templates. Το πρώτο είναι ότι, τα templates μπορούν να περιέχουν εξ ορισμού items, triggers και γράφημα. Επίσης επιτρέπουν την προσθήκη περισσότερων items ή να επεξεργαστούν τα ήδη υπάρχοντα triggers.

Το δεύτερο πλεονέκτημα είναι ότι ένα template μπορεί να χρησιμοποιηθεί για απεριόριστο αριθμό hosts. Επιπλέον, εάν υπάρχει ανάγκη να γίνουν κάποιες αλλαγές στην παρακολούθηση, οι αλλαγές πρέπει να γίνουν μόνο σε ένα template. Οι hosts που χρησιμοποιούν αυτό το πρότυπο ενημερώνονται αυτόματα. Ως αποτέλεσμα, τα templates φέρνουν μεγάλη ευελιξία στην παρακολούθηση δικτύου. Για τους προαναφερθέντες λόγους στην δημιουργία των hosts έχουν χρησιμοποιηθεί έτοιμα Zabbix templates ώστε να μην χρειάζεται για τον κάθε host να δημιουργούνται ξεχωριστά items, triggers και γραφήματα.

Παρακάτω παρουσιάζονται ενδεικτικές φωτογραφίες με τα γραφήματα που δημιουργήθηκαν κατά την εισαγωγή του “Linux by Zabbix agent” template στον host zeus.iee.ihu.gr

Εικόνα 6. Storage Usage Pie

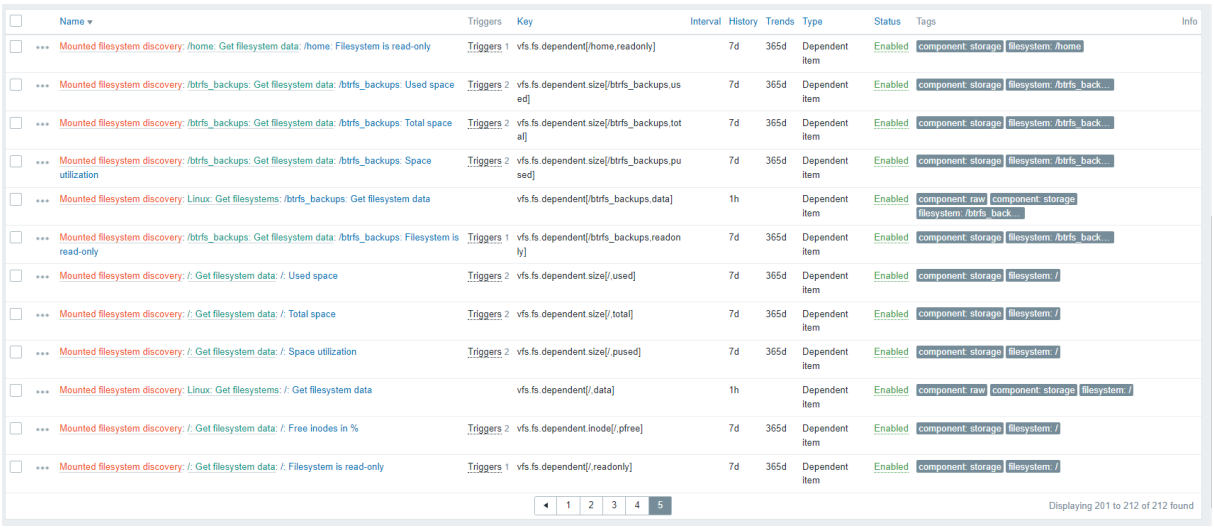
Εικόνα 7. Ram Usage Graph

4.4.4 Items

Τα items παίζουν καθοριστικό ρόλο στην λειτουργία της παρακολούθησης καθώς τα items χρειάζονται διότι χρησιμοποιούνται για τη συλλογή δεδομένων. Μόλις διαμορφωθεί ένας host, πρέπει να προστεθούν items ώστε να λαμβάνει πραγματικά δεδομένα. Ένας τρόπος για να προστεθούν γρήγορα πολλά items είναι να εφαρμοστεί κάποιο template στον host. Ωστόσο, για τη βελτιστοποιημένη απόδοση του συστήματος, μπορεί να χρειαστεί να ρυθμιστεί λεπτομερώς το template ώστε να υπάρχουν όσα items χρειάζονται αλλά και με την συχνότητα που επιθυμείτε να παρακολουθούνται [17].

Για να καθοριστεί το είδος των δεδομένων που θα συλλέγονται από έναν host, χρησιμοποιείτε το item key. Για παράδειγμα, ένα στοιχείο με όνομα κλειδιού system.cpu.load θα συλλέγει δεδομένα φόρτου επεξεργαστή, ενώ ένα στοιχείο με όνομα key net.if.in θα συλλέγει πληροφορίες εισερχόμενης κυκλοφορίας. Πρόσθετες παράμετροι μπορούν να καθοριστούν σε αγκύλες μετά το όνομα του key. Για παράδειγμα, το system.cpu.load[avg5] θα επιστρέψει το μέσο όρο του φορτίου του επεξεργαστή για τα

τελευταία 5 λεπτά, ενώ το net.if.in[eth0] θα εμφανίσει την εισερχόμενη κυκλοφορία στη διασύνδεση "eth0" [17]. Στην Εικόνα 8. παρουσιάζονται ενδεικτικά items από κάποιον host του συστήματος.



Εικόνα 8 items

4.4.5 Triggers

Τα triggers είναι λογικές εκφράσεις που "αξιολογούν" τα δεδομένα που συλλέγονται από τα items και αντιπροσωπεύουν την τρέχουσα κατάσταση του συστήματος. Ενώ τα items χρησιμοποιούνται για τη συλλογή δεδομένων του συστήματος, είναι εξαιρετικά ανέφικτο να παρακολουθούνται αυτά τα δεδομένα όλη την ώρα περιμένοντας μια κατάσταση που είναι ανησυχητική ή χρήζει προσοχής [17].

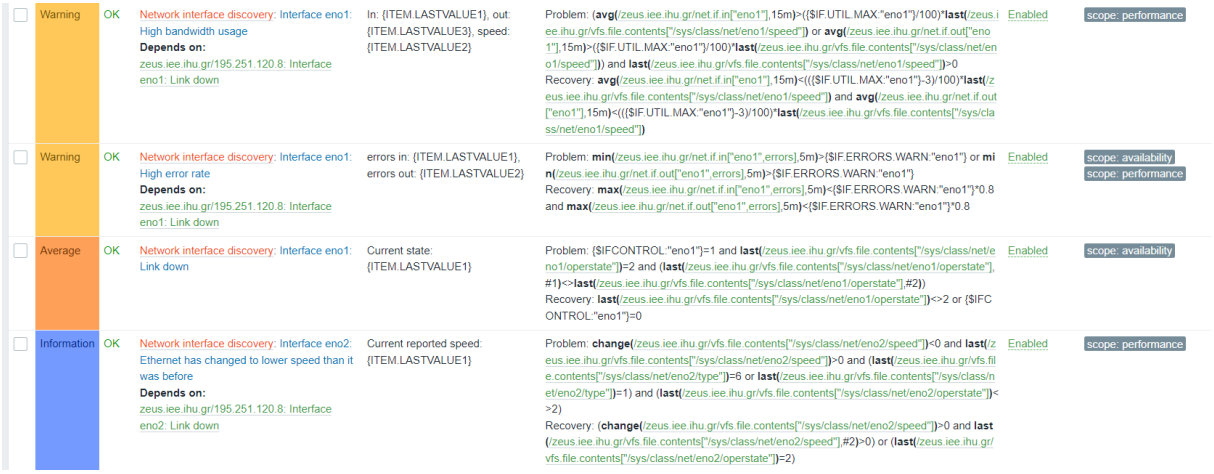
Οι εκφράσεις των triggers επιτρέπουν τον ορισμό ενός ορίου για το ποια κατάσταση δεδομένων είναι "αποδεκτή". Επομένως, εάν τα εισερχόμενα δεδομένα υπερβούν την αποδεκτή κατάσταση, ένα trigger "πυροδοτείτε" ή αλλάζει την κατάστασή του σε ΠΡΟΒΛΗΜΑ [17]. Οι καταστάσεις που μπορεί να έχει ένα trigger φαίνονται στην Εικόνα 9.

Status	Description
OK	This is a normal trigger status.
Problem	Something has happened. For example, the processor load is too high.
Unknown	The trigger value cannot be calculated. See Unknown status.

Εικόνα 9 Trigger status

Στην παρακάτω Εικόνα 10. Και Εικόνα 11. απεικονίζονται ενδεικτικά κάποια από τα triggers που εφαρμόζονται στον host.

Εικόνα 10 Storage Triggers



	Warning	OK	Network interface discovery: Interface eno1: High bandwidth usage **Depends on:** zeus.iee.ihu.gr/195.251.120.8: Interface eno1: Link down	In: {ITEM.LASTVALUE1}, out: {ITEM.LASTVALUE3}, speed: {ITEM.LASTVALUE2}	Problem: (**avg**(/zeus.iee.ihu.gr/net.if.in["eno1"],15m)>({$IF.UTIL.MAX:"eno1"}/100)***last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno1/speed"]) or **avg**(/zeus.iee.ihu.gr/net.if.out["eno1"],15m)>({$IF.UTIL.MAX:"eno1"}/100)***last**(zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno1/speed"])) and **last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno1/speed"])>0 Recovery: **avg**(/zeus.iee.ihu.gr/net.if.in["eno1"],15m)<(({$IF.UTIL.MAX:"eno1"}-3)/100)***last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno1/speed"]) and **avg**(/zeus.iee.ihu.gr/net.if.out["eno1"],15m)<(({$IF.UTIL.MAX:"eno1"}-3)/100)***last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno1/speed"])	Enabled	scope: performance
	Warning	OK	Network interface discovery: Interface eno1: High error rate **Depends on:** zeus.iee.ihu.gr/195.251.120.8: Interface eno1: Link down	errors in: {ITEM.LASTVALUE1}, errors out: {ITEM.LASTVALUE2}	Problem: **min**(/zeus.iee.ihu.gr/net.if.in["eno1",errors],5m)>{$IF.ERRORS.WARN:"eno1"} or **min**(/zeus.iee.ihu.gr/net.if.out["eno1",errors],5m)>{$IF.ERRORS.WARN:"eno1"} Recovery: **max**(/zeus.iee.ihu.gr/net.if.in["eno1",errors],5m)<{$IF.ERRORS.WARN:"eno1"}*0.8 and **max**(/zeus.iee.ihu.gr/net.if.out["eno1",errors],5m)<{$IF.ERRORS.WARN:"eno1"}*0.8	Enabled	scope: availability scope: performance
	Average	OK	Network interface discovery: Interface eno1: Link down	Current state: {ITEM.LASTVALUE1}	Problem: {$IFCONTROL:"eno1"}=1 and **last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno1/operstate"])=2 and (**last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno1/operstate"],#1)<>**last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno1/operstate"],#2)) Recovery: **last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno1/operstate"])<>2 or {$IFCONTROL:"eno1"}=0	Enabled	scope: availability
	Information	OK	Network interface discovery: Interface eno2: Ethernet has changed to lower speed than it was before **Depends on:** zeus.iee.ihu.gr/195.251.120.8: Interface eno2: Link down	Current reported speed: {ITEM.LASTVALUE1}	Problem: **change**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno2/speed"])<0 and **last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno2/speed"])>0 and (**last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno2/type"])=6 or **last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno2/type"])=1) and (**last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno2/operstate"])<>2) Recovery: (**change**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno2/speed"])>0 and **last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno2/speed",#2)>0) or (**last**(/zeus.iee.ihu.gr/vfs.file.contents["/sys/class/net/eno2/operstate"])=2)	Enabled	scope: performance

Εικόνα 11 Network Triggers

Τέλος στην Εικόνα 12. φαίνεται πως παρουσιάζονται τα triggers όταν εμφανίζεται το πρόβλημα στο dashboard του Zabbix. Από τα αριστερά προς τα δεξιά φαίνεται η χρονολογία που εμφανίστηκε το trigger, το status του trigger (problem/resolved), το όνομα του host, το είδος προβλήματος και την κρισιμότητά του, κάποια δεδομένα για το trigger, και τέλος την διάρκεια που υπάρχει το trigger.

Time ▾	Recovery time	Status	Info	Host	Problem • Severity	Operational data	Duration	Update	Actions
2024-05-22 15:55:46		PROBLEM		lt-infoSK/[redacted]	Processor: High memory utilization (>90% for 5m)	100 %	1d 6h 26m	Update	1
Yesterday									
2024-05-14 15:11:51		PROBLEM		lt-info302/[redacted]	Cisco IOS: Unavailable by ICMP ping	Down (0)	9d 7h 10m	Update	1
2024-05-14 15:11:51		PROBLEM		lt-info301/[redacted]	Cisco IOS: Unavailable by ICMP ping	Down (0)	9d 7h 10m	Update	1
2024-05-14 15:09:42		PROBLEM		sw-lt2/[redacted]	Interface Gi0/9(lt-info301): Link down	Current state: down (2)	9d 7h 12m	Update	1
2024-05-14 15:09:37		PROBLEM		sw-lt1/[redacted]	Interface Gi0/9(lt-info301): Link down	Current state: down (2)	9d 7h 12m	Update	1
2024-05-13 09:32:48		PROBLEM		lt-info210a/[redacted]	I/O: High memory utilization (>90% for 5m)	93.5968 %	10d 12h 49m	Update	1
2024-05-06 23:26:48		PROBLEM		test alerting	Linux: No SNMP data collection	Current state: not available (0)	16d 22h 55m	Update	1
2024-05-02 21:56:12		PROBLEM		hermes/[redacted]	MySQL: Service is down	*UNKNOWN*	21d 26m	Update	1

Εικόνα 12 dashboard triggers

4.5 Agentless και Agent client

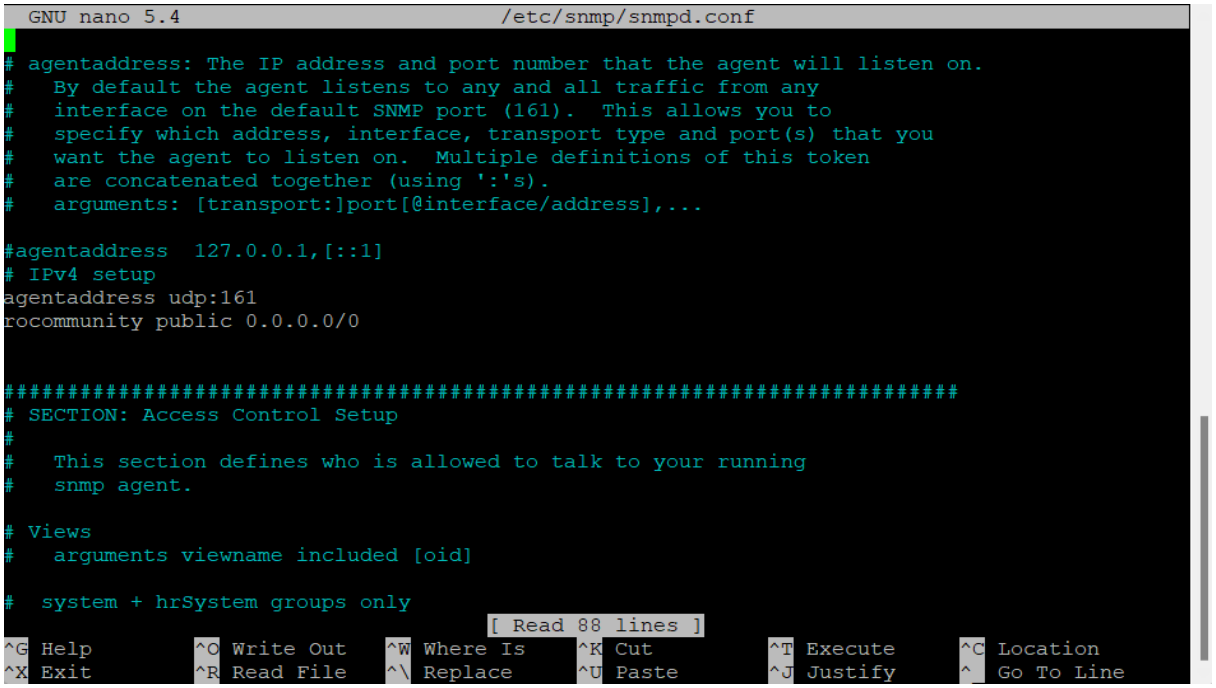
Μέχρι στιγμής έχει οριστεί από την πλευρά του Zabbix Server να τραβάει δεδομένα από συγκεκριμένους hosts όταν αυτοί δημιουργήθηκαν. Στην μια περίπτωση υλοποιήθηκε με SNMPv2 το οποίο είναι Agentless τρόπος και στην δεύτερη περίπτωση με Zabbix agent ο οποίος είναι agent based. Παρακάτω θα παρουσιαστούν τα βήματα που ακολουθήθηκαν από τον κάθε host ώστε να μπορεί να στέλνει τα δεδομένα και αντίστοιχα ο Zabbix Server να μπορεί να τα λαμβάνει.

4.5.1 Agentless SNMP client

Σε αυτή την παράγραφο θα παρουσιαστούν τα βήματα που χρειάζεται να ακολουθηθούν ώστε να εγκατασταθεί το SNMP πρωτόκολλο και να επιτρέπει να τραβάει τα δεδομένα ο Zabbix Server. Στην συγκεκριμένη περίπτωση τα βήματα αφορούν Linux Debian σύστημα. Ανάλογα με το είδος της συσκευής (Switch,PDU,Linux,Windows) ο τρόπος αυτός διαφέρει. Για την εγκατάσταση του SNMP στο Debian χρειάζεται να τρέξει η εντολή «apt install snmpd». Έπειτα πρέπει να μορφοποιηθεί το αρχείο «/etc/snmp/snmpd.conf» που δημιουργήθηκε κατά την εγκατάσταση του SNMP και να προστεθούν οι παρακάτω γραμμές όπως φαίνεται και στην Εικόνα 13.

- `agentaddress udp:161`
- `rocommunity public 0.0.0.0/0`

Με τις παραπάνω εντολές ενεργοποιείται η πόρτα 161 για το πρωτόκολλο udp και επιτρέπει να τραβάει δεδομένα οποιασδήποτε IP αρκεί να γνωρίζει το «rocommunity» που στην περίπτωση αυτή είναι το «public»



Εικόνα 13 Μορφοποίηση SNMP.conf

Με την ολοκλήρωση της μορφοποίησης του αρχείου, θα χρειαστεί να γίνει restart του snmpd service ώστε να πάρει τις αλλαγές «systemctl restart snmpd.service».

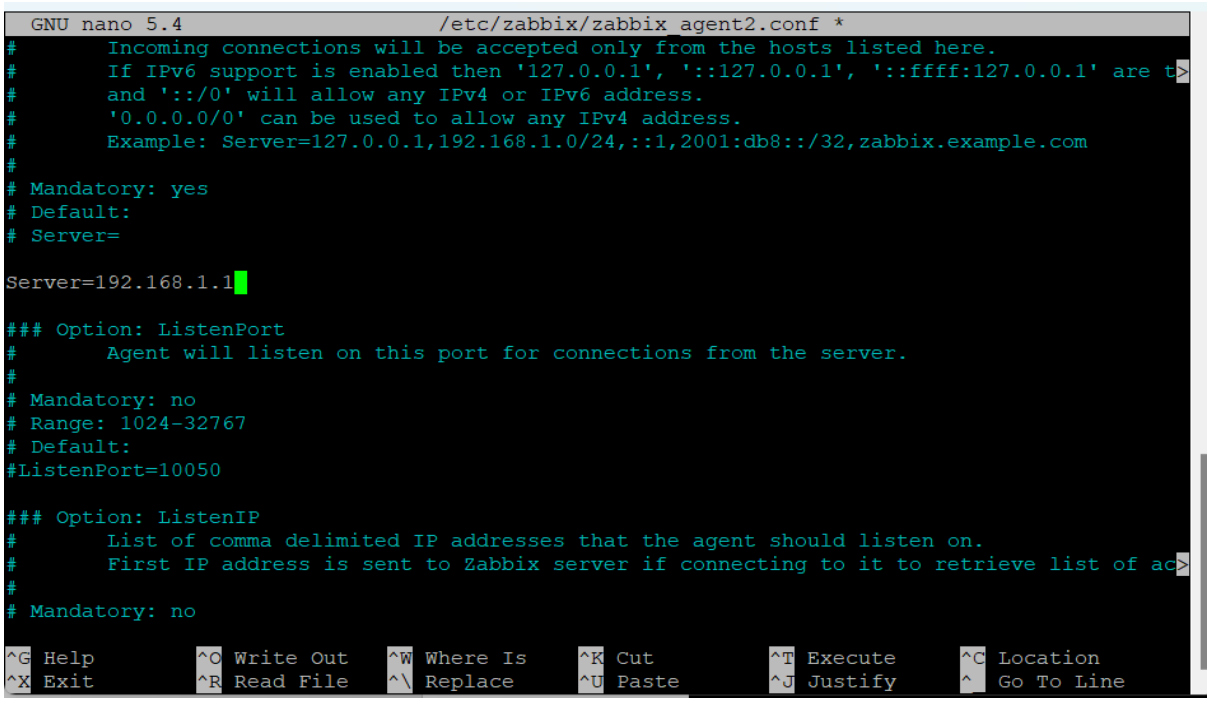
4.5.2 Agent based Zabbix client

Στην Agent based παρακολούθηση η παρακολούθηση γίνεται από έναν agent. Πρόκειται για λογισμικό που εγκαθίσταται στον υπολογιστή προκειμένου να ανακτήσει τα δεδομένα παρακολούθησης. Το Zabbix χρησιμοποιεί Zabbix-agent. Στις περισσότερες περιπτώσεις χρησιμοποιείται για την παρακολούθηση διεργασιών στον Server. Το βασικό πλεονέκτημα της παρακολούθησης με Agent είναι ότι πραγματοποιεί πιο ακριβή συλλογή δεδομένων από ό,τι η agentless παρακολούθηση. Σε αυτό το κεφάλαιο θα παρουσιαστούν τα βήματα εγκατάστασης και υλοποίησης παρακολούθησης με τη χρήση του Zabbix Agent. Ο Server που θα εγκατασταθεί ο Zabbix-agent είναι Linux Debian. Το πρώτο βήμα είναι η εγκατάσταση του Zabbix Agent στον Server. Για την εγκατάσταση του agent ανάλογα με την έκδοση του Zabbix-Server χρειάζεται να εγκατασταθεί και ο ανάλογος agent. Στο official site του Zabbix υπάρχουν οι διαθέσιμοι agent. Για την εγκατάσταση ακολουθήθηκαν οι παρακάτω εντολές στο CLI του Debian Server.

- `wget https://repo.zabbix.com/zabbix/6.4/debian/pool/main/z/zabbix-release/zabbix-release_6.4-1+debian11_all.deb`
- `dpkg -i zabbix-release_6.4-1+debian11_all.deb`
- `apt update`
- `apt install zabbix-agent2 zabbix-agent2-plugin-*`

- `systemctl restart zabbix-agent2`

Έπειτα χρειάζεται να γίνει παραμετροποίηση του αρχείου /etc/zabbix/zabbix_agentd.conf που δημιουργήθηκε κατά την εγκατάσταση με τις παραμέτρους «Server=και την IP του Zabbix-Server» και «ServerActive= και την IP του Zabbix-Server» όπως φαίνεται και στην Εικόνα 14.



Εικόνα 14. Αλλαγές στο αρχείο Zabbix_agentd.conf

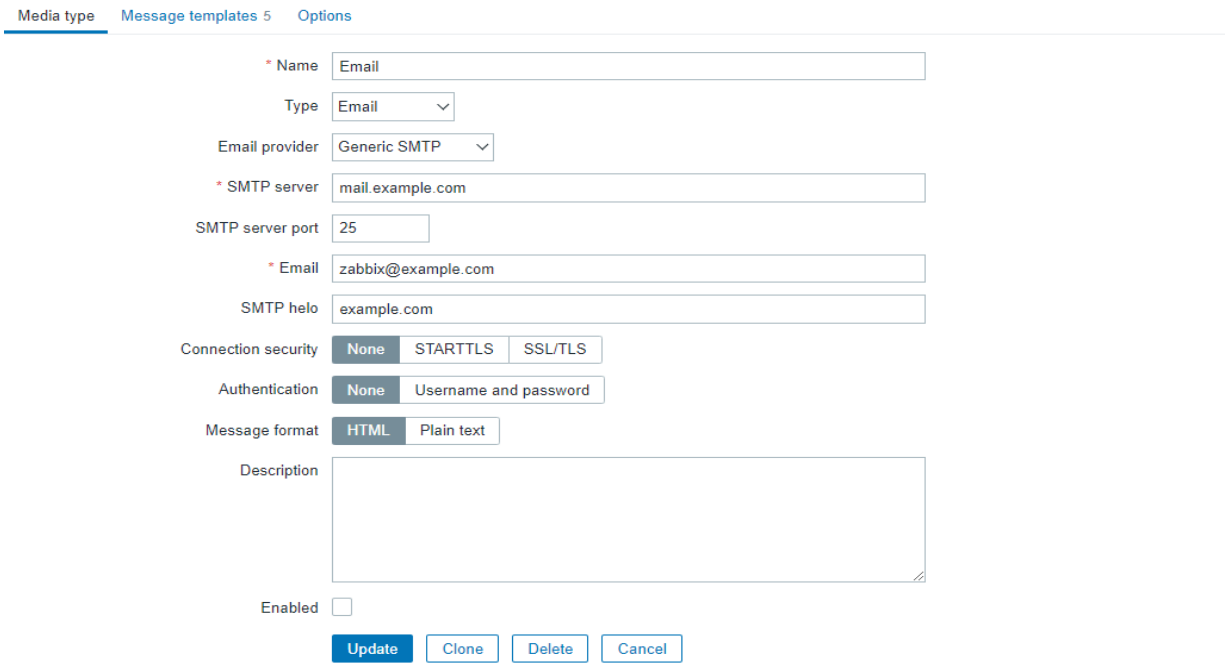
Τέλος χρειάζεται να γίνει restart του Zabbix-agent ώστε να οριστικοποιηθούν οι αλλαγές «systemctl restart zabbix-agent2».

4.6 Σύστημα ειδοποίησης

Αναπόσπαστο κομμάτι από ένα σύστημα που έχει υλοποιηθεί για να παρακολουθεί το δίκτυο και να μπορεί να παρατηρείτε πότε έχει συμβεί κάποια δυσλειτουργία είναι και να ενημερώνει τους αρμόδιους για την ύπαρξη της βλάβης. Είναι πολύ σημαντικό η βλάβη να παρατηρηθεί το συντομότερο καθώς παίζει καθοριστικό ρόλο στον χρόνο αντιμετώπισης της. Αυτό το ζήτημα έρχεται να λύσει το σύστημα ειδοποίησης που μπορεί να υλοποιηθεί μέσω του Zabbix συστήματος.

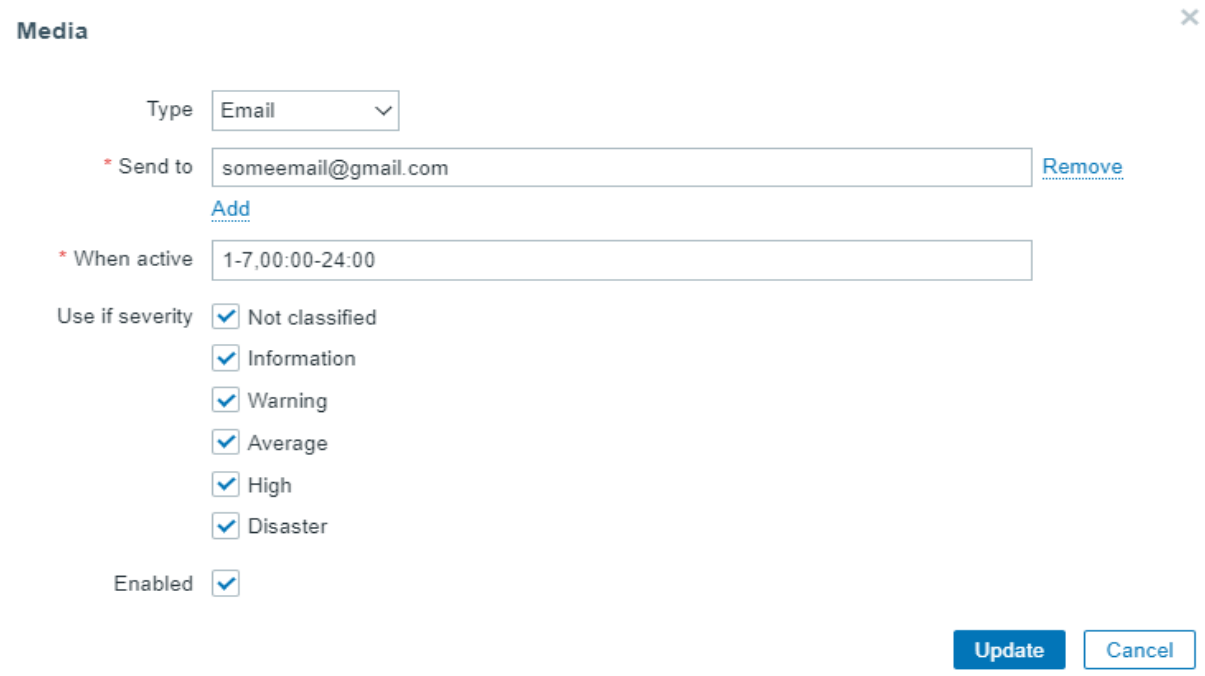
4.6.1 Σύστημα ειδοποίησης μέσω Email

Το Zabbix παρέχει την δυνατότητα να στηθεί σύστημα ειδοποίησης μέσω email. Σε αυτή την παράγραφο θα παρουσιαστούν τα βήματα που χρειάζονται για να υλοποιηθεί αυτή η υπηρεσία. Για αρχή χρειάζεται να ενεργοποιηθεί η email υπηρεσία καθώς και να συμπληρωθούν τα απαραίτητα πεδία ώστε να μπορεί να στέλνει emails. Για να επιτευχθούν τα προαναφερθέντα πρέπει μέσω του Zabbix UI από το «Alerts» -> «Media types» -> «Email» και να συμπληρωθούν τα πεδία που φαίνονται στην Εικόνα 15.



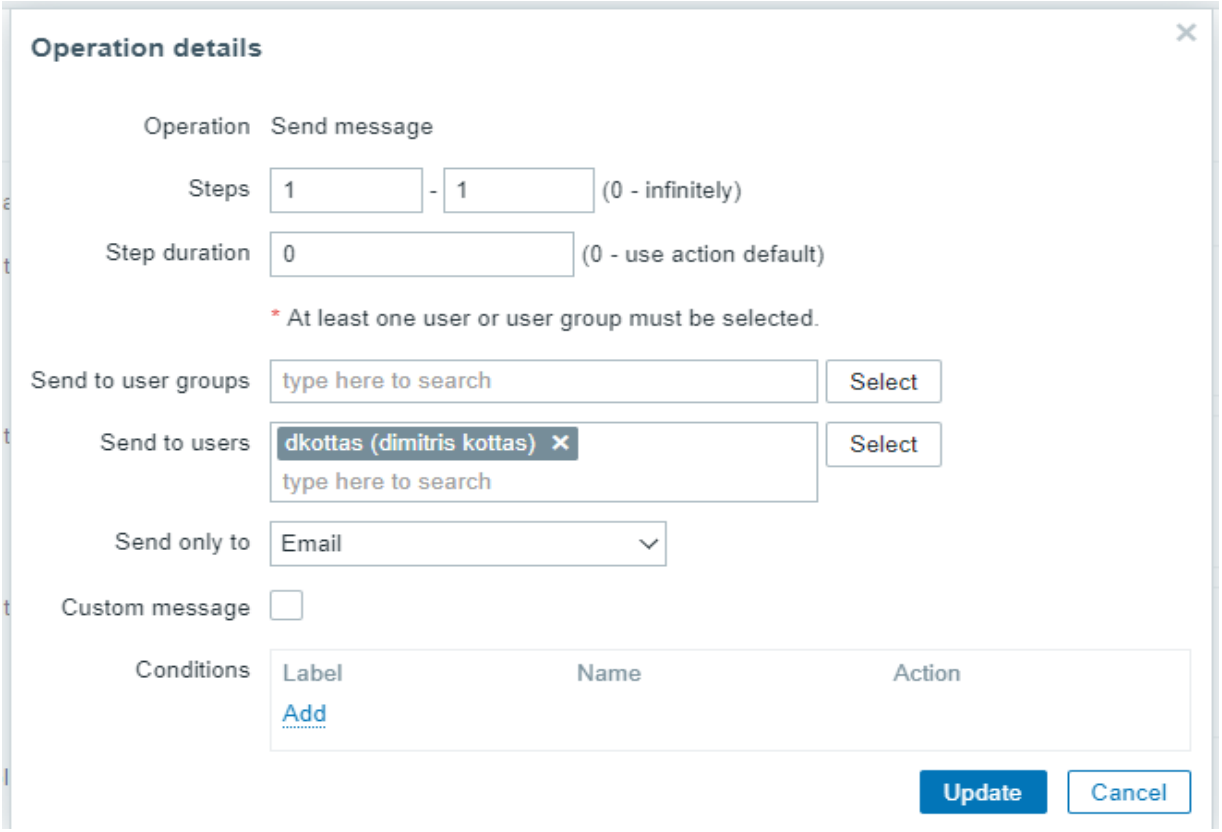
Εικόνα 15. Υλοποίηση Email συστήματος

Κατόπιν αυτών των αλλαγών πρέπει να οριστεί στους χρήστες που επιθυμείτε να στέλνονται οι ειδοποιήσεις, επιλέγοντας τον χρήστη από το «Users» -> «Users» -> και στην καρτέλα Media πρέπει να οριστεί το email που θα λαμβάνουν τις ειδοποιήσεις καθώς και τη σοβαρότητα της κάθε ειδοποίησης που θα στέλνετε όπως φαίνεται στην Εικόνα 16.



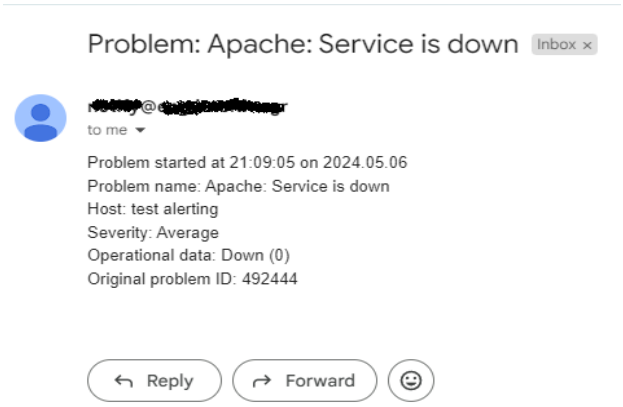
Εικόνα 16. Email και σοβαρότητα ειδοποιήσεων

Τέλος πρέπει να δημιουργηθεί «Action» μέσω του «Alerts» -> «Actions» -> «Create action» όπου από εκεί γίνεται να οριστεί για ποια triggers (ίσως και όλα) θα στέλνονται ειδοποιήσεις καθώς και σε ποιους χρήστες η Groups (Εικόνα 17.).

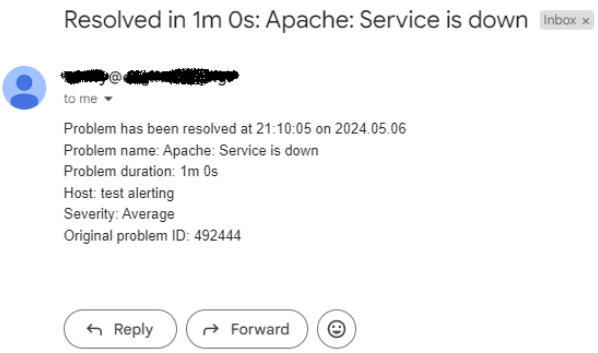


Εικόνα 17. Action παραμετροποίηση

Στις επόμενες δύο εικόνες (Εικόνα 18, Εικόνα 19) παρουσιάζεται ο τρόπος που έρχεται η ειδοποίηση του προβλήματος μέσω email καθώς και η επίλυση του.



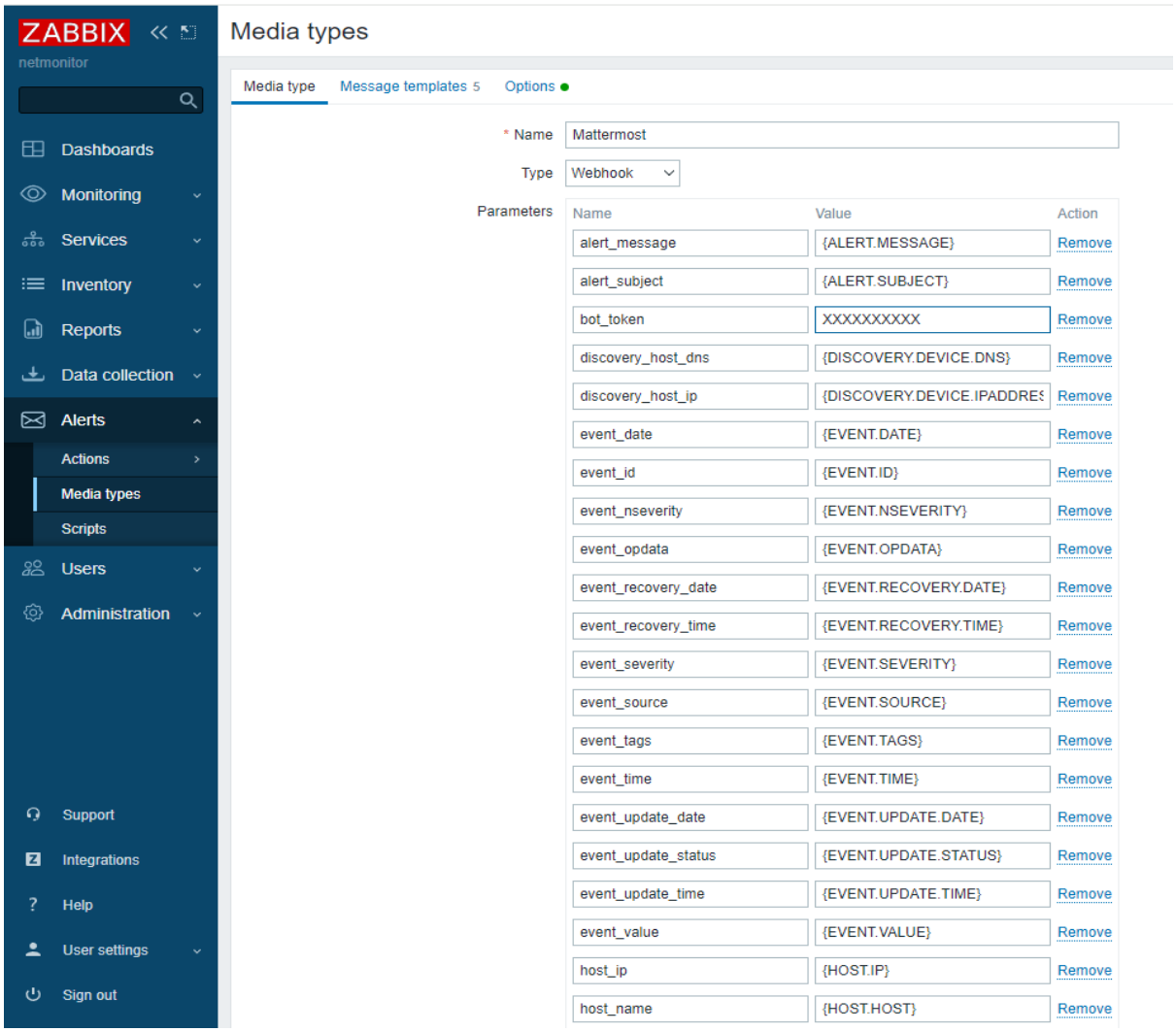
Εικόνα 18. Ειδοποίηση προβλήματος



Εικόνα 19. Ειδοποίηση επίλυσης προβλήματος

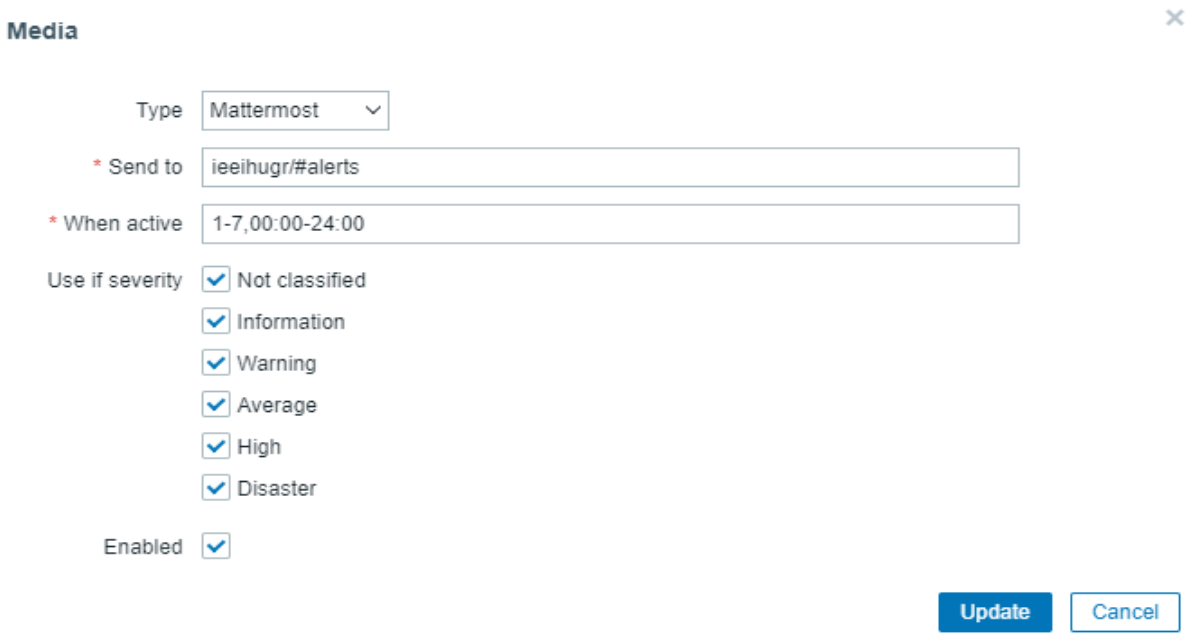
4.6.2 Σύστημα ειδοποίησης στο Mattermost

Στο Zabbix πέρα την ειδοποίηση μέσω emails μπορούν να υλοποιηθούν και άλλοι εναλλακτικοί τρόποι ιδιοποίησης ένας εκ των οποίων είναι μέσω Mattermost. Για την εφαρμογή αυτής της μεθόδου πέραν από τις τροποποιήσεις που πρέπει να γίνουν από την πλευρά του Mattermost που θα παρουσιαστούν στην επόμενη ενότητα χρειάζεται να γίνουν και κάποιες αλλαγές στο Zabbix. Αρχικά πρέπει να ενεργοποιηθεί το Mattermost webhook καθώς και να γίνουν κάποιες αλλαγές οι οποίες μπορούν να βρεθούν στο official site του Zabbix. Για την παραμετροποίηση του Mattermost webhook από το «Alerts» -> «Media types» -> «Mattermost» πρέπει να γίνουν κάποιες αλλαγές όπως και να οριστεί το bot_token που παράχθηκε κατά την δημιουργία του Bot μέσω του Mattermost (Εικόνα 20).



Εικόνα 20. Mattermost webhook

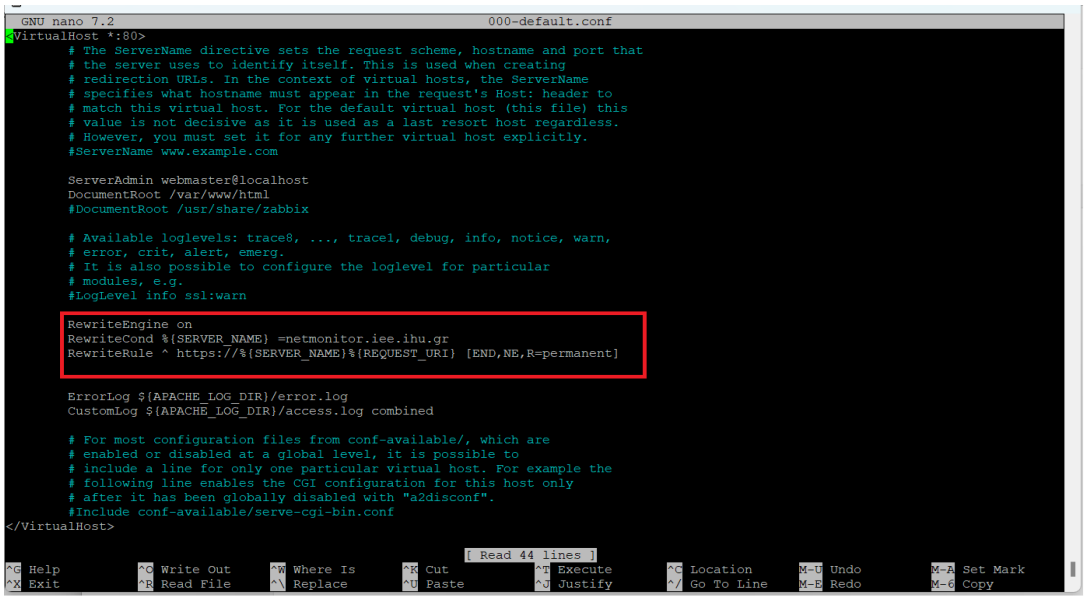
Έπειτα πρέπει να δημιουργηθεί ένας Zabbix χρήστης ο οποίος θα έχει administrator δικαιώματα και εκεί να οριστεί το channel ή ο χρήστης που θα στέλνονται τα alerts από το mattermost bot (εικόνα 21.).



Εικόνα 21. Mattermost κανάλι και σοβαρότητα ειδοπίησης

4.7 Zabbix apache

Για την πρόσβαση του Zabbix μέσω browser χρειάζεται η ρύθμιση της υπηρεσίας apache η οποία θα εξυπηρετεί τα request των «πελατών» και θα μπορούν να συνδέονται στο Zabbix UI. Επιπλέον χρειάζεται ένα domain για ποιο εύκολη απομνημόνευση του Zabbix URL. Στην περίπτωση αυτής της υλοποίησης χρησιμοποιήθηκε το domain netmonitor.iee.ihu.gr καθώς και δημιουργήθηκε ssl certificate μέσω της letsencrypt για την ασφαλή περιήγηση στο Zabbix. Στην Εικόνα 22. φαίνονται οι ρυθμίσεις του apache που κάνουν redirect τα http request στο domain netmonitor.iee.ihu.gr σε https.



Εικόνα 22. Redirect http σε https

Τέλος στην Εικόνα 23. παρουσιάζεται το κομμάτι κώδικα του apache όπου ότι request γίνεται στο https με το domain netmonitor.iee.ihu.gr (πόρτα 443) θα εμφανίζει το Login UI του Zabbix καθώς και το letsencrypt certificate που χρησιμοποιείτε για την ασφαλή περιήγηση.

```
  GNU nano 7.2                                        000-default-le-ssl.conf
<IfModule mod_ssl.c>
<Virtualhost *:443>
    ServerName netmonitor.iee.ihu.gr
    DocumentRoot /usr/share/zabbix

SSLCertificateFile /etc/letsencrypt/live/netmonitor.iee.ihu.gr/fullchain.pem
SSLCertificateKeyFile /etc/letsencrypt/live/netmonitor.iee.ihu.gr/privkey.pem
Include /etc/letsencrypt/options-ssl-apache.conf
</Virtualhost>
</IfModule>
```

Εικόνα 23. HTTPS ρυθμίσεις

Κεφάλαιο 5ο: Mattermost

Στο κεφάλαιο αυτό θα γίνει παρουσίαση της εφαρμογής Mattermost με γενικά στοιχεία, τα κύρια χαρακτηριστικά, την διαδικασία εγκατάστασης, τις λειτουργίες που χρησιμοποιήθηκαν καθώς και τον τρόπο που συνεργάζεται με το Zabbix για να παρέχει τις επιθυμητές λειτουργίες.

5.1 Επισκόπηση του Mattermost

Το Mattermost είναι μια πλατφόρμα επικοινωνίας ανοικτού κώδικα που επιτρέπει στις ομάδες εργασίας να συνδέονται, να συνεργάζονται και να μοιράζονται πληροφορίες με ασφαλή και αποτελεσματικό τρόπο. Σε αντίθεση με άλλα εργαλεία επικοινωνίας, το Mattermost βασίζεται στην άμεση ανταλλαγή μηνυμάτων και έχει σχεδιαστεί για επιχειρήσεις ή ομάδες που χρειάζονται μια πλατφόρμα συνεργασίας υψηλής απόδοσης. Το Mattermost έχει σχεδιαστεί για να προσφέρει στις ομάδες εργασίας μια πλατφόρμα επικοινωνίας που τους επιτρέπει να μοιράζονται πληροφορίες σε πραγματικό χρόνο. Το εργαλείο διαθέτει μια σειρά από λειτουργίες που επιτρέπουν στους διαχειριστές της πλατφόρμας να προσαρμόζουν την εμπειρία τους αναλόγως με τις ανάγκες κάθε ομάδας εργασίας. Το Mattermost προσφέρει ένα ευρύ φάσμα λειτουργιών, όπως ιδιωτικά μηνύματα, διαχείριση έργων, αποθήκευση αρχείων και διαδικτυακή συνεργασία. Λόγω της φύσης του ως ανοικτού κώδικα, κάθε εταιρεία ή ομάδα μπορεί να προσαρμόσει το εργαλείο στις δικές της ανάγκες [19].

5.2 Τρόπος λειτουργίας του Mattermost

Το Mattermost λειτουργεί ως πλατφόρμα επικοινωνίας σε πραγματικό χρόνο που επιτρέπει στις ομάδες εργασίας να συνδέονται, να συνεργάζονται και να μοιράζονται πληροφορίες. Το εργαλείο είναι πολύ ευέλικτο και προσαρμόζεται στις ανάγκες κάθε ομάδας εργασίας. Για να χρησιμοποιηθεί το Mattermost, είναι απαραίτητο να δημιουργηθεί ένας λογαριασμός στην πλατφόρμα και να ενταχθεί στην ομάδα εργασίας. Οι ομάδες εργασίας οργανώνονται σε κανάλια, τα οποία είναι ομάδες συζήτησης στις οποίες τα μέλη της ομάδας μπορούν να αλληλεπιδρούν και να μοιράζονται πληροφορίες. Τα κανάλια στο Mattermost είναι αρκετά ευέλικτα, επιτρέποντας στα μέλη της ομάδας να δημιουργούν ιδιωτικά ή δημόσια κανάλια. Τα ιδιωτικά κανάλια είναι ιδανικά για συζητήσεις στην ομάδα ή για την οργάνωση συγκεκριμένων έργων που πρέπει να κοινοποιούνται μόνο σε ένα υποσύνολο μελών της ομάδας. Τα δημόσια κανάλια, από την άλλη πλευρά, είναι ιδανικά για ανοιχτές συζητήσεις και για την οργάνωση έργων στα οποία πρέπει να συμμετέχουν όλα τα μέλη της ομάδας. Το Mattermost επιτρέπει επίσης τη διαχείριση project μέσω της δημιουργίας εργασιών και της ανάθεσης αρμοδιοτήτων. Τα μέλη της ομάδας μπορούν να χρησιμοποιούν την πλατφόρμα για να δημιουργούν εργασίες, να ορίζουν προθεσμίες και να παρακολουθούν την πρόοδο του project. Η πλατφόρμα επιτρέπει στα μέλη της ομάδας να ενημερώνονται για την πρόοδο του project ανά πάσα στιγμή και από οπουδήποτε. Ένα άλλο ωραίο χαρακτηριστικό του Mattermost είναι η διαχείριση αρχείων. Το εργαλείο επιτρέπει στα μέλη της ομάδας να ανεβάζουν και να μοιράζονται αρχεία, γεγονός που διευκολύνει τη συνεργασία και την ομαδική εργασία. Το Mattermost επιτρέπει επίσης στα μέλη της ομάδας να έχουν πρόσβαση στα κοινόχρηστα αρχεία σε πραγματικό χρόνο, καθιστώντας την online συνεργασία απλή [19].

5.3 Πλεονεκτήματα του Mattermsot

Το κύριο πλεονέκτημα της χρήσης του Mattermost είναι η δυνατότητα να διατηρούνται οι ομάδες εργασίας συνδεδεμένες σε πραγματικό χρόνο. Το εργαλείο είναι πολύ ευέλικτο και επιτρέπει στα μέλη της ομάδας να μοιράζονται πληροφορίες, να ορίζουν εργασίες και να συνεργάζονται διαδικτυακά. Ένα

άλλο πλεονέκτημα της χρήσης του Mattermost είναι η ασφάλεια. Η πλατφόρμα είναι πολύ ασφαλής και πληρεί τα υψηλότερα πρότυπα ασφαλείας. Αυτό σημαίνει ότι το εργαλείο είναι ιδανικό για εταιρείες που πρέπει να προστατεύσουν τις εμπιστευτικές πληροφορίες των πελατών τους ή της επιχείρησής τους. Το Mattermost είναι επίσης πολύ εύκολο στη χρήση με διαισθητικό περιβάλλον εργασίας χρήστη. Η πλατφόρμα είναι εξαιρετικά προσαρμόσιμη και οι διαχειριστές του εργαλείου μπορούν να το προσαρμόσουν στις συγκεκριμένες ανάγκες των ομάδων εργασίας τους. Επίσης, επειδή πρόκειται για εργαλείο ανοικτού κώδικα, η κοινότητα χρηστών του Mattermost εξελίσσεται και βελτιώνεται συνεχώς. Εκτός από αυτά τα πλεονεκτήματα, το Mattermost προσφέρει μια σειρά χαρακτηριστικών που επιτρέπουν στα μέλη της ομάδας να παραμένουν οργανωμένα και να συνεργάζονται αποτελεσματικότερα. Ορισμένα από αυτά τα χαρακτηριστικά περιλαμβάνουν ειδοποιήσεις σε πραγματικό χρόνο, ενσωμάτωση με άλλα εργαλεία συνεργασίας, ιστορικό συνομιλιών και δυνατότητα προσαρμογής των καναλιών [19].

5.4 Εγκατάσταση του Mattermost

Σε αυτή την παράγραφο θα αναφέρονται τα βήματα που ακολουθήθηκαν για την εγκατάσταση του Mattermost στον Linux Debian 12 Server. Αρχικά για την εγκατάσταση του Mattermost χρειάζεται να υπάρχει εγκατεστημένη postgresql καθώς και mattermost βάση με χρήστη. Παρακάτω παρατίθενται τα βήματα αυτά.

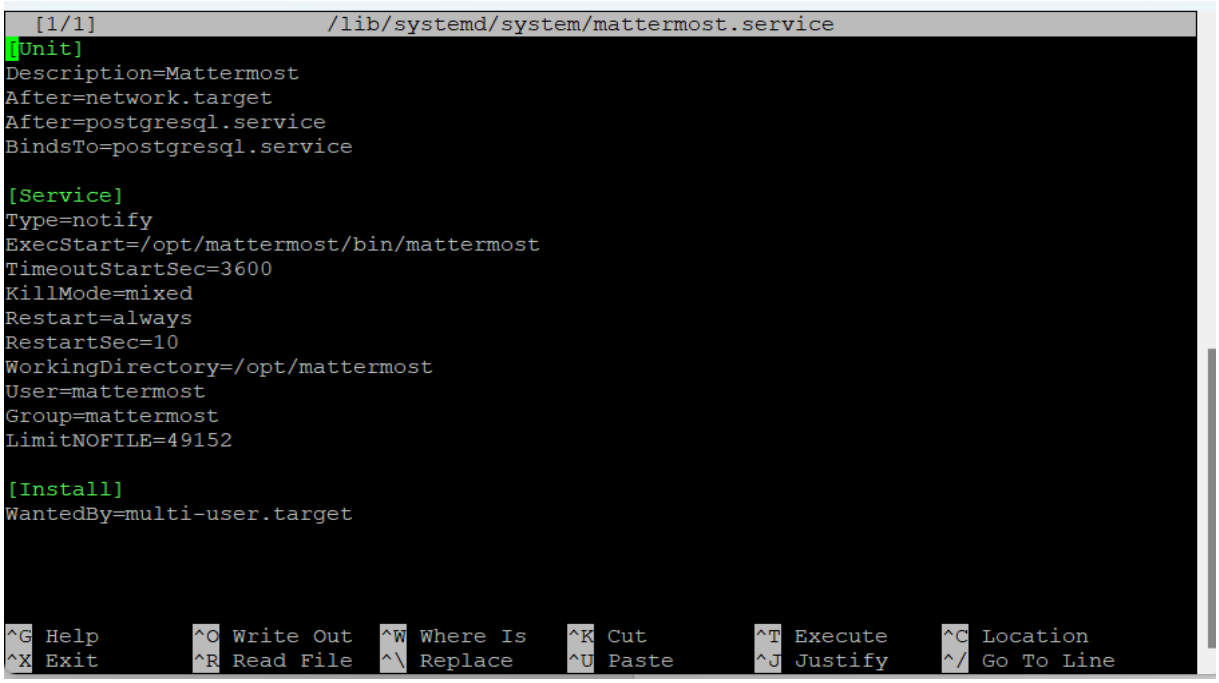
- `apt install postgresql postgresql-contrib -y`
- `sudo -u postgres psql`
- `CREATE USER mmuser WITH PASSWORD 'P@ssw0rd!';`
- `CREATE DATABASE mattermost;`
- `GRANT ALL PRIVILEGES ON DATABASE mattermost to mmuser;`
- `\q`

Έπειτα είναι εφικτό να γίνει η εγκατάσταση του Mattermost. Για την εγκατάσταση ακολουθήθηκαν τα βήματα που υπάρχουν στο official site του Mattermost.

- `wget https://releases.mattermost.com/9.6.1/mattermost-9.6.1-linux-amd64.tar.gz`
- `tar -xvzf mattermost*.gz`
- `mv mattermost /opt`
- `mkdir /opt/mattermost/data`
- `useradd --system --user-group mattermost`
- `chown -R mattermost:mattermost /opt/mattermost`
- `chmod -R g+w /opt/mattermost`

Για να μπορέσει να δημιουργηθεί το mattermost service χρειάζεται να δημιουργιθεί το αρχείο με το service configuration που φαίνεται στην Εικόνα 24.

- `nano /lib/systemd/system/mattermost.service.`

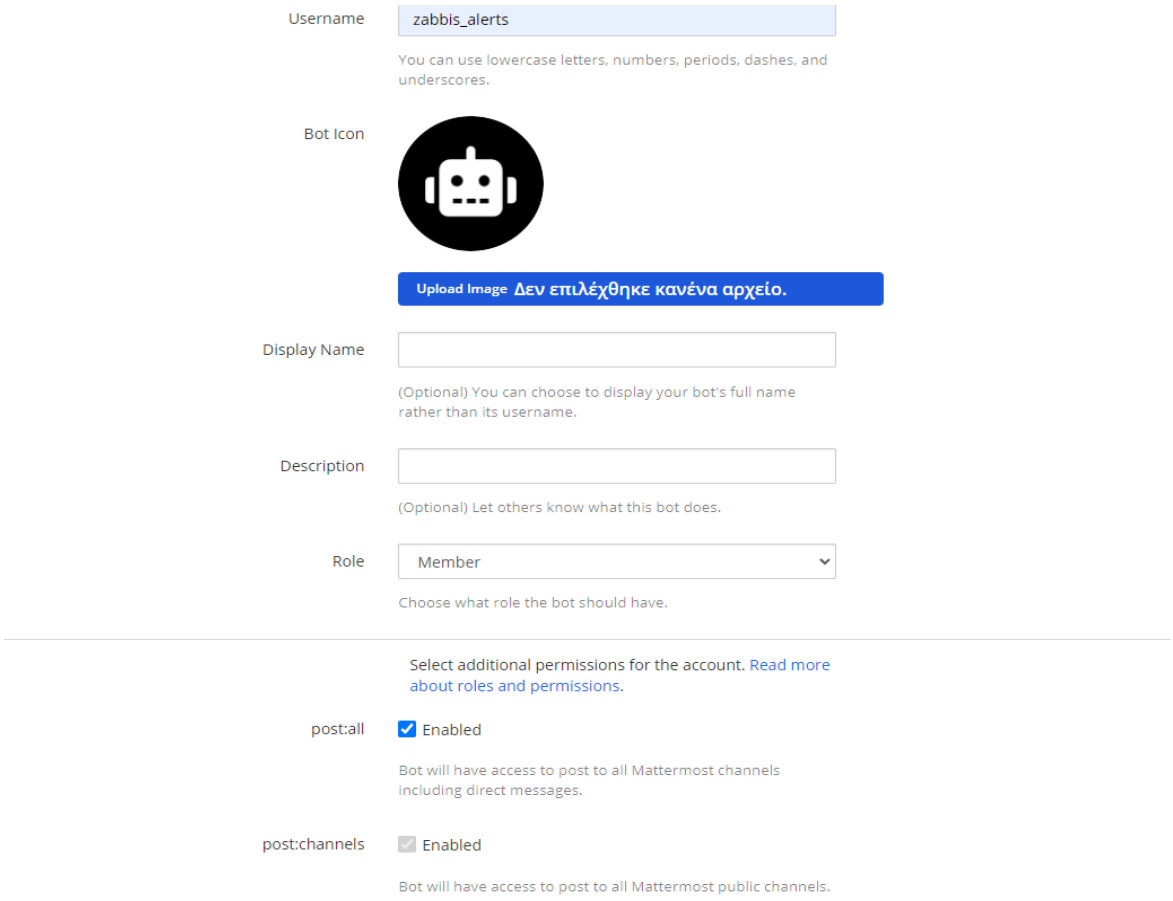


Εικόνα 24. Mattermost.service αρχείο

Τέλος χρειάζεται να γίνουν κάποιες παραμετροποιήσεις στο /opt/mattermost/config/config.json αρχείο που αναφέρονται στις οδηγίες και να γίνει start το service «systemctl start mattermost» και enable ώστε αν γίνει επανεκκίνηση του Server να ξεκινήσει αυτόματα το service (systemctl enable mattermost.service).

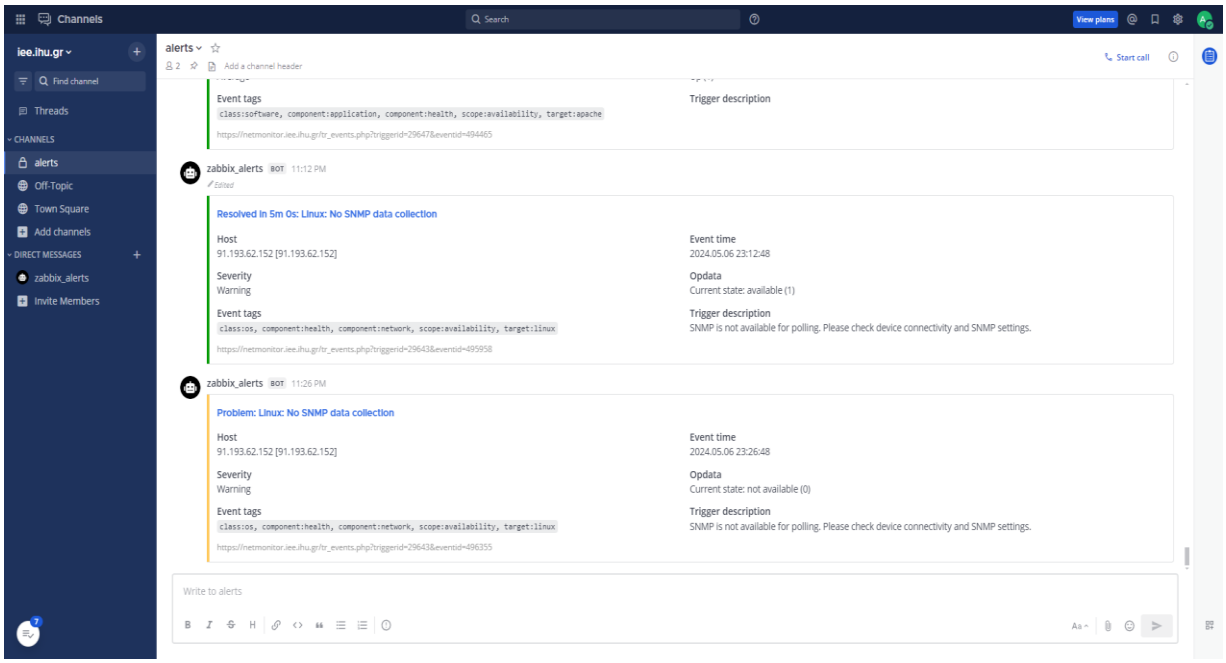
5.5 Mattermost bot για τις Zabbix ειδοποιήσεις

Το Mattermost έχει πολλές και χρήσιμες λειτουργίες όπως προαναφέρθηκαν στην ενότητα 5.2. Στην συγκεκριμένη περίπτωση το Mattermost πρόκειται να χρησιμοποιηθεί ώστε οι ιδιοποιήσεις που δημιουργούνται από το Zabbix να στέλνονται σε ιδιωτικό κανάλι που υπάρχει στο Mattermost για να μπορεί όποιους επιθυμεί ο διαχειριστής του δικτύου να βλέπουν μαζικά τις ειδοποιήσεις. Για αυτόν τον λόγο έχει υλοποιηθεί το Mattermost bot το οποίο στέλνει τις ειδοποιήσεις μέσω του mattermost webhook που παρέχει το Zabbix όπως παρουσιάστηκε και στην ενότητα 4.6.2. Για την υλοποίηση αυτή πρέπει μέσω του Mattermost UI από το «Main menu» -> «Integrations» -> «Bot accounts» -> «Add Bot Account» να δημιουργηθεί ένα bot (Εικόνα 25.). Κατόπιν της δημιουργίας παράγεται ένα Token το οποίο πρέπει να εισαχθεί στο mattermost webhook του Zabbix (Εικόνα 20.).



Εικόνα 25. *Δημιουργία Bot*

Τέλος χρειάζεται το bot που δημιουργήθηκε να εισαχθεί στο Channel που θα στέλνει τις ειδοποιήσεις. Με αυτή την υλοποίηση οι ειδοποιήσεις που έχουν ρυθμιστεί να στέλνονται με το mattermost-webhook του Zabbix θα αποστέλλονται στο κανάλι που έχει οριστεί. Στην Εικόνα 26. φαίνεται το UI του Mattermost καθώς και ο τρόπος που παρουσιάζονται οι ειδοποιήσεις.



Εικόνα 26. *Mattermost κανάλι με τις ειδοποιήσεις*

5.6 Mattermost apache

Για την πρόσβαση του Mattermost μέσω browser χρειάζεται η ρύθμιση της υπηρεσίας apache η οποία θα εξυπηρετεί τα request τον «πελατών» και θα μπορούν να συνδέονται στο Mattermost UI. Επιπλέον χρειάζεται ένα domain για ποιο εύκολη απομνημόνευση του Mattermost URL. Στην περίπτωση αυτής της υλοποίησης χρησιμοποιήθηκε το domain mattermost.iee.ihu.gr καθώς και δημιουργήθηκε ssl certificate μέσω της letsencrypt για την ασφαλή περιήγηση στο Mattermost. Στην Εικόνα 27. φαίνονται οι ρυθμίσεις του apache που κάνουν redirect τα http request στο domain mattermost.iee.ihu.gr σε https.

```
  GNU nano 7.2
<VirtualHost *:80>
    ServerName mattermost.iee.ihu.gr
    Redirect permanent / https://mattermost.iee.ihu.gr/
RewriteEngine on
RewriteCond %{SERVER_NAME} =mattermost.iee.ihu.gr
RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]
</VirtualHost>
```

Εικόνα 27. *http to https redirect*

Επίσης, στην Εικόνα 28. παρουσιάζεται το κομμάτι κώδικα του apache όπου ότι request γίνεται στο https με το domain mattermost.iee.ihu.gr (πόρτα 443) θα το προωθεί μέσω proxy στην πόρτα 8065 όπου και ακούει η υπηρεσία του mattermost και θα εμφανίζει το UI του mattermost. Τέλος φαίνεται το κομμάτι κώδικα που χρησιμοποιεί το letsencrypt certificate για την ασφαλή περιήγηση.

```
<VirtualHost *:443>
    ServerName mattermost.iee.ihu.gr

    SSLEngine on
    ProxyRequests off

    ProxyPass / http://127.0.0.1:8065/ retry=0
    ProxyPassReverse /  http://127.0.0.1:8065/

    RewriteEngine On
    RewriteCond %{REQUEST_URI} /api/v[0-9]+/(users/)?websocket [NC]
    RewriteCond %{HTTP:UPGRADE} ^WebSocket$ [NC,OR]
    RewriteCond %{HTTP:CONNECTION} ^Upgrade$ [NC]
    RewriteRule .* ws://127.0.0.1:8065%{REQUEST_URI} [P,QSA,L]

    SSLCertificateFile /etc/letsencrypt/live/mattermost.iee.ihu.gr/fullchain.pem
    SSLCertificateKeyFile /etc/letsencrypt/live/mattermost.iee.ihu.gr/privkey.pem
    Include /etc/letsencrypt/options-ssl-apache.conf
</VirtualHost>
```

Εικόνα 28. *Https ρυθμίσεις και proxy*

Κεφάλαιο 6ο: Συμπεράσματα

Στις μέρες μας τα δίκτυα όλο και μεγαλώνουν κατά συνέπια να είναι όλο και ποιο δύσκολη η συνεχόμενη παρακολούθηση τους. Αυτή την λύση έδωσε η συγκεκριμένη πτυχιακή στην οποία υλοποιήθηκε το Zabbix monitoring σύστημα το οποίο παρέχει συνεχόμενη παρακολούθηση σε όλη την διάρκεια της ημέρας καθώς και εκτενή δεδομένα. Παρέχει άμεση ειδοποίηση στους διαχειριστές του δικτύου για τα προβλήματα που προκαλούνται στο δίκτυο ούτος ώστε να μπορεί να αντιμετωπιστεί το συντομότερο δυνατό το πρόβλημα. Με αυτόν τον τρόπο παρέχεται μεγαλύτερη αξιοπιστία καθώς αρκετές φορές αν το πρόβλημα εντοπιστεί και αντιμετωπιστεί αρκετά γρήγορα τότε ο τελικός χρήστης ίσως να μην αντιληφθεί καν την βλάβη. Στη πτυχιακή αυτή έχει δημιουργηθεί μια κατανόηση του τι είναι το εργαλείο παρακολούθησης δικτύου, τα πλεονεκτήματα που επιφέρει και τα βασικά χαρακτηριστικά του. Το Zabbix επιλέχθηκε προκειμένου να πραγματοποιηθεί η παρακολούθηση του δικτύου του πανεπιστημίου λόγω του ότι, είναι ανοιχτού κώδικα αλλά και πληρεί όλες τις προϋποθέσεις και τις λειτουργίες που χρειάστηκε να επιτευχθούν. Σαν αποτέλεσμα της πτυχιακής υπάρχει πλέον ένα Zabbix σύστημα παρακολούθησης αλλά και σύστημα ειδοποίησης μέσω email όπως και ειδοποίησης στο Mattermost κανάλι.

Κατά την εκπόνηση της πτυχιακής, θα ήθελα να επισημάνω ότι υπήρξαν αρκετές δυσκολίες στο πρακτικό κομμάτι οι οποίες χρειάστηκαν έρευνα και μελέτη για να μπορέσουν να υλοποιηθούν. Κάποιες από αυτές ήταν η υλοποίηση του mattermost bot ώστε να μπορεί να προωθεί τις ειδοποιήσεις στο mattermost κανάλι. Επίσης αρκετά δύσκολο ήταν να γίνει αντιληπτό ποιες θα είναι οι ιδιοποιήσεις που θα πρέπει να λαμβάνονται για κάθε συσκευή καθώς ανάλογα με την συσκευή και τις λειτουργίες που πραγματοποιεί, το είδος των ιδιοποιήσεων παραμετροποιείται. Αλλά μέσα από τις δυσκολίες και την έρευνα κατάφερα να μάθω πως να υλοποιώ τέτοιου είδους σύστημα παρακολούθησης αλλά και ποιες ανάγκες και δυσκολίες υπάρχουν κατά την εφαρμογή του.

Έπειτα από την εγκατάσταση του Zabbix συστήματος θα πρέπει πλέον οι διαχειριστές του δικτύου και συστήματος να ειδοποιούνται για οποιαδήποτε δυσλειτουργία του συστήματος εγκαίρως καθώς και να έχουν εκτενή δεδομένα για την λειτουργία των συσκευών. Κάποιες ενδεικτικές ειδοποιήσεις που έχουν εφαρμοστεί είναι η χρήση αυξημένης CPU και RAM, η έλλειψη διαθέσιμου χώρου, χάσιμο επικοινωνίας με την συσκευή (το οποίο τις περισσότερες φορές σημαίνει ότι έχει κλείσει η κολλήσει) και άλλες εξιδεικευμένες ειδοποιήσεις ανάλογα με την συσκευή και τις ανάγκες της.

Όσον αφορά τις μελλοντικές βελτιώσεις που μπορούν να γίνουν στο ήδη υπάρχων σύστημα, μια από αυτές θα μπορούσε να ήταν η υλοποίηση αυτοματοποιημένων scripts που θα τρέχουν σε συγκεκριμένα triggers όταν γίνονται αντιληπτά από το σύστημα. Για παράδειγμα θα μπορούσε να εφαρμοζόταν script το οποίο θα έκανε επανεκκίνηση του apache όταν αντιλαμβανόταν ότι η apache υπηρεσία είναι κάτω. Επίσης μια ακόμα μελλοντική βελτίωση θα μπορούσε να ήταν η εφαρμογή αυτόματης αναγνώρισης νέων συσκευών, ώστε να μην χρειάζεται κάθε φορά που πρέπει να εγκατασταθεί νέα συσκευή να πρέπει να γίνονται ενέργειες από την πλευρά του διαχειριστή του συστήματος για την αναγνώριση της νέας συσκευής. Τέλος θα μπορούσε να γίνει ενσωμάτωση συστήματος το οποίο θα μπορεί να πραγματοποιεί τηλεφωνικές κλήσεις στα triggers που θα οριστούν στο Zabbix. Αυτή η λειτουργία είναι αρκετά κρίσιμη σε κάποιες από τις ειδοποιήσεις καθώς θα πρέπει να γίνει όσο το δυνατόν συντομότερα αντιληπτό το πρόβλημα για την άμεση επίλυση του.

Βιβλιογραφία

[1] paessler, «paessler AG,» [Ηλεκτρονικό]. Available: https://www.paessler.com/it-explained/server. [Πρόσβαση 19 Μάιος 2024].

[2] vmware, «vmware,» [Ηλεκτρονικό]. Available: https://www.vmware.com/topics/glossary/content/virtual-machine.html. [Πρόσβαση 15 Μάιος 2024].

[3] Wikipedia, «en.wikipedia.org,» [Ηλεκτρονικό]. Available: https://en.wikipedia.org/wiki/Network_switch. [Πρόσβαση 19 Μάιος 2024].

[4] E. Neuens, «SANS,» 10 Ιούλιος 2023. [Ηλεκτρονικό]. Available: https://www.sans.org/blog/what-is-linux/. [Πρόσβαση 17 Μάιος 2024].

[5] proxmox, «proxmox,» [Ηλεκτρονικό]. Available: https://pve.proxmox.com/pve-docs/pve-admin-guide.html. [Πρόσβαση 14 Μάιος 2024].

[6] proxmox, «proxmox,» [Ηλεκτρονικό]. Available: https://www.proxmox.com/en/proxmox-virtual-environment/overview. [Πρόσβαση 14 Μάιος 2024].

[7] proxmox, «proxmox,» [Ηλεκτρονικό]. Available: https://www.proxmox.com/en/proxmox-virtual-environment/features. [Πρόσβαση 14 Μάιος 2024].

[8] I. Wikithon, «el.wikipedia.org,» 15 Μάιος 2024. [Ηλεκτρονικό]. Available: https://el.wikipedia.org/wiki/Apache_HTTP_%CE%B5%CE%BE%CF%85%CF%80%CE%B7%CF%81%CE%B5%CF%84%CE%B7%CF%84%CE%AE%CF%82. [Πρόσβαση 16 Μάιος 2024].

[9] Cloudflare, «Cloudflare,» [Ηλεκτρονικό]. Available: https://www.cloudflare.com/learning/dns/glossary/what-is-a-domain-name/. [Πρόσβαση 17 Μάιος 2024].

[10] Cloudflare, «Cloudflare,» [Ηλεκτρονικό]. Available: https://www.cloudflare.com/learning/ssl/what-is-an-ssl-certificate/. [Πρόσβαση 16 Μάιος 2024].

[11] R. W. X. C. Yongjin Xu, «Implementation of Sluice and Pumping Station,» Shanghai, China, 2011.

[12] Cloudflare, «Cloudflare,» [Ηλεκτρονικό]. Available: https://www.cloudflare.com/learning/ddos/glossary/internet-control-message-protocol-icmp/. [Πρόσβαση 19 Μάιος 2024].

[13] M. C. G. L. R. N. a. B. Y. Rui Castro, «Network Tomography: Recent Developments,» 2004.

[14] D. Drogseth, «NETWORKWORLD,» 10 Νοέμβριος 2003. [Ηλεκτρονικό]. Available: https://www.networkworld.com/article/892035/infrastructure-management-hp-invests-in-route-analytics-with-packet-design.html. [Πρόσβαση 16 Μάιος 2024].

[15] L. Barnes, «MetricFire,» 10 Φεβρουάριος 2023. [Ηλεκτρονικό]. Available: https://www.metricfire.com/blog/agent-vs-agentless-monitoring/. [Πρόσβαση 17 Μάιος 2024].

[16] R. Laflamme, «Aukiv,» [Ηλεκτρονικό]. Available: https://www.auvik.com/franklyit/blog/agent-vs-agentless-monitoring/. [Πρόσβαση 17 Μάιος 2024].

[17] Zabbix, «Zabbix,» [Ηλεκτρονικό]. Available: https://www.zabbix.com/documentation/current/en/manual. [Πρόσβαση 17 Μάιος 2024].

[18] A. Vladishev, Interviewee, *Zabbix Conference.* [Συνέντευξη]. 20 Νοέμβριος 2014.

[19] ServersX5, «ServersX5,» [Ηλεκτρονικό]. Available: https://x5servers.com/en/what-is-mattermost-and-how-does-it-work/. [Πρόσβαση 15 Μάιος 2024].