



Τ.Ε.Ι. ΠΕΛΟΠΟΝΝΗΣΟΥ

Σχολή Τεχνολογικών Εφαρμογών (έδρα Σπάρτη)

Τμήμα Μηχανικών Πληροφορικής Τ.Ε.

*Σχεδίαση και υλοποίηση desktop
virtualization με τεχνικές cloud computing
και έμφαση στις κινητές συσκευές.*

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

ΠΑΠΑΔΟΠΟΥΛΟΥ ΜΙΧΑΗΛ



Τ.Ε.Ι. ΠΕΛΟΠΟΝΝΗΣΟΥ

Σχολή Τεχνολογικών Εφαρμογών (έδρα Σπάρτη)

Τμήμα Μηχανικών Πληροφορικής Τ.Ε.

*Σχεδίαση και υλοποίηση desktop
virtualization με τεχνικές cloud computing
και έμφαση στις κινητές συσκευές.*

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΤΟΥ

ΠΑΠΑΔΟΠΟΥΛΟΥ ΜΙΧΑΗΛ

Επιβλέπων: Μπάρδης Γεώργιος

Σπάρτη , Οκτώβριος 2014

(Υπογραφή)

(Υπογραφή)


(Υπογραφή)

.....

.....

.....

(Υπογραφή)

A handwritten signature in blue ink, consisting of stylized, overlapping loops and a long horizontal stroke extending to the right. The signature is positioned above a dotted line.

ΠΑΠΑΔΟΠΟΥΛΟΣ ΜΙΧΑΗΛ

© 2014 – All rights reserved

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ' ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον καθηγητή κ. Γεώργιο Μπάρδη για τη δυνατότητα που μου έδωσε να εκπονήσω αυτή την πτυχιακή εργασία, μέσα από την οποία είχα την ευκαιρία να ασχοληθώ με πολύ ενδιαφέροντα και πρακτικά ζητήματα, καθώς επίσης για την εποπτεία και τη βοήθειά του.

Περίληψη

Η πληροφορική και οι τεχνολογίες της αναπτύσσονται με πολύ γρήγορους ρυθμούς. Μάλιστα η τεχνολογία του virtualization (εικονικοποίησης) με την εξέλιξη της έχει βελτιώσει αισθητά αρκετούς τομείς της πληροφορικής. Μερικά χρόνια πριν, ήταν αναγκαίο να υπάρχουν πολλά υπολογιστικά συστήματα ώστε να μπορούν να υποστηριχτούν πολλές εφαρμογές. Τώρα πια μπορούμε, σε μία μόνο επεξεργαστική μονάδα, να φιλοξενήσουμε πολλές εφαρμογές και να έχουμε κεντρική διαχείριση. Όλο αυτό μάλιστα είναι και ο προπομπός του cloud computing(υπολογιστικό νέφος), δηλαδή απομακρυσμένης πρόσβασης και επεξεργασίας αρχείων.

Με αυτή την εργασία, σκοπός μου είναι να μπορέσω να εισάγω έναν απλό αναγνώστη στις τεχνολογίες του virtualization και του cloud computing, παραθέτοντας τα είδη τους αρκετά περιεκτικά όπως και πλεονεκτήματα ή μειονεκτήματα που υπάρχουν.

Στην εργασία υπάρχει και ανάπτυξη μίας πολύ σημαντικής περίπτωσης χρήσης του virtualization στο cloud computing. Δηλαδή, ανάπτυξη πλατφόρμας η οποία να επιτρέπει desktop virtualization με χρήση cloud computing, με σκοπό να παρέχει την δυνατότητα για λειτουργία από απόσταση εφαρμογών γραφείου σε κινητές συσκευές και ταμπλέτες με μεγάλη διαθεσιμότητα. Έχει στηθεί ένα σύστημα HA clustering με χρήση του Heartbeat μεταξύ δύο Ubuntu servers που έχουν ως υπηρεσία το Moodle.

Αυτή η εργασία αποτελεί σημείο αναφοράς για όποιον επιθυμεί να γνωρίσει τον κόσμο και τα πλεονεκτήματα του virtualization και του cloud computing καθώς και όποιο επιθυμεί να μάθει βήμα προς βήμα πώς να εγκαταστήσει ένα βασικό σύστημα HA clustering ή απλά να μάθει το σκεπτικό της τεχνολογίας που χρησιμοποιούν μεγάλες εταιρείες clouding και είναι σχεδόν πάντα διαθέσιμες προς του πελάτες τους.

Λέξεις κλειδιά: Εικονικοποίηση, υπολογιστικό νέφος, κινητές συσκευές, μεγάλη διαθεσιμότητα διακομιστών

Abstract

IT and its technologies develop at a very fast rate. Indeed, the technology of virtualization and its development over the years have considerably improved several areas of IT. In the past, in order to support many applications it was necessary to have multiple computer systems. Nowadays, it is possible to have multiply applications supported by only ONE computer system and also have central management of them. All this indeed is the forerunner of cloud computing, i.e. remote access and file processing.

The aim of my thesis is to introduce technologies of virtualization and cloud computing to a simple reader by quoting their categories, advantages and disadvantages.

Furthermore, in my thesis, I have developed a case study that is an example of a very important use of virtualization in cloud computing. That is, the development of a platform that allows desktop virtualization using cloud computing, designed to provide the ability of remote operation of office applications on mobile devices and tablets with high availability. I have set up a system of HA clustering using Heartbeat between two Ubuntu servers whose service is Moodle.

My thesis is a reference point for anyone wishing to discover the world and the benefits of virtualization and cloud computing as well as for anyone wishing to learn step by step how to install a basic HA availability system or simply want to learn the reasoning of this technology that many cloud companies use and are almost always available to their customers.

Key words: Virtualization, cloud computing, mobile devices, high availability servers

Πίνακας περιεχομένων

Περίληψη	1
Abstract	3
1. Εισαγωγή.....	6
1.1 Αντικείμενο της πτυχιακής.....	6
1.2 Δομή	6
2. Virtualization (Εικονικοποίηση).....	8
2.1. Ιστορική αναδρομή στο virtualization (εικονικοποίηση)	8
2.2 Είδη Virtualization (εικονικοποίησης)	9
2.2.1 Hypervisor (υπερόπτης)	9
2.2.2 Virtual machine (εικονικές μηχανές).....	13
2.2.3 Desktop virtualization (εικονικοποίηση επιφάνειας εργασίας)	14
2.2.4 Server virtualization (εικονικοποίηση διακομιστών)	17
2.2.5 Storage virtualization (εικονικοποίηση αποθήκευσης)	19
2.2.6 Network virtualization (εικονικοποίηση δικτύου)	20
2.2.7 Virtualization of applications (εικονικοποίηση εφαρμογών).....	21
2.3 Τα οφέλη του Virtualization (Εικονικοποίηση).....	24
3. Cloud computing (Υπολογιστικό νέφος).....	25
3.1 Εισαγωγή	25
3.2 Χαρακτηριστικά του Cloud computing (Υπολογιστικό νέφος)	27
3.3 Cloud Computing Services (Υπηρεσίες)	30
3.4 Cloud Computing types (Είδη)	34
3.5 Προβλήματα και περιορισμοί του Cloud Computing	35
4.Υλοποίηση περίπτωσης Virtualization & Cloud computing	37
4.1 Περιγραφή περίπτωσης.....	37
4.2 Εγκατάσταση των Virtual Servers	38
4.3 Εγκατάσταση MOODLE	38
4.3.1 Σύνδεση σαν Super User στους servers	39
4.3.2 Αλλαγή hostname στους servers.....	39
4.3.3 Static IP στους servers	40
4.3.4 Εγκατάσταση των απαραίτητων στοιχείων Apache, MySQL, PHP.....	41
4.3.5 Λήψη του Moodle	47

4.3.6 Αντιγραφή του /opt/moodle στο /var/www/html	48
4.3.7 Ρύθμιση του MySQL server	49
4.3.8 Ολοκλήρωση ρυθμίσεων του Moodle	50
4.4 Servers υψηλής διαθεσιμότητας με το Heartbeat	57
4.4.1 Επίλυση των hostnames μεταξύ των 2 servers.....	57
4.4.2 Εγκατάσταση του Heartbeat	58
4.4.3 Τελικές ρυθμίσεις του heartbeat	60
4.5 Αλλαγές που χρειάζονται για να λειτουργήσει το HA clustering του moodle μέσω internet.....	63
4.6 Παράδειγμα εφαρμογής του HA clustering στο LAN.....	64
Βιβλιογραφία	67

1. Εισαγωγή

Στο παρελθόν πολλά πράγματα ήταν πιο περίπλοκα συγκριτικά με σήμερα που με την εξέλιξη της τεχνολογίας γίνονται πιο απλά. Αυτό όσο και αν ακούγεται παράξενο, ισχύει και για την ίδια την τεχνολογία. Δηλαδή η εξέλιξη της, μπορεί να βοηθήσει στην απλούστευση ήδη υπαρχόντων τεχνολογιών. Για παράδειγμα η τεχνολογία του virtualization (εικονικοποίηση) με την εξέλιξη της έχει βελτιώσει αισθητά αρκετούς τομείς της πληροφορικής. Παλιότερα ήταν αναγκαίο για να υποστηριχτούν πολλαπλές εφαρμογές να υπάρχουν πολλά υπολογιστικά συστήματα. Σήμερα όμως με την εξέλιξη του virtualization μπορούμε, με τη χρήση ενός υπολογιστικού συστήματος, να φιλοξενήσουμε πολλαπλές εφαρμογές και να έχουμε κεντρική διαχείριση και έλεγχο αυτών. Εξέλιξη του virtualization μάλιστα είναι το cloud computing (υπολογιστικό νέφος), δηλαδή η δυνατότητα της απομακρυσμένης πρόσβασης αλλά και επεξεργασίας αρχείων.

1.1 Αντικείμενο της πτυχιακής

Το αντικείμενο της πτυχιακής εργασίας είναι να εισάγει τον αναγνώστη σε έννοιες όπως είναι το virtualization και το cloud computing. Αναλύονται τα βασικά είδη του virtualization όπως και του cloud computing καθώς και τα πλεονεκτήματα ή τα μειονεκτήματα που υπάρχουν. Επίσης αναφέρεται στην ανάπτυξη πλατφόρμας η οποία επιτρέπει desktop virtualization με χρήση cloud computing, με σκοπό να παρέχει την δυνατότητα για λειτουργία από απόσταση εφαρμογών γραφείου σε κινητές συσκευές και ταμπλέτες. Παρουσιάζεται ένα σημαντικό παράδειγμα της δύναμης του virtualization σε συνεργασία με το cloud computing. Σε αυτή την υλοποίηση δείχνουμε το πώς λειτουργούν πολλοί οργανισμοί cloud computing και τι τεχνικές χρησιμοποιούν ώστε να είναι σχεδόν πάντα διαθέσιμες και προσβάσιμες οι υπηρεσίες τους από τους χρήστες και πως λύνονται προβλήματα στο μηχανικό υλικό χωρίς να υπάρχουν απώλειες.

1.2 Δομή

Η πτυχιακή εργασία αποτελείται από 4 ενότητες. Η πρώτη ενότητα είναι η εισαγωγή της εργασίας, η δεύτερη ενότητα παρουσιάζει την έννοια του virtualization (εικονικοποίηση). Γίνεται μια μικρή ιστορική αναδρομή και παρουσιάζονται τα

βασικά είδη που υπάρχουν καθώς και πλεονεκτήματα και μειονεκτήματα στη χρήση αυτών. Στην τρίτη ενότητα παρουσιάζεται η έννοια του cloud computing (υπολογιστικό νέφος), τα χαρακτηριστικά γνωρίσματά του και αναφέρονται οι διαφορετικές cloud υπηρεσίες που παρέχονται όπως και πλεονεκτήματα και μειονεκτήματα τους. Τέλος, στην τέταρτη και τελευταία ενότητα παρουσιάζεται η υλοποίηση ενός συστήματος high availability server, με τη χρήση δύο virtual ubuntu server και το heartbeat.

2. Virtualization (Εικονικοποίηση)

Η εικονικοποίηση είναι ο διαμοιρασμός των διαθέσιμων πόρων ενός φυσικού ηλεκτρονικού υπολογιστή σε πολλές εικονικές μηχανές. Η εικονικοποίηση έχει τις ρίζες της στα πρώτα βήματα των υπολογιστών και χρησιμοποιείται σε πολλά τμήματα της πληροφορικής όπως οι υπολογιστές, τα δίκτυα ακόμα και διάφορες εφαρμογές. Σαν τεχνολογία μπορεί να εντοπιστεί σε οτιδήποτε υπολογιστές, από απλούς οικιακούς υπολογιστές μέχρι και (κυρίως) σε servers εταιρειών που δίνουν έμφαση στη σχέση κόστους κτήσης και απόδοσης επένδυσης. Η ανάγκη για virtualization δημιουργήθηκε για να καλύψει (κυρίως) τα εξής προβλήματα :

- Την αύξηση στο κόστος που υπήρχε στην διαχείριση πληροφοριακών συστημάτων.
- Τις απαιτήσεις φυσικού χώρου που ήταν απαραίτητος για την εξέλιξη των κέντρων δεδομένων.
- Την αδυναμία εκμετάλλευσης σε μεγάλο ποσοστό των φυσικών πόρων, των διαθέσιμων μηχανημάτων.

Το virtualization δηλαδή έφερε στα γεμάτα κέντρα δεδομένων και στους υποχρησιμοποιούμενους, από πλευράς ισχύς, servers τη δυνατότητα συμπίεσης πολλών φυσικών servers μέσα σε έναν server ο οποίος μπορεί να τρέχει πολλά virtual machines. Αυτό επιτρέπει στον server να χρησιμοποιεί το μέγιστο των δυνατοτήτων του και να μην πηγαίνει τόση ισχύ αχρησιμοποίητη. Έτσι μπήκε τέλος στην χρήση “one server, one application” (ένας server, μία εφαρμογή) που κατασπαταλούσε την ισχύ των servers αλλά και έφερνε μεγάλη σπατάλη χώρου στα κέντρα δεδομένων όπως και τεράστια κατανάλωση ρεύματος.

2.1. Ιστορική αναδρομή στο virtualization (εικονικοποίηση)

Στην εικονικοποίηση δομικό συστατικό είναι τα εικονικά μηχανήματα (virtual machines), τα οποία για πρώτη φορά χρησιμοποιήθηκαν το 1960 περίπου από την εταιρεία IBM σε μια προσπάθεια μείωσης του κόστους του εξοπλισμού που χρειαζόταν για τις διάφορες λειτουργίες. Τις πρώτες εφαρμογές της εικονικοποίησης τις συναντάμε στον IBM 7044 που δημιουργήθηκε στο τεχνολογικό ινστιτούτο της Μασαχουσέτης. Ακολούθησαν διάφορες προσπάθειες και μελέτες μέχρι που έφτασε

ο Άτλας του πανεπιστημίου του Μάντσεστερ, ο οποίος ήταν ο πρωτοπόρος στη χρησιμοποίηση εικονικής μνήμης αλλά και διαχείρισης κάποιων μηχανικών εξαρτημάτων ,όπως ο μερισμός του χρονισμού του επεξεργαστή για την εκτέλεση διαφόρων εργασιών, από τον επόπτη (supervisor) που ήταν ένα στοιχείο διαχείρισης πόρων. Αυτή η διαχείριση γινόταν με ειδικές εντολές ή επιπλέον κώδικα στον επόπτη (supervisor). [1][2]

2.2 Είδη Virtualization (εικονικοποίησης)

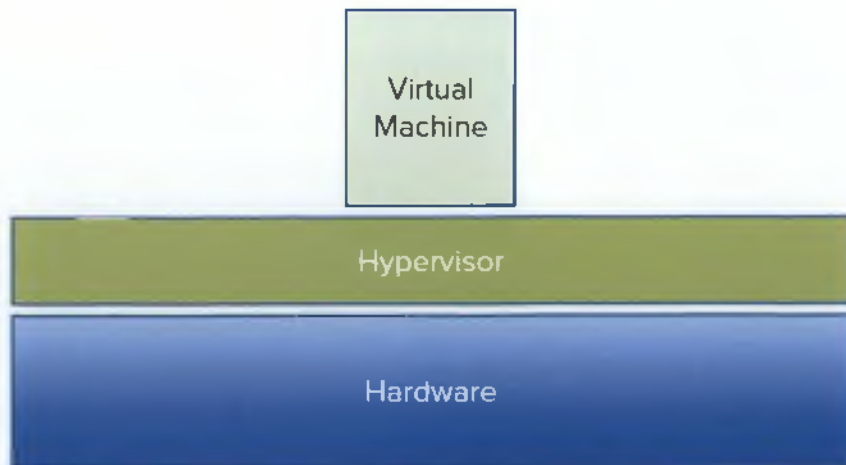
Η εικονικοποίηση εντοπίζεται σε πολλούς τομείς και συνοψίζεται στις εξής κύριες κατηγορίες :

- Desktop virtualization (εικονικοποίηση επιφάνειας εργασίας)
- Server virtualization (εικονικοποίηση διακομιστών)
- Storage virtualization (εικονικοποίηση αποθήκευσης)
- Network virtualization (εικονικοποίηση δικτύων)
- Application virtualization (εικονικοποίηση εφαρμογών)

Στη συνέχεια θα παρουσιάσω κάποιες γενικές, απαραίτητες για την εικονικοποίηση έννοιες, όπως και κάποιες πληροφορίες για κάθε είδος της.

2.2.1 Hypervisor (υπερόπτης)

Οι hypervisors είναι ένα στρώμα λογισμικού που βρίσκεται στην ουσία ενδιάμεσα από το υλικό (μηχανικά μέρη) και τις virtual machines (εικονικές μηχανές).



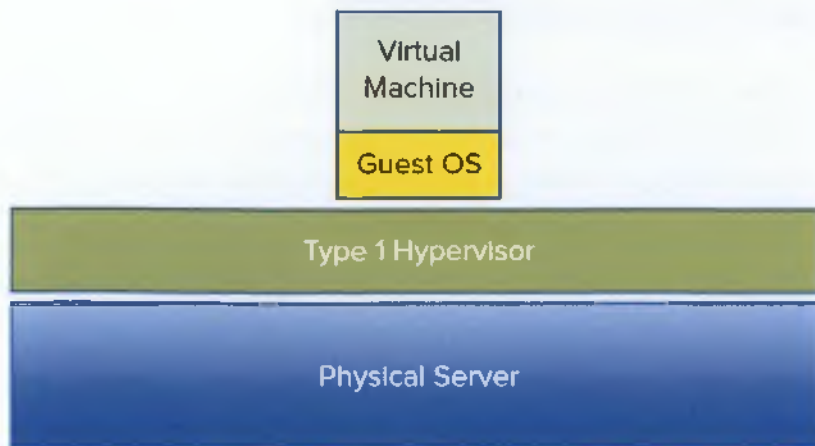
Εικόνα 1. Θέση hypervisor

Χωρίς την ύπαρξη του hypervisor, ένα λειτουργικό σύστημα επικοινωνεί απευθείας με το μηχανικό υλικό του συστήματος. Όλες οι λειτουργίες του δίσκου γίνονται άμεσα από το υποσύστημα του δίσκου και η μνήμη που χρειάζεται διαχειρίζεται απευθείας από το φυσική μνήμη του συστήματος. Χωρίς την ύπαρξη του hypervisor αν υπάρχουν πολλά λειτουργικά συστήματα μέσω πολλών εικονικών μηχανημάτων, θα έπαιρναν ταυτόχρονα τον έλεγχο του μηχανικού υλικού, κάτι που θα οδηγούσε σε καταστροφικές καταστάσεις. Ο hypervisor οργανώνει τις αλληλεπιδράσεις μεταξύ της κάθε εικονικής μηχανής και του μηχανικού υλικού, που όλοι οι συνδεδεμένοι χρήστες στην ουσία μοιράζονται.

Υπάρχουν δύο κατηγορίες του hypervisor. Ο τύπου 1 που χρησιμοποιείται κυρίως σε server virtualization (εικονικοποίηση διακομιστών) και ο τύπου 2 που χρησιμοποιείται σε desktop virtualization (εικονικοποίηση επιφάνειας εργασίας).

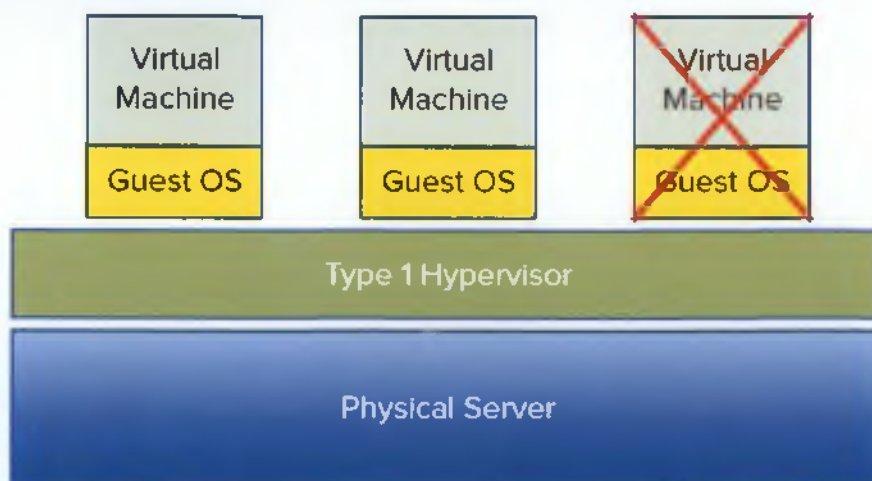
- *Hypervisor type 1 (υπερόπτης τύπου 1)*

Ο πρώτος τύπος του hypervisor, δουλεύει κατευθείαν στο υλικό του server χωρίς να υπάρχει λειτουργικό σύστημα ενδιάμεσα.



Εικόνα 2. Hypervisor type 1

Επίσης υπάρχει αρκετή ασφάλεια, δηλαδή ένας guest δεν μπορεί να επηρεάσει τον hypervisor από τον οποίο υποστηρίζεται. Μια virtual machine (εικονική μηχανή) μπορεί να βλάψει μόνο τον εαυτό της, προκαλώντας κάποιο πρόβλημα το οποίο δεν ξεφεύγει από τα όρια των virtual machines. Όλοι οι άλλοι guests μένουν ανεπηρέαστοι και μπορούν να συνεχίζουν τις εργασίες τους. Ανεπηρέαστος μένει και ο hypervisor. Επίσης, ένας malicious guest όπου προσπαθεί με κώδικα να παρέμβει στον hypervisor ή στους άλλους guests απλά δεν θα μπορέσει όπως βλέπουμε και στο παρακάτω σχήμα καθώς ακυρώνεται η σύνδεση του ως guest.



Εικόνα 3. Malicious guest failure

Όποτε υπάρχει hypervisor type 1, απαιτείται λιγότερη επεξεργαστική ισχύ καθώς πολλά virtual machines μπορούν να τρέξουν σε κάθε host server. Μερικά παραδείγματα τύπου 1 είναι : VMware ESX , Microsoft Hyper-V , Citrix Xen κ.α.

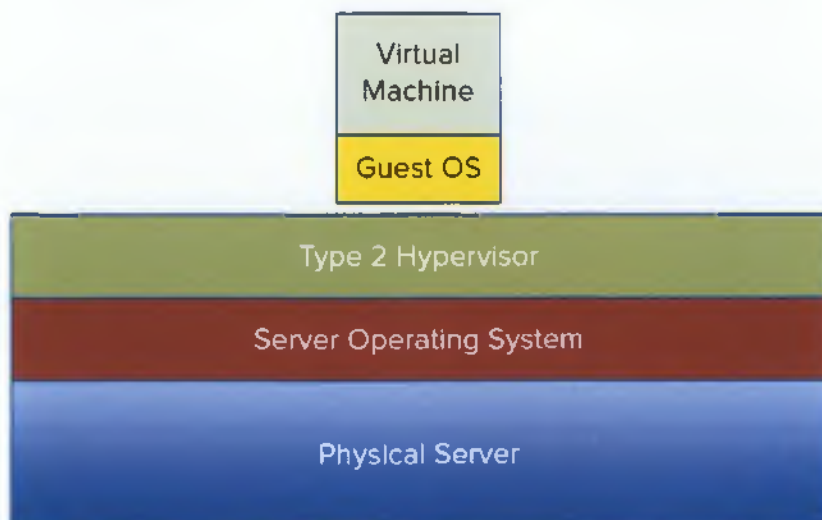
- Hypervisor type 2 (υπερόπτης τύπου 2)

Ο hypervisor type 2 είναι μια εφαρμογή που δουλεύει πάνω σε ένα λειτουργικό σύστημα. Σε 32 bit συστήματα , οι πρώτες και γρηγορότερες λύσεις ήταν type 2 . Το σύστημα ελέγχει πλήρως το υλικό, έτσι ο hypervisor θα εκμεταλλευτεί αυτήν την δυνατότητα του συστήματος προς όφελος του.

Το πλεονέκτημα αυτού του είδους είναι ότι έχει υποστήριξη για πάρα πολύ μεγάλη ποικιλία υλικού των υπολογιστών. Αυτή η μεγάλη συμβατότητα προκύπτει από την μεγάλη εξάρτηση που έχει από το λειτουργικό σύστημα. Ο type 2 συνήθως είναι εύκολο να εγκατασταθεί και να αναπτυχθεί καθώς εργασίες διαμόρφωσης υλικού, όπως το να μπει σε δίκτυο είναι ολοκληρωμένες εξαρχής από το λειτουργικό σύστημα.

Ο type 2 δεν είναι όσο αποδοτικός όσο ο type 1 λόγω του έξτρα στρώματος ανάμεσα στον hypervisor και στο υλικό. Κάθε φορά που μία virtual machine εκτελεί κάποια λειτουργία στο δίκτυο ή διαβάζει κάτι από το δίσκο ή οποιαδήποτε άλλη ενέργεια που χρησιμοποιείται το υλικό , το ζητάει πρώτα από τον hypervisor και εκείνος πρέπει να το ζητήσει από το λειτουργικό σύστημα το οποίο διαχειρίζεται όλα αυτά τα αιτήματα . Το λειτουργικό σύστημα στέλνει την πληροφορία πίσω στον hypervisor και μετά στον guest. Σε αντίθεση με τον hypervisor type 1 που μόνος του χρησιμοποιεί το υλικό χωρίς να μεσολαβεί το λειτουργικό σύστημα.

Ο hypervisor type 2 είναι λιγότερο αξιόπιστος από τον type 1, καθώς οτιδήποτε επηρεάζει την διαθεσιμότητα του λειτουργικού συστήματος μπορεί να έχει αντίκτυπο και στον υπερόπτη αλλά και στους guests. Μερικά παραδείγματα type 2 είναι τα εξής : VMware player , VMware workstation, Microsoft virtual server κ.α. [3][4][5]

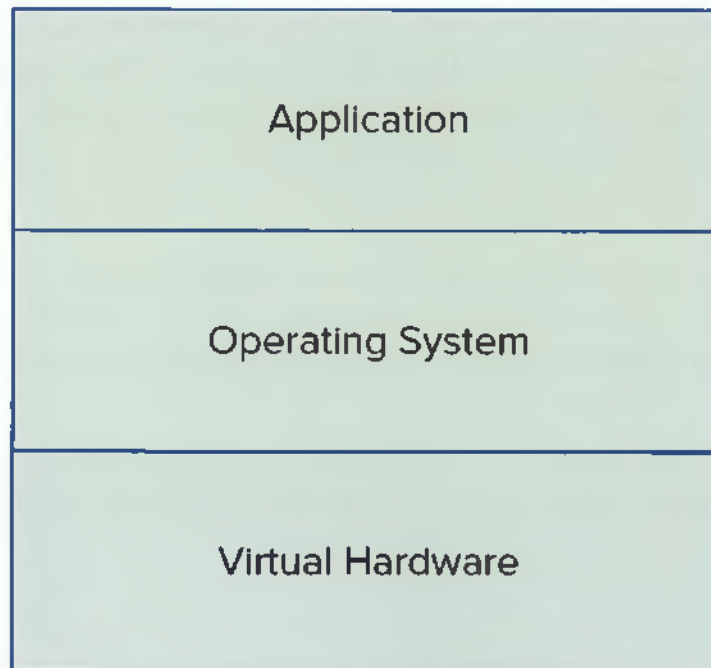


Εικόνα 4. Hypervisor type 2

2.2.2 Virtual machine (εικονικές μηχανές)

Οι virtual machines (VM) είναι οι θεμέλιοι λίθοι του virtualization. Είναι ένα απομονωμένο τμήμα λογισμικού που έχει την δυνατότητα να τρέχει πάνω του ένα λειτουργικό σύστημα. Στην ουσία λειτουργεί παρόμοια με έναν φυσικό server. Όπως ένας φυσικός server, έτσι και το VM είναι ρυθμισμένο με ένα σύνολο πόρων και υποστηρίζει ένα λειτουργικό σύστημα και οι εφαρμογές που τρέχουν σε αυτό μπορούν να ζητήσουν πρόσβαση στους πόρους αυτούς. Ένας φυσικός server μπορεί να τρέξει ένα λειτουργικό σύστημα τη φορά και μία ή λίγες υποστηριζόμενες εφαρμογές σε αυτό. Εν αντιθέσει πολλά VM μπορούν να δουλεύουν ταυτόχρονα σε έναν φυσικό server και το καθένα από αυτά μπορεί να τρέχει πολλά διαφορετικά λειτουργικά συστήματα το οποίο σημαίνει ότι θα μπορεί να υποστηρίξει πολλές εφαρμογές που θα τρέχουν ταυτόχρονα. Το VM δεν είναι τίποτα παραπάνω από μερικά αρχεία που περιγράφουν και στην ουσία περιλαμβάνουν το ίδιο το virtual server.

Τα κύρια αρχεία που αποτελούν ένα λειτουργικό VM είναι τα αρχεία ρυθμίσεων του αλλά και τα virtual αρχεία δίσκου. Τα αρχεία ρυθμίσεων περιγράφουν τους πόρους υλικού που διαθέτει το συγκεκριμένο VM, δηλαδή: CPU , RAM , disk , κάρτα δικτύου κλπ.



Εικόνα 5. Απλή απεικόνιση ενός VM

Virtual machines: Τα πλεονεκτήματά τους.

Οι virtual machines μπορούν να προσφέρουν πολλά πλεονεκτήματα σε σχέση με την εγκατάσταση του λειτουργικού συστήματος και του λογισμικού απευθείας στο φυσικό υλικό. Η απομόνωση, τους διασφαλίζει ότι οι εφαρμογές και οι υπηρεσίες που τρέχουν μέσα σε ένα VM δεν γίνεται να επηρεαστούν από το λειτουργικό σύστημα που πάνω του τρέχει το VM αλλά ούτε από τα άλλα VMs που συνυπάρχουν στο σύστημα. Επίσης τα VMs μπορούν εύκολα να αντιγραφούν και να μετακινηθούν σε άλλους host servers και να ξαναπροσαρμοστούν/ρυθμιστούν στο νέο φυσικό υλικό.[6]

2.2.3 Desktop virtualization (εικονικοποίηση επιφάνειας εργασίας)

Το desktop virtualization είναι η υποκατάσταση των laptops, desktops όπου οι χρήστες εργάζονται και αποθηκεύουν τις εργασίες τους σε αυτά. Στην ουσία δηλαδή σε αυτήν την περίπτωση αναφερόμαστε στην κατάσταση όπου το περιβάλλον του υπολογιστή όπου εργαζόμαστε δεν βρίσκεται στην φυσική μηχανή άμεσα αλλά είναι προσβάσιμο μέσω ενός υπολογιστικού μοντέλου πελάτη-διακομιστή. Η επιφάνεια εργασίας που εργαζόμαστε είναι αποθηκευμένη σε έναν απομακρυσμένο διακομιστή αντί του αποθηκευτικού χώρου που έχει η φυσική μηχανή του πελάτη. Επιπροσθαίτως όλα τα δεδομένα είναι διαθέσιμα από οποιοδήποτε σημείο αρκεί να υπάρχει πρόσβαση μέσω μιας φυσικής μηχανής όπως pc/laptop/tablet /smartphone.

Αυτό το μοντέλο δίνει την δυνατότητα σε πολλούς χρήστες να έχουν άμεση πρόσβασή ο καθένας στη δική του εξατομικευμένη επιφάνεια εργασίας όπως και στα δεδομένα του. Η πρόσβαση μπορεί να γίνει εφικτή μέσω τοπικού δικτύου ή το διαδίκτυο.

Το desktop virtualization έχει δύο κατηγορίες. Το desktop virtualization από την πλευρά του διακομιστή (hosted side ή server side) και από την πλευρά του πελάτη (client side).

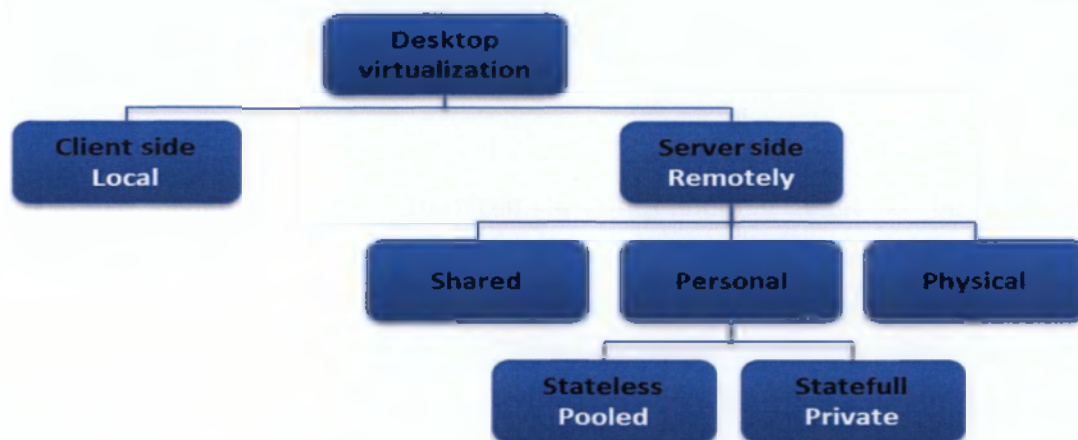
A. Desktop virtualization - Hosted side (πλευρά διακομιστή)

Σε ένα κέντρο δεδομένων διάφορες εικονικές μηχανές είναι αποθηκευμένες σε servers και οι χρήστες παίρνουν πρόσβαση σε αυτά ,για να χρησιμοποιήσουν τυχόν επιφάνειες εργασίας και δεδομένα, μέσω μεσιτών σύνδεσης (connection brokers).

Έτσι συνδέονται στον διακομιστή και αποκτούν την επιθυμητή πρόσβαση μέσω πρωτοκόλλων όπως το remote desktop protocol.

B. Desktop virtualization - Client side (πλευρά πελάτη)

Σε αυτή την περίπτωση όλες οι εργασίες και τα δεδομένα είναι αποθηκευμένα στο ίδιο PC/Server του πελάτη χωρίς να μεσολαβεί διακομιστής.



Εικόνα 2. Ένα γενικό διάγραμμα για τα είδη του desktop virtualization.

Πιο συγκεκριμένα:

Hosted side desktop virtualization:

Server hosted shared desktops RDS (διαμοιραζόμενες εικονικές επιφάνειες εργασίας)

Είναι ο τρόπος με τον οποίο έχουμε απομακρυσμένη πρόσβαση σε δεδομένα , εφαρμογές και λειτουργικά συστήματα τα οποία είναι αποθηκευμένα και εκτελούνται σε έναν κεντρικό διακομιστή που βρίσκεται σε κάποιο κέντρο δεδομένων. Δεν υπάρχουν περιορισμοί όσο αφορά το που βρίσκεται ο πελάτης και όλες οι διεργασίες γίνονται στον κεντρικό διακομιστή. Μέσω του πρωτόκολλου remote display protocol ο κάθε χρήστης έχει πρόσβαση στις ίδιες εφαρμογές και λειτουργικό όπως όλοι οι χρήστες αλλά ο καθένας ξεχωριστά ανοίγει δικό του desktop session. Τέτοιες υπηρεσίες προσφέρονται από εταιρείες όπως η Quest , η Microsoft , η Citrix , Cisco, VMware κ.α.

Server hosted personal remote virtual desktops VDI (εξατομικευμένες εικονικές επιφάνειες εργασίας)

Είναι μία λύση για απομακρυσμένη πρόσβαση σε επιφάνειες εργασίας που λειτουργούν σε κάποια από τις εικονικές μηχανές στο κέντρο δεδομένων. Αυτή η υπηρεσία είναι γνωστή ως Virtual desktop infrastructure (VDI) . Το VDI είναι η πιο διάσημη τεχνολογία desktop virtualization επιτρέποντας εγκατάσταση λειτουργικών συστημάτων , εφαρμογών αλλά και ξεχωριστό προφίλ του κάθε χρήστη ώστε να έχει ο καθένας την δική του μοναδική επιφάνεια εργασίας. Μέσω του πρωτοκόλλου remote desktop protocol (RDP) γίνεται ο χειρισμός οποιουδήποτε στοιχείου υπάρχει και λειτουργεί απομακρυσμένα στον διακομιστή . Τέτοιες λύσεις δίνουν η Microsoft, VMware, Citrix κ.α.

Server hosted personal remote physical desktops (εξατομικευμένες φυσικές επιφάνειες εργασίας)

Αυτές οι επιφάνειες εργασίας έχουν και δυνατότητα εκτέλεσης εφαρμογών πολυμέσων αλλά και 2D και 3D εφαρμογών , κάτι που σημαίνει ότι τα φυσικά μηχανήματα που τις φιλοξενούν διαθέτουν περίσσεια ισχύ επεξεργασίας γραφικών . Τα περισσότερα φυσικά μηχανήματα στις υπόλοιπες κατηγορίες δεν περιέχουν ισχυρή GPU, οπότε δεν μπορούν να ανταπεξέλθουν σε τέτοιες απαιτήσεις.

Επίσης σημαντικό ρόλο έχει και η κατάσταση/προφίλ του χρήστη στο virtualization. Δηλαδή αναφερόμαστε στα **stateless desktops & statefull desktops**. Στο statefull desktops (επιφάνειες εργασίας με κατάσταση) οι χρήστες έχουν την δυνατότητα να εγκαταστήσουν λογισμικά αλλά και να επεξεργαστούν την επιφάνεια εργασίας τους , οι αλλαγές που κάνουν αποθηκεύονται. Από την άλλη πλευρά στα stateless desktops (επιφάνειες εργασίας χωρίς κατάσταση) με την αποσύνδεση των χρηστών χάνονται οι περισσότερες αλλαγές που έκαναν κατά την σύνδεση τους. Βέβαια ανάλογα με τις ρυθμίσεις του διακομιστή κάποια δεδομένα αποθηκεύονται. Συνοψίζοντας τα stateless desktops είναι πιο εύκολα στην συντήρηση τους καθώς οι χρήστες δεν μπορούν να χειριστούν σημαντικά στοιχεία για την σωστή λειτουργία του συστήματος , όπως αναβάθμιση , αποθηκευτικός χώρος και γενικότερα ασφάλεια.[7][8][9]

Τα πλεονεκτήματα του Desktop virtualization:

- Συντήρηση όλων των σταθμών εργασίας από μια κεντρική τοποθεσία. Δεν υπάρχει ανάγκη για πρόσβαση στα φυσικά μηχανήματα για εγκατάσταση ή αναβάθμιση λογισμικού ή αλλαγή παραμετροποίησης.

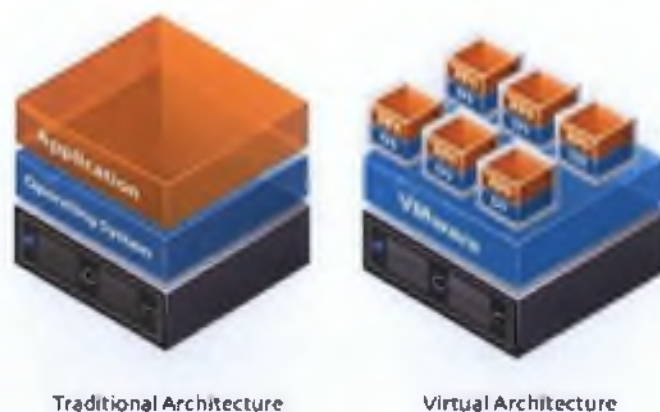
- Η υποστήριξη είναι βελτιωμένη με την σκίαση των συνεδριών (shadowing sessions) και την παρακολούθηση των ενεργειών των χρηστών. Με την χρήση εργαλείων απομακρυσμένης πρόσβασης RDP ή την κονσόλα διαχείρισης του VMware διακομιστή μπορούμε να παρακολουθήσουμε τι κάνει ο χρήστης όταν παρέχεται απομακρυσμένη υποστήριξη ή παίρνουμε τον έλεγχο του υπολογιστή.
- Εύκολη αποκατάσταση ή αντικατάσταση των υπολογιστών.
- Μείωση του κόστους καθώς παλιοί αδύναμοι υπολογιστές μπορούν να αξιοποιηθούν. Όλες οι επεξεργασίες γίνονται στον διακομιστή και όχι στους πελάτες. Ακόμα και ο απαιτούμενος χρόνος τεχνικής υποστήριξης είναι πολύ μειωμένος.

Περιορισμοί του Desktop virtualization:

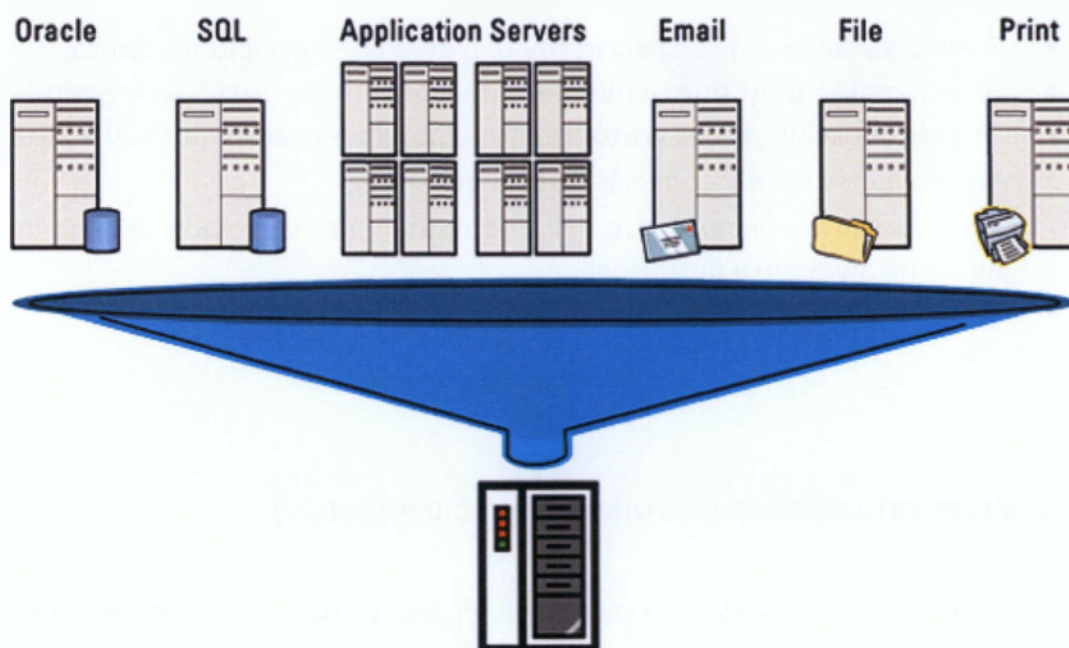
- Πιθανοί κίνδυνοι στην ασφάλεια εάν το δίκτυο δεν διαχειρίζεται σωστά.
- Μερική απώλεια της αυτονομίας και της ιδιωτικότητας των τελικών χρηστών.
- Πιθανές δυσκολίες στην εγκατάσταση και διαχείριση προγραμμάτων οδηγών για εκτυπωτές και άλλες περιφερειακές συσκευές.
- Ο χρόνος μη διαθεσιμότητας των συστημάτων σε περίπτωση βλάβης στο δίκτυο θα είναι πολύ αυξημένος.
- Οι υποδομές πολλές κοστίζουν αρκετά.

2.2.4 Server virtualization (εικονικοποίηση διακομιστών)

Η πιο διάσημη και πιο διαδεδομένη εφαρμογή του virtualization είναι το server virtualization. Τα τμήματα οργάνωσης πληροφοριών διαφόρων οργανώσεων IT χρησιμοποιούν πάρα πολύ αυτή την τεχνολογία καθώς βοηθάει πάρα πολύ στην μείωση του κόστους ενέργειας, με λιγότερη ανάγκη τροφοδοσίας των server όπως και λιγότερες απαιτήσεις για ψύξη, όπως και στην ανάγκη για περισσότερο φυσικό χώρο, κάτι που θα χρειαζόταν για την αποθήκευση περισσότερων φυσικών μηχανών. Έτσι με το server virtualization μετατρέπουμε πολλούς φυσικούς διακομιστές σε εικονικούς που συνυπάρχουν μέσα σε έναν φυσικό. Ο φυσικός διακομιστής ονομάζεται host και οι εικονικοί μέσα σε αυτόν ονομάζονται guests. Στις παρακάτω εικόνες απεικονίζεται χαρακτηριστικά το server virtualization.



Εικόνα 2. Παράδειγμα φυσικού και εικονικού διακομιστή



Εικόνα 3. Χαρακτηριστικό παράδειγμα της λειτουργίας του server virtualization

Σε έναν διακομιστή υπάρχει το λειτουργικό σύστημα και οι εφαρμογές που έχουν εγκατασταθεί πάνω σε αυτό (όπως βλέπουμε στις παραπάνω εικόνες). Όταν ο διακομιστής είναι φυσικός όμως, μπορεί να προκύψουν αρκετά προβλήματα σε τυχόν αλλαγή κάποιου μηχανικού μέρους λόγω βλάβης ή απλά αναβάθμισης. Δηλαδή κρίνεται απαραίτητο, όταν γίνει κάποια μετατροπή στα υλικά στοιχεία του, να γίνει επανεγκατάσταση και του λειτουργικού συστήματος και ως συνέπεια αυτού και των εφαρμογών. Στο server virtualization όμως υπάρχει ένα «στρώμα προστασίας

(στρώμα virtualization)» ανάμεσα στο υλικό και του λειτουργικού συστήματος, που επιτρέπει την εγκατάσταση πολλών αντιτύπων από λειτουργικά συστήματα και εφαρμογών στο φυσικό server. Έτσι βλάβες ή μεταβολές στο φυσικό υλικό δεν επηρεάζουν καθόλου τα virtual machines. [8]

2.2.5 Storage virtualization (εικονικοποίηση αποθήκευσης)

Η αναποτελεσματική αξιοποίηση αποθηκευτικών μέσων αλλά και η περιττή αγορά νέων , οδήγησε στην υιοθέτηση λύσεων storage virtualization. Αυτού του είδους το virtualization επιτρέπει ο διαθέσιμος αποθηκευτικός χώρος πολλών συστημάτων να συναθροίζεται σε έναν λογικό αποθηκευτικό χώρο. Αυτός ο εικονικός χώρος αποθήκευσης προορίζεται για χρήση από χρήστες ,servers και εφαρμογές.

Το Storage virtualization φέρνει αποτελεσματικότητα στο περιβάλλον της αποθήκευσης δεδομένων. Με τη συγκέντρωση πολλών αποθηκευτικών πόρων σε έναν ενιαίο, οι διαχειριστές μπορούν να διαχειριστούν όλους τους πόρους ανεξαρτήτως των θέσεων. Αυτό λέγεται ότι μπορεί να επιτρέψει ,ακόμα και 80% ή και περισσότερο σε κάποιες περιπτώσεις , αξιοποίηση των αποθηκευτικών χώρων.

Η διαχείριση των αποθηκευτικών χώρων είναι πολύ πιο εύκολη. Μέσω μιας μοναδικής κονσόλας ένας διαχειριστής μπορεί να ελέγξει τις τάσεις χρήσης αλλά και να κάνει σχέδια ανάπτυξης με μεγαλύτερη σαφήνεια. Έτσι θα μπορέσει να γίνει καλύτερη αναβάθμιση στους αποθηκευτικούς χώρους λόγω του καλύτερου σχεδιασμού και ελέγχου που θα υπάρχει.

Επίσης ,το storage virtualization είναι πολύ ευέλικτο. Με το που αποσυνδεθεί ένα κομμάτι από έναν φυσικό δίσκο ή από συστοιχία δίσκων ,είναι πολύ απλή η αντιγραφή και μετά διαμοίραση αυτού του χώρου μεταξύ διαφόρων συστημάτων . Για παράδειγμα, ένα εικονικό logical unit number (LUN) μπορεί να μεταφερθεί από ένα παλιό σύστημα σε ένα καινούργιο για καλύτερη απόδοση χωρίς να χρειάζεται έξτρα ρυθμίσεις και μετατροπές στην εφαρμογή για να δουλέψει σωστά. Παρόμοια, ένα εικονικό LUN μπορεί να αντιγραφεί σε ένα άλλο τοπικό σύστημα αποθήκευσης για λόγους κράτησης αντιγράφων ασφαλείας εντός του τοπικού δικτύου ή ακόμα και απομακρυσμένα ώστε να υπάρχουν αντίγραφα σε περίπτωση καταστροφής.

Υπάρχουν πολλά στοιχεία που εμπλέκονται στις λύσεις του storage virtualization. Μπορούν να υλοποιηθούν εντός της ίδιας συστοιχίας αποθήκευσης με χρήση προϊόντων ,όπως το Universal Storage Platform V από την Hitachi Data Systems (HDS) ή το EqualLogic PS Family από την Dell, ή μέσω ειδικών συσκευών, όπως το Virtual SAN Appliance (VSA) για το VMware από το LeftHand Networks. Υπάρχουν επίσης προϊόντα που βασίζονται κυρίως στο software του storage virtualization , όπως είναι το SANsymphony software από τη DataCore software, το Invista software από τη EMC

Corporation ή το Veritas Virtual infrastructure από τη Symantec Corporation. Υπάρχουν πολλές εταιρείες για τέτοιες λύσεις που η κάθε μία έχει τα θετικά και τα αρνητικά της, όμως όλες προσπαθούν για την πιο αποτελεσματική λύση στις ανάγκες του πελάτη τους.

Προβλήματα και περιορισμοί του storage virtualization.

Τα πρώτα προβλήματα που αντιμετωπίζουν μικρότερες επιχειρήσεις είναι τα έξοδα εκμάθησης τέτοιου είδους υπηρεσιών όπως και η εγκατάσταση τους. Το κόστος για τον πελάτη συνήθως αρχίζει με αναβάθμιση του τοπικού δικτύου (LAN) ή του storage area network (SAN). Μεγαλύτερη αξιοποίηση του αποθηκευτικού χώρου σημαίνει μεγαλύτερη επισκεψιμότητα από τους χρήστες και μεγαλύτερη αντιγραφή και χρήση των δεδομένων μεταξύ των διαφόρων συστημάτων αποθήκευσης. Έτσι το πιο σημαντικό μέλημα της εταιρείας παροχής των υπηρεσιών storage virtualization, πρέπει να είναι η μέγιστη ετοιμότητα και μηδενικά λάθη του συστήματος. Οι εταιρείες παροχής πρέπει επίσης να προσέξουν πολύ την λειτουργικότητα των συστημάτων τους στο περιβάλλον των πελατών τους.

Παρόλο που υπάρχουν πολλά οφέλη με το storage virtualization υπάρχουν πολλές φορές περιορισμοί, ειδικά όταν το περιβάλλον των αποθηκευτικών συστημάτων είναι heterogeneous (ετερογενή). Δηλαδή όταν μηχανικά μέρη, οι υπολογιστές, τα λειτουργικά συστήματα και οι βάσεις δεδομένων είναι από διαφορετικό προμηθευτή. Σε αντίθεση με τα homogeneous περιβάλλοντα που οτιδήποτε αποτελεί το αποθηκευτικό περιβάλλον είναι π.χ. All-windows ή all- IBM.

Τέλος, όταν όλες οι αξιολογήσεις και οι αναβαθμίσεις έχουν ολοκληρωθεί, το κόστος περνάει στην εγκατάσταση, διαμόρφωση και τον έλεγχο σωστής λειτουργίας του συστήματος storage virtualization. Επίσης θα πρέπει να οι χρήστες να εκπαιδευτούν στον νέο τρόπο χρήσης του αποθηκευτικού σε τέτοιο βαθμό που να μην υπάρχουν προβλήματα στην χρήση του.[10]

2.2.6 Network virtualization (εικονικοποίηση δικτύου)

Στην επιστήμη των υπολογιστών, το network virtualization είναι η διαδικασία που συνδυάζει τους πόρους του υλικού και του λογισμικού ενός υπάρχοντος δικτύου σε ένα virtual network (VN). Αυτό είναι εφικτό με λογισμικά και υπηρεσίες που επιτρέπουν την κοινή χρήση δικτυακών πόρων. Ένα virtual network βλέπει το υλικό αλλά και το λογισμικό στο δίκτυο (routers, switches, etc.) ως ενιαία συλλογή πόρων.

Το network virtualization χρησιμεύει για την διαμοίραση των δικτυακών πόρων από εξουσιοδοτημένους χρήστες αλλά και για τη μίμηση συνδέσεων μεταξύ υπολογιστών, κινητών ακόμα και εφαρμογών, για πρακτικούς λόγους αλλά και για ανάπτυξη εφαρμογών και ελέγχου αυτών με τη μίμηση πραγματικών υλικών, συστημάτων και δικτύων.

Οι κατηγορίες του Network Virtualization:

A. External Network Virtualization ENV (εικονικοποίηση εξωτερικών δικτύων)

Το ENV ενώνει ή υποδιαιρεί ένα ή περισσότερα local area networks (LANs) σε VNs και έτσι η απόδοση και η αποτελεσματικότητα των μεγάλων δικτύων ή κέντρων δεδομένων αυξάνεται σε μεγάλο βαθμό. Ένα VLAN και ένα switch περιέχουν όλα τα απαραίτητα στοιχεία. Ο διαχειριστής του συστήματος μπορεί να ρυθμίσει, ότι συστήματα είναι φυσικά συνδεδεμένα στο ίδιο LAN, σε ξεχωριστά VNs και αντίστροφα, δηλαδή μπορεί να συνδέσει συστήματα από διαφορετικά LANs στο ίδιο VN που θα συνδέει τμήματα ενός μεγάλου δικτύου.

B. Internal Network Virtualization INV (εικονικοποίηση εσωτερικών δικτύων)

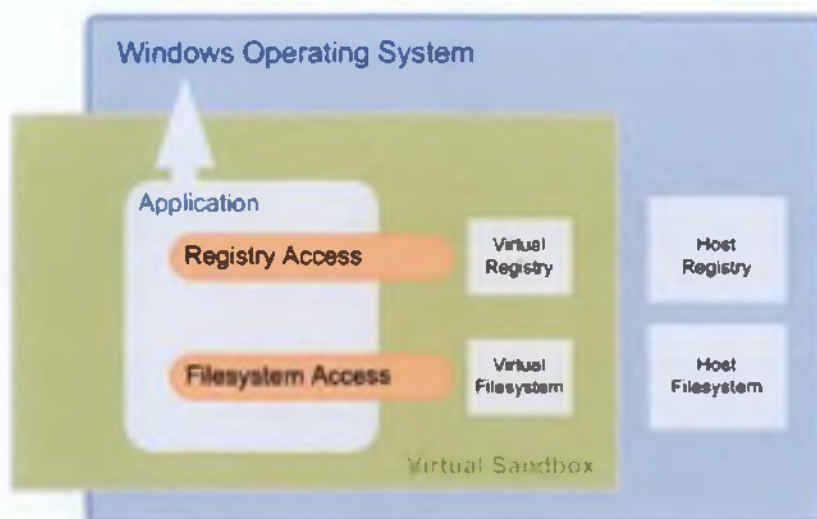
Στο INV υπάρχει ένα μοναδικό σύστημα με containers λογισμικού, όπως το XEN hypervisor control programs. Αυτό μπορεί να βελτιώσει την αποτελεσματικότητα ενός ενιαίου συστήματος με το να απομονώνει τις εφαρμογές σε ξεχωριστά containers.[8][11]

2.2.7 Virtualization of applications (εικονικοποίηση εφαρμογών)

Προγράμματα υπολογιστών ή ακόμα και εφαρμογές μπορούν επίσης να εικονικοποιηθούν. Η εικονικοποίηση εφαρμογών σχετίζεται με ένα είδος διαχωρισμού της εφαρμογής από το εκάστοτε λειτουργικό σύστημα στο οποίο έχει εγκατασταθεί. Αυτού του είδους η εικονικοποίηση περιλαμβάνει τον εγκλεισμό της εφαρμογής σε έναν εικονικό περιέκτη (container) που συμπεριλαμβάνει το μητρώο και τις θέσεις του συστήματος αρχείων του.

Στην ανάπτυξη μιας εφαρμογής περιλαμβάνεται η εγκατάσταση της σε ένα λειτουργικό σύστημα όπως και οι συνεχείς ενημερώσεις στο μητρώο και στο σύστημα

αρχείων του συστήματος. Έτσι η εφαρμογή δεσμεύεται από το εκάστοτε λειτουργικό σύστημα.



Εικόνα 4. Application virtualization

Κάθε εφαρμογή αφαιρείτε από το φιλοξενούμενο λειτουργικό σύστημα χρησιμοποιώντας ένα εικονικό μητρώο και ένα σύστημα αρχείων. Η εφαρμογή δουλεύει απομονωμένη από άλλες εφαρμογές σε ένα δικό της εικονικό περιβάλλον (sandbox), λύνοντας με αυτό τον τρόπο τις μεταξύ τους ασυμβατότητες και επιτρέποντας πολλαπλές εκδόσεις της ίδιας εφαρμογής να εκτελεστούν σε ένα φιλοξενούμενο λειτουργικό σύστημα (εικόνα 4).

Οι εικονικές εφαρμογές μπορούν να είναι φορητές (portable) καθώς βρίσκονται σε απομονωμένα περιβάλλοντα (standalone) έχοντας ενσωματωμένα τα αρχεία του εκτελέσιμου κώδικα (runtime executables) ή συμπιεσμένου περιεχομένου (cab files). Οι βασικοί μέθοδοι εικονικοποίησης εφαρμογών είναι δύο : agent-less και agent-based

- **Agent-less:** Αυτή η μέθοδος περιλαμβάνει τη χρήση ενός ενσωματωμένου λειτουργικού συστήματος το οποίο είναι εγκαταστημένο σαν κομμάτι της εικονικής εφαρμογής. Δηλαδή αυτές οι εικονικοποιημένες εφαρμογές έχουν την δυνατότητα να λειτουργούν ως ένα εκτελέσιμο αρχείο είτε από μια μονάδα δίσκου δικτύου, είτε από μια τοπική μονάδα δίσκου είτε και από ένα USB drive.
- **Agent-based:** Σε αυτή τη μέθοδο υπάρχει μια έτοιμη εφαρμογή σε έναν κεντρικό διακομιστή που από εκεί γίνεται η αποστολή και παράδοση και τοπικά υπάρχει εγκατεστημένος ένας πράκτορας (agent) που στην ουσία είναι ένα πρόγραμμα οδήγησης που αποτελείται από τον πυρήνα (kernel) ή μία υπηρεσία (service).

Ικανότητα να κατευθύνουν την εκάστοτε εφαρμογή μέσω δικτύου στην τοπική συσκευή έχουν και οι δύο παραπάνω μέθοδοι. Η συνεχής ροή (streaming) δίνει την δυνατότητα της άμεσης εκτέλεσης της εφαρμογής από το τελικό μέσο.

Τα πλεονεκτήματα της εικονικοποίησης των εφαρμογών:

- Εφαρμογές που είναι σχεδιασμένες για διαφορετικά λειτουργικά συστήματα μπορούν να αποκτήσουν συμβατότητα με τη βοήθεια κάποιων λογισμικών όπως το Wine που επιτρέπει σε κάποιες εφαρμογές των Windows να δουλεύουν σε Linux.
- Εξοικονόμηση πόρων του συστήματος , αφού χρησιμοποιεί λιγότερους πόρους από ότι η δημιουργία μια εικονικής μηχανής.
- Εγγύηση ότι οι εφαρμογές θα δουλεύουν σωστά με οτιδήποτε ρυθμίσεις και καταχωρήσεις στο μητρώο χωρίς να επηρεάζεται καθόλου το σύστημα.
- Βελτιώνει την ασφάλεια του λειτουργικού συστήματος μιας και οι εικονικές εφαρμογές είναι απομονωμένες από το σύστημα.
- Το migration του λειτουργικού συστήματος απλοποιείται ακόμα περισσότερο.

Προβλήματα της εικονικοποίησης εφαρμογών:

- Δεν είναι εφικτό να εικονικοποιηθούν όλα τα λογισμικά. Πολλά λογισμικά απαιτούν την παρουσία κάποιου προγράμματος συσκευής και 16bit λογισμικά χρειάζονται κοινόχρηστο χώρο μνήμης με το σύστημα.
- Μερικές εφαρμογές επίσης έχουν τέτοιου είδους αρχεία άδειας χρήσης που δεν επιτρέπουν η εφαρμογή να τρέχει σε εικονική μορφή. [4][12]

2.3 Τα οφέλη του Virtualization (Εικονικοποίηση)

Η τεχνολογία της εικονικοποίησης είναι ίσως από τα πιο σημαντικά ζητήματα στον τομέα της πληροφορικής. Η αυξανόμενη συνειδητοποίηση των πλεονεκτημάτων της εικονικοποίησης μας επιτρέπει να εκτελέσουμε πολλαπλό φόρτο εργασίας για το ίδιο υλικό με ένα προβλέψιμο, ασφαλές, και απομονωμένο τρόπο. Υπάρχουν μια σειρά από βασικά επιχειρηματικά οφέλη που οδηγούν τους IT οργανισμούς των επιχειρήσεων να υιοθετήσουν τεχνολογίες εικονικοποίησης. Μερικά από τα πιο σημαντικά είναι:

1) Η μεγαλύτερη παραγωγικότητα του IT προσωπικού.

- Λιγότερα σημεία διαχείρισης
- Λιγότερη συντήρηση του υλικού και λιγότερες απαιτήσεις υποδομής
- Γρηγορότερη εγκατάσταση διακομιστών
- Προσαρμόζονται πιο εύκολα στις ανάγκες της επιχείρησης
- Γρήγορη αντικατάσταση διακομιστών

2) Η μείωση στο λειτουργικό κόστος ενός IT οργανισμού.

- Λιγότερες απαιτήσεις σε άδειες λογισμικών
- Καλύτερη και πιο αποτελεσματική διαχείριση του εξοπλισμού
- Μείωση της κατανάλωσης ενέργειας από την τροφοδοσία έξτρα μηχανημάτων και συστημάτων ψύξης όπως και εξοικονόμηση φυσικού χώρου, που χρειαζόταν για τα φυσικά μηχανήματα

3) Η ενοποίηση των κέντρων δεδομένων.

- Ενοποίηση διακομιστών
- Ενοποίηση σταθμών εργασίας

4) Η ευκολία δοκιμών και ανάπτυξης σε εικονικά εργαστήρια.

5) Οι βελτιώσεις στις επιχειρήσεις.

- Ευκολότερη προσαρμοστικότητα στις απαιτούμενες αλλαγές της επιχείρησης
- Καλύτερη διαθεσιμότητα εφαρμογών
- Καλύτερη προστασία στα δεδομένα και στις υποδομές

6) Μετάβαση στο νέφος.

3. Cloud computing (Υπολογιστικό νέφος)

Με τον όρο “Cloud Computing” ή “Υπολογιστικό Νέφος” εννοούμε την παροχή υπολογιστικών πόρων ή/και υπηρεσιών μέσω ενός δικτύου. Οι υπηρεσίες Cloud Computing κάνουν δυνατή την ευέλικτη διάθεση τεχνολογικού κεφαλαίου των παρόχων στους τελικούς χρήστες, μέσω της δικτυακής παροχής πρόσβασης σε ένα σύνολο παραμετροποιήσιμων υπολογιστικών πόρων. Υπολογιστικούς πόρους ονομάζουμε κάθε φυσικό ή εικονικό τμήμα ενός συστήματος υπολογιστών, καθώς και κάθε συσκευή που είναι συνδεδεμένη σε αυτό. Στους υπολογιστικούς πόρους που διατίθενται μέσω νέφους περιλαμβάνονται αποθηκευτικοί χώροι, χρόνος σε CPU, μνήμη, αρχεία, δίκτυα, servers, εφαρμογές και υπηρεσίες.

3.1 Εισαγωγή

Πριν εμφανιστεί το νέφος η ανάπτυξη εφαρμογών βασιζόταν αποκλειστικά σε ιδίους πόρους των χρηστών. Οι πάροχοι των υπηρεσιών ανέπτυσαν τις εφαρμογές τους σε δικό τους εξοπλισμό και στη συνέχεια οι χρήστες των εφαρμογών εγκαθιστούσαν το λογισμικό στα δικά τους μηχανήματα. Παρότι με αυτόν τον τρόπο εξασφαλίζεται ανεξαρτησία και πλήρης έλεγχος των δεδομένων τόσο του παρόχου όσο και των τελικών χρηστών, δημιουργούνται ταυτόχρονα πολλά προβλήματα.

Για την παροχή υπηρεσιών απαιτείται κατ’ αρχάς μεγάλο αρχικό κεφάλαιο το οποίο αφορά στο λογισμικό στο οποίο θα αναπτυχθεί η υπηρεσία, datacenters, servers, εγκατάσταση δικτύου, βάσεις δεδομένων καθώς και προσωπικό για την λειτουργία και ρύθμιση όλων των παραπάνω. Το κόστος συντήρησης είναι επίσης μεγάλο, καθώς είναι απαραίτητο να αναπτυχθούν μηχανισμοί ανάκτησης δεδομένων σε περίπτωση failover, να είναι εγγυημένη η ασφάλεια των δεδομένων των χρηστών και να εξασφαλίζεται διαρκής ενημέρωση και έλεγχος του συστήματος ώστε νέες εκδόσεις λογισμικού να μην θέτουν σε κίνδυνο την ομαλή λειτουργία της υπηρεσίας. Συχνά δεν χρησιμοποιείται ολόκληρος ο εξοπλισμός ή απαιτούνται περισσότερα μηχανήματα από τα διαθέσιμα προκειμένου να καλυφθούν οι ανάγκες των χρηστών. Στην πρώτη περίπτωση πόροι μένουν αναξιοποίητοι ενώ στη δεύτερη οι χρήστες δεν λαμβάνουν ομαλές και σίγουρες υπηρεσίες.

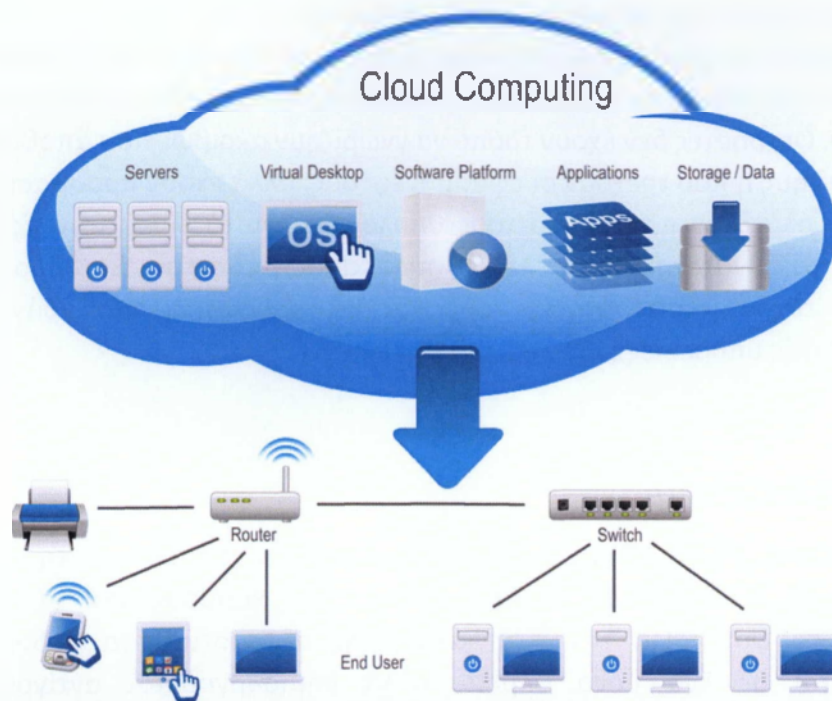
Για τους παραπάνω λόγους ήταν απαραίτητη μια αλλαγή στον τρόπο παροχής υπηρεσιών. Το cloud παρέχει λύσεις στους παραπάνω περιορισμούς ανάπτυξης εφαρμογών με ιδίους πόρους, εξασφαλίζοντας παράλληλα την ασφάλεια των δεδομένων των τελικών χρηστών και των παρόχων των υπηρεσιών. Το cloud, μία αναπαράσταση του οποίου δίνεται από την Εικόνα 5 (σκίτσο) και λίγο πιο αναλυτικά από την Εικόνα 6, προσφέρει τη δυνατότητα της απομακρυσμένης παροχής

εφαρμογών και υπηρεσιών. Χρησιμοποιείται τόσο για την ανάπτυξη των υπηρεσιών, όσο και για την προσφορά τους στους τελικούς χρήστες. Όσον αφορά στην ανάπτυξη των εφαρμογών, το cloud χρησιμοποιείται για εξοικονόμηση πόρων. Οι πάροχοι των υπηρεσιών αναπτύσσουν τις εφαρμογές τους, όχι σε ίδιο εξοπλισμό, αλλά σε λογισμικό που τρέχει σε απομακρυσμένα μηχανήματα. Με αυτή τη τεχνική η εγκατάσταση και διαχείριση του απαραίτητου λογισμικού πραγματοποιείται χωρίς τη συμμετοχή του παρόχου της αναπτυσσόμενης εφαρμογής, αλλά από την πλευρά του cloud, δηλαδή από τον πάροχο cloud computing. Ταυτόχρονα τα δεδομένα δεν διατηρούνται σε μηχανήματα του παρόχου, αλλά απομακρυσμένα, διαμοιραζόμενα datacenters. Σε αυτά έχουν αναπτυχθεί αυτόματοι μηχανισμοί failover και ασφάλειας των δεδομένων, καθώς και εργαλεία δυναμικής παραχώρησης μηχανημάτων ανάλογα με τις εκάστοτε απαιτήσεις των παρόχων σε υπολογιστικούς πόρους.

Το νέφος χρησιμοποιείται επίσης για την διάθεση υπηρεσιών στους τελικούς χρήστες. Μέσω του cloud οι εφαρμογές τρέχουν απομακρυσμένα, χωρίς ανάγκη εγκατάστασης λογισμικού στον εξοπλισμό των τελικών χρηστών. Οι ενημερώσεις των εφαρμογών, η αναβάθμιση και συντήρησή τους δεν γίνεται στην πλευρά του χρήστη, αλλά από τον πάροχο. Οι χρήστες αρκεί να συνδεθούν στο δίκτυο, από οποιαδήποτε συσκευή, PC, tablet, Smartphone, και άμεσα μπορούν να χρησιμοποιήσουν τις εφαρμογές. Όλα τα παραπάνω χαρακτηριστικά του cloud, οι διαφορετικοί τύπου υπηρεσιών που παρέχονται καθώς και κάποια ζητήματα που εγείρονται κατά τη χρήση του, αναλύονται στις επόμενες ενότητες.



Εικόνα 5. Σκίτσο για το cloud computing



Εικόνα 6. Cloud Computing

3.2 Χαρακτηριστικά του Cloud computing (Υπολογιστικό νέφος)

Όπως αναφέρθηκε, ο κλάδος παροχής υπηρεσιών αντιμετωπίζει διάφορα προβλήματα. Από την πλευρά των παρόχων των υπηρεσιών απαιτείται μεγάλο αρχικό κεφάλαιο, ειδικευση σε θέματα ασφαλείας των δεδομένων, ανάπτυξη μηχανισμών που εξασφαλίζουν την αξιοπιστία της υπηρεσίας, διασύνδεση των δεδομένων ώστε να τρέχει η υπηρεσία σε πολλές πλατφόρμες και διατήρηση εξοπλισμού και προσωπικού. Από την πλευρά των τελικών χρηστών είναι απαραίτητη η εγκατάσταση της υπηρεσίας σε κάθε μηχανήμά τους, η αγορά λογισμικού και η διαρκής ενημέρωση των εφαρμογών τους. Παρακάτω παρουσιάζονται τα χαρακτηριστικά του cloud τα οποία δίνουν λύσεις σε όλα τα παραπάνω προβλήματα.

Scalability (Επεκτασιμότητα)

Στο cloud είναι δυνατή η χειροκίνητη ή δυναμική προσθήκη και αφαίρεση κόμβων εκτέλεσης εργασιών ανάλογα με την αυξομείωση των απαιτήσεων, χωρίς να αλλοιώνεται η υπηρεσία που παρέχεται στο χρήστη. Όσον αφορά στους πόρους, αυτοί εξοικονομούνται καθώς τα μηχανήματα χρησιμοποιούνται μόνο όταν χρειάζονται.

Με τη μέθοδο της εικονικοποίησης οι λεπτομέρειες υλοποίησης και κατάστασης στην οποία βρίσκονται οι πόροι αποκρύπτονται από το χρήστη. Δίνεται η εντύπωση του ενιαίου κάτι που μπορεί να μην ισχύει, όπως στην περίπτωση των καταναμημένων συστημάτων. Οι χρήστες δεν έχουν τρόπο να γνωρίζουν ακριβώς που αποθηκεύονται τα δεδομένα τους ή που τρέχουν οι εφαρμογές τους, αλλά έχουν πρόσβαση σε αυτά σαν να ήταν όλα στον προσωπικό τους υπολογιστή. Αυτό δίνει στον πάροχο της υπηρεσίας τη δυνατότητα να ελέγξει πως αποθηκεύονται τα δεδομένα, να αξιοποιήσει τυχόν ιδιαιτερότητές τους και να προσφέρει πιο γρήγορες και αποτελεσματικές υπηρεσίες.[13]

Reliability (Αξιοπιστία)

Το cloud εγγυάται την αποθήκευση των δεδομένων των χρηστών και την αξιόπιστη μεταφορά τους. Τα δεδομένα αποθηκεύονται στους κόμβους από τους οποίους αποτελείται το cloud. Για την εξασφάλιση της αξιοπιστίας, τα δεδομένα δεν αποθηκεύονται σε ένα μόνο σημείο, αλλά δημιουργούνται αντίγραφα των δεδομένων είτε στον ίδιο κόμβο είτε σε άλλους κόμβους του cloud. Ο συνηθισμένος αριθμός αντιγράφων είναι τρία. Επίσης χρησιμοποιούνται μηχανισμοί ανάκτησης δεδομένων σε περίπτωση απώλειας ή αστοχίας ενός κόμβου αποθήκευσης.

Για την μέτρηση της αξιοπιστίας χρησιμοποιούνται δείκτες όπως ο MTBF, Mean Time Between Failures, δηλαδή ο μέσος χρόνος μεταξύ δύο διαδοχικών αποτυχιών του συστήματος, και ο MTTF, Mean Time to Failure, δηλαδή ο μέσος χρόνος μέχρι την αποτυχία του συστήματος. Ο MTBF χρησιμοποιείται για συστήματα στα οποία υπάρχει μηχανισμός ανάκτησης και ο MTTF χρησιμοποιείται για συστήματα στα οποία δεν υπάρχει μηχανισμός ανάκτησης, αλλά μηχανισμός αντικατάστασης.

$$MTBF = \frac{\sum (\text{start of downtime} - \text{start of uptime})}{\text{number of failures}}$$

Π.χ.

Εικόνα 7. Εξίσωση του MTBF

Downtime = Η χρονική στιγμή που το σύστημα απέτυχε.

Uptime = Η χρονική στιγμή που το σύστημα επανήλθε ακριβώς πριν το downtime.
[40]

Maintenance (Συντήρηση)

Με τον όρο συντήρηση εννοούμε την διόρθωση των εφαρμογών από προγραμματιστικά λάθη και την εγκατάσταση νέων εκδόσεων της εφαρμογής. Με χρήση του cloud η συντήρηση των εφαρμογών είναι ευκολότερη. Οι χρήστες δεν ασχολούνται οι ίδιοι με την εγκατάσταση των ενημερώσεων σε κάθε συσκευή τους

ξεχωριστά. Από όπου και αν τρέξει η εφαρμογή, προσωπικός υπολογιστής, tablet, Smartphone, είναι στη δικαιοδοσία του παρόχου να προσφέρει συμβατό λογισμικό, χωρίς καμία ενέργεια από την πλευρά του χρήστη. Ταυτόχρονα προγραμματιστικά λάθη μπορούν άμεσα να διορθωθούν σε όλους τους χρήστες.

Performance (Απόδοση)

Η απόδοση αναφέρεται σε διάφορα κριτήρια όπως χαμηλό χρόνο απόκρισης, υψηλή διαμεταγωγή δεδομένων, χαμηλή χρήση υπολογιστικών πόρων, υψηλή διαθεσιμότητα του συστήματος, γρήγορη επεξεργασία δεδομένων και υψηλό εύρος ζώνης του δικτύου. Χρησιμοποιώντας το cloud ο τελικός χρήστης μπορεί να παρακολουθεί την απόδοση των εφαρμογών του. Οι πάροχοι των εφαρμογών έχουν στη διάθεσή τους μέσω του cloud καλύτερα μηχανήματα και περισσότερα εργαλεία βελτίωσης της απόδοσης, με αποτέλεσμα να παρέχουν πιο αποδοτικές υπηρεσίες. Οι τελικοί χρήστες μέσω των αναβαθμίσεων που γίνονται είτε σε επίπεδο υποδομής του cloud είτε σε επίπεδο βελτίωσης της εφαρμογής από τον πάροχο έχουν αυτόματα βελτιωμένη απόδοση.[15]

Multi-tenancy (Πολύ-μίσθωση)

Κάθε χρήστης μιας υπηρεσίας cloud δεν χρειάζεται να διαθέτει δικό του μοναδικό αντίγραφο της εφαρμογής. Αρκεί ένα μοναδικό στιγμιότυπο (instance) της εφαρμογής το οποίο είναι ευέλικτο και μπορεί να προσαρμοστεί στην ανάγκες του κάθε χρήστη. Αυτό έχει ως αποτέλεσμα την εξοικονόμηση πόρων στο cloud, την ευκολότερη συντήρηση της εφαρμογής. Ταυτόχρονα διευκολύνεται η εξόρυξη δεδομένων (data mining). Οι πληροφορίες όλων των χρηστών είναι αποθηκευμένες σε ένα κεντρικό σημείο, χωρίς να υπολογιστούν οι μηχανισμοί αντιγραφής των δεδομένων για αξιοπιστία, το οποίο κάνει ευκολότερη την επεξεργασία των δεδομένων για εξαγωγή στατιστικών αποτελεσμάτων και πληροφοριών χρήσιμων στον πάροχο της υπηρεσίας.

Στην ασφάλεια συμπεριλαμβάνονται όλοι οι μηχανισμοί που προστατεύουν τα δεδομένα και τα μηχανήματα από ακούσια και μη εξουσιοδοτημένη πρόσβαση, αλλαγή ή καταστροφή, καθώς και από φυσικές καταστροφές. Στο cloud η ασφάλεια που παρέχεται είναι καλύτερη λόγω της κεντρικής αποθήκευσης των δεδομένων (centralization) και της προσφοράς περισσότερων εργαλείων που στοχεύουν στην ασφάλεια των δεδομένων. Σε κατανεμημένα συστήματα αποθήκευσης οι μηχανισμοί που εγγυούνται την ασφάλεια των δεδομένων είναι πιο πολύπλοκοι και δυσκολεύουν πολύ την παρακολούθηση του αρχείου καταγραφής ελέγχου (audit log). Μέσω του cloud οι χρήστες δεν χρειάζεται να ασχολούνται οι ίδιοι με την πολύπλοκη ασφάλεια των δεδομένων τους. Υπάρχουν και τα ιδιωτικά cloud στα οποία οι χρήστες έχουν έλεγχο πάνω στην υποδομή και σε ό,τι αφορά στην ασφάλεια της πληροφορίας, όπως ποιος έχει πρόσβαση στα δεδομένα, τι χρήση γίνεται.

Ταυτόχρονα επιλέγονται και οι χειρισμοί για την ασφάλεια των προσωπικών και ευαίσθητων δεδομένων των χρηστών.[16]

Cloud API, Application Programmina Interface (Διεπαφή προγραμματισμού εφαρμογών)

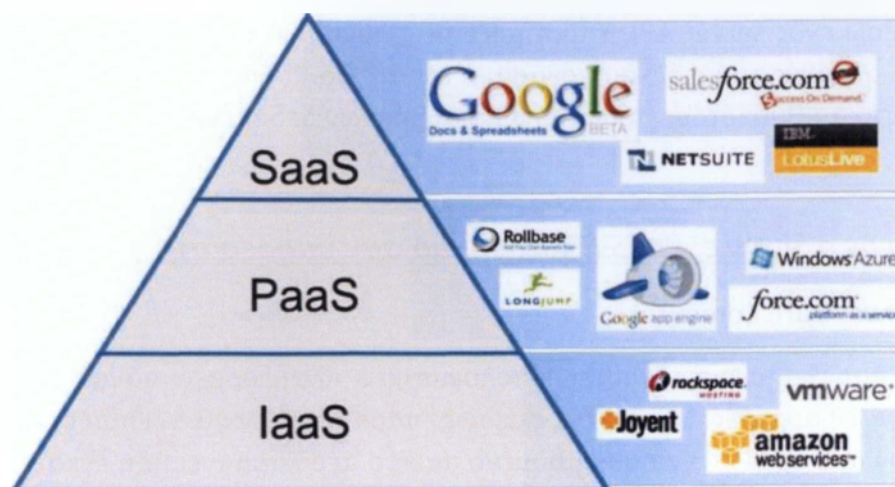
API ονομάζουμε τη διεπαφή που χρησιμοποιούν οι εφαρμογές για να επικοινωνήσουν με το χρήστη ή άλλες εφαρμογές και μηχανήματα. Τα cloud APIs χρησιμοποιούνται για την επικοινωνία του λογισμικού με τις υπηρεσίες μέσω διεπαφών (interfaces). Τα cloud APIs συνήθως χρησιμοποιούν αρχιτεκτονικές τύπου REST. Απομακρύνονται από την λογική της αρχιτεκτονικής SOAP στην οποία κάθε λειτουργία πρέπει να ξαναγραφτεί και να σταλεί με XML αρχεία προς αποκωδικοποίηση. Η αρχιτεκτονική τύπου REST προσφέρει επεκτασιμότητα και ανεξάρτητη ανάπτυξη των components του συστήματος.

On demand self-service

Ένα ακόμα χαρακτηριστικό του Νέφους είναι το “on demand self-service”, το οποίο δίνει τη δυνατότητα στους χρήστες γρήγορα και εύκολα να δεσμεύουν και να παραμετροποιήσουν μηχανήματα και να αναπτύξουν εφαρμογές, χωρίς να είναι απαραίτητη η ανθρώπινη αλληλεπίδραση με υπευθύνους του cloud. Οι πάροχοι cloud προτρέπονται να δημιουργήσουν templates, πρότυπα, τα οποία, ύστερα από σύγκριση με άλλα, οδηγούν τους χρήστες να επιλέξουν γρήγορα τις υπηρεσίες που ταιριάζουν καλύτερα στις ανάγκες τους. Τα πρότυπα διαθέτουν προκαθορισμένα το λειτουργικό τους σύστημα, τη βάση δεδομένων, ρυθμίσεις ασφαλείας και υπηρεσίες web, καθώς και οδηγίες για εύκολη χρησιμοποίηση διαφορετικών cloud και παραμετροποίηση των εφαρμογών να ταιριάζουν σε αυτά. Διατίθενται επίσης εργαλεία για μετακίνηση εφαρμογών από το ένα περιβάλλον cloud στο άλλο.

3.3 Cloud Computing Services (Υπηρεσίες)

Σε αυτή την ενότητα παρουσιάζονται τα διαφορετικά είδη υπηρεσιών cloud computing που έχουν αναπτυχθεί μέχρι σήμερα. Ο διαχωρισμός των υπηρεσιών γίνεται με βάση το αφαιρετικό επίπεδο ελέγχου των χρηστών στους υπολογιστικούς πόρους που τους παρέχονται. Όπως φαίνεται στην Εικόνα 8, το υψηλότερο αφαιρετικό επίπεδο παροχής υπηρεσιών αποτελεί το SaaS. Σε αυτό παρέχονται στους τελικούς χρήστες εφαρμογές, τις οποίες μπορούν να τις παραμετροποιήσουν ανάλογα με τις ανάγκες τους, χωρίς κανέναν έλεγχο στον τρόπο εκτέλεσής τους.



Εικόνα 8. Cloud computing services

Infrastructure as a Service, IaaS (Υποδομή σαν υπηρεσία)

πχ: Amazon Cloud-Formation.

Σε αυτή την υπηρεσία παρέχεται στους χρήστες υποδομή, δηλαδή υπολογιστές, servers και υπολογιστικοί πόροι, είτε ως μηχανήματα ή πιο συχνά ως εικονικές μηχανές.

Ο πάροχος της υπηρεσίας είναι υπεύθυνος για την συντήρηση των μηχανημάτων στα οποία τρέχουν οι εικονικές μηχανές των χρηστών. Φροντίζει για την ασφάλεια των δεδομένων των χρηστών, την αξιοπιστία των μηχανημάτων ώστε σε περίπτωση αποτυχίας ενός κόμβου να μην αλλοιωθεί η απόδοση της υπηρεσίας και την επεκτασιμότητά τους ώστε δυναμικά να δίνονται στους χρήστες οι πόροι που χρειάζονται. Συνήθως εφαρμόζεται πολιτική πληρωμής ανάλογη με την ποσότητα των πόρων που δεσμεύονται και χρησιμοποιούνται.

Οι χρήστες έχουν τη δυνατότητα να ορίσουν μερικώς ή πλήρως τα χαρακτηριστικά των μηχανημάτων που χειρίζονται, CPU, RAM, σκληρός δίσκος, λειτουργικό σύστημα, δίκτυο και άλλα. Έχουν τη δυνατότητα να χειριστούν τα μηχανήματα με απόλυτη ελευθερία, να εγκαταστήσουν λογισμικό, να στήσουν βάσεις δεδομένων και να υλοποιήσουν εφαρμογές ακριβώς όπως θα έκαναν σε δικά τους μηχανήματα. Το περιβάλλον πρέπει να είναι επίσης ικανό να χρησιμοποιηθεί για υλοποίηση υπηρεσιών PaaS και SaaS.

Ένα παράδειγμα αποτελεί το Rackspace Cloud, το οποίο ήταν ένας από τους πρώτους παρόχους IaaS. Διαθέτει τρεις κύριες υπηρεσίες, Cloud Files, Cloud Servers και Cloud Sites. Τα Cloud Files αποτελούν τον αποθηκευτικό χώρο των χρηστών. Η πρόσβαση σε αυτά γίνεται μέσω API ανοιχτού κώδικα. Τα δεδομένα κρατούνται σε τρία αντίγραφα. Με τους Cloud Servers οι χρήστες έχουν τη δυνατότητα να σηκώσουν

servers μέσω ενός server API. Υποστηρίζεται η δυναμική διάθεση μηχανημάτων σε απότομες αλλαγές του φόρτου εργασίας. Με το Cloud Sites οι χρήστες διατηρούν απεριόριστο αριθμό ιστοσελίδων, emails και βάσεων δεδομένων.

Platform as a Service, PaaS (Πλατφόρμα σαν υπηρεσία)

πχ: Windows Azure Compute.

Το PaaS παρέχει στους χρήστες υπολογιστικές πλατφόρμες πάνω στις οποίες υλοποιούν εφαρμογές. Είναι το αμέσως παραπάνω λογικό επίπεδο παροχής υπηρεσιών από το IaaS. Ένα συνηθισμένο πακέτο περιλαμβάνει ήδη εγκατεστημένο λειτουργικό σύστημα, περιβάλλον προγραμματισμού, βιβλιοθήκες, βάση δεδομένων, δίκτυα και έναν web server. Προσφέρονται εργαλεία σχεδίασης των εφαρμογών και testing (ελέγχου) και υπηρεσίες σχεδιασμού βάσεων δεδομένων, υπηρεσιών web, application versioning, αποθήκευσης, ασφάλειας, αξιοπιστίας των δεδομένων, αυτόματου scaling ανάλογα με τις απαιτήσεις της εφαρμογής, διαχείρισης της κατάστασης των user interfaces και συνεργασίας απομακρυσμένων ομάδων ανάπτυξης μιας εφαρμογής.

Με το PaaS οι χρήστες αναπτύσσουν εφαρμογές που έχουν δημιουργήσει ή εφαρμογές που έχουν αποκτήσει, με τα εργαλεία και τη γλώσσα προγραμματισμού που υποστηρίζονται από τον πάροχο, αλλά δεν έχουν κανένα έλεγχο πάνω στην υποδομή. Δεν χρειάζεται να διατηρούν μηχανήματα και λογισμικό, κάτι που μειώνει σημαντικά το κόστος ανάπτυξης εφαρμογών. Προβλήματα συμβατότητας των εκδόσεων λογισμικού, ανάπτυξης μηχανισμών ασφαλείας για τα δεδομένα των χρηστών και χειροκίνητης δέσμευσης πόρων ανάλογα με τη χρήση της εφαρμογής εξαλείφονται. Τέλος η ποικιλία στους παρόχους βοηθάει στην επιλογή του κατάλληλου για την ανάπτυξη των επιθυμητών εφαρμογών.

Software as a Service, SaaS (Λογισμικό σαν υπηρεσία)

πχ: Google apps.

Το SaaS, γνωστό και ως “on-demand software”, παρέχει στους χρήστες έτοιμες εφαρμογές, οι οποίες τρέχουν στο cloud και είναι προσβάσιμες μέσω ενός δικτύου. Είναι το αμέσως παραπάνω λογικό επίπεδο στην παροχή υπηρεσιών από το PaaS. Λογισμικό και δεδομένα βρίσκονται κεντρικά στο cloud και διατίθενται στους χρήστες μέσω ενός web browser. Ο πάροχος είναι υπεύθυνος για τη διάθεση εφαρμογών, τις οποίες προσφέρει στους χρήστες on-demand. Χρησιμοποιείται multi-tenant αρχιτεκτονική, με ένα μοναδικό version της εφαρμογής να εξυπηρετεί πολλούς χρήστες. Κάποιοι πάροχοι επιτρέπουν στους χρήστες την παραμετροποίηση των εφαρμογών. Οι ενημερώσεις και η συντήρηση των εφαρμογών γίνονται κεντρικά με αποτέλεσμα οι χρήστες να έχουν αυτόματα τις νέες εκδόσεις. Οι πάροχοι διαθέτουν μηχανισμούς scaling με αποτέλεσμα ο φόρτος εργασίας να κατανέμεται ανάλογα με

τις εκάστοτε απαιτήσεις. Καθώς το λογισμικό μπορεί να χρησιμοποιηθεί από διάφορες συσκευές, υπολογιστής, tablet, Smartphone, οι πάροχοι είναι υπεύθυνοι για την ανάπτυξη συμβατών εκδόσεων λογισμικού. Το SaaS προσφέρει αξιοπιστία και ασφάλεια των δεδομένων των χρηστών.

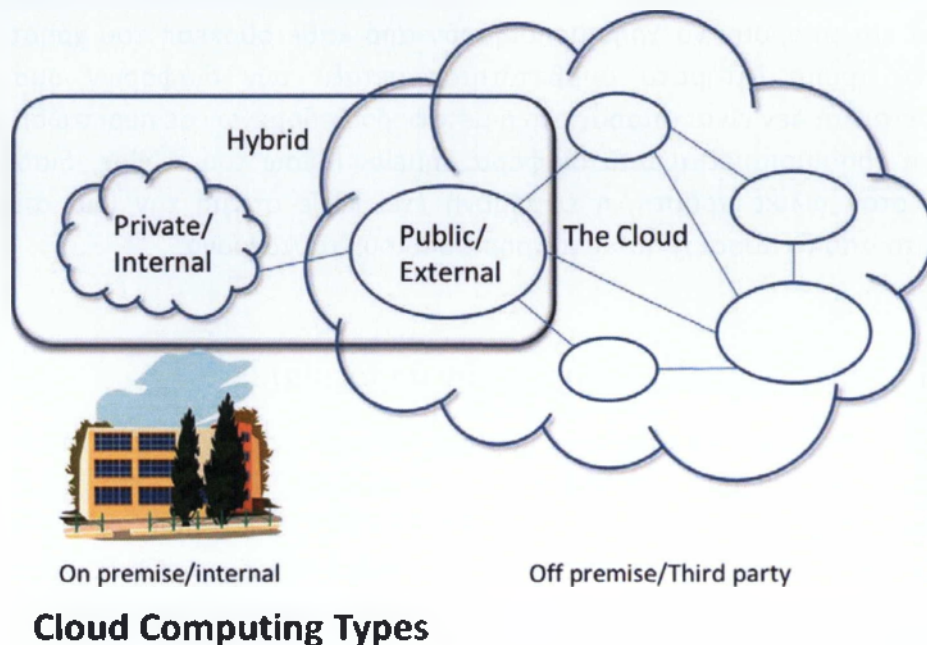
Οι χρήστες δεν έχουν τον έλεγχο της υποδομής ή τις πλατφόρμες πάνω στις οποίες τρέχουν οι εφαρμογές. Με το SaaS δεν απαιτείται πλέον η εγκατάσταση και διαρκής ενημέρωση του λογισμικού ούτε η δέσμευση αποθηκευτικού χώρου για αποθήκευση των δεδομένων σε κάθε συσκευή του χρήστη. Οι εφαρμογές αρκεί να νοικιαστούν μία φορά και μπορούν να χρησιμοποιηθούν από κάθε συσκευή του χρήστη. Με αυτόν τον τρόπο ζητήματα συμβατότητας μεταξύ των διάφορων συσκευών εξαλείφονται και δεν είναι απαραίτητη η μεταφορά δεδομένων σε περίπτωση που η εφαρμογή χρησιμοποιείται από διάφορα σημεία. Μέσω του scaling, διαδικασία διαφανή στον τελικό χρήστη, η εφαρμογή έχει κάθε στιγμή την ίδια απόδοση ανεξάρτητα από το πόσοι χρήστες τη χρησιμοποιούν ταυτόχρονα.

Network as a Service, NaaS (Δίκτυο σαν υπηρεσία)

Το NaaS αποτελεί μια νέα υπηρεσία παροχής υπηρεσιών μέσω cloud. Με το NaaS παρέχεται στους χρήστες υποδομή και υπηρεσίες δικτύων. Στόχος είναι η αποδοτικότερη χρήση της υποδομής του δικτύου ενός datacenter. Στην υπηρεσία περιλαμβάνονται η γνώση της τοπολογίας του δικτύου, ώστε χρήστες να μπορούν να επιλέξουν τους κόμβους που θα αποθηκεύουν τα δεδομένα τους, ο έλεγχος των προωθούμενων πακέτων στα switches του δικτύου, ορισμός firewalls από τους χρήστες και επεξεργασία των δεδομένων εσωτερικά του δικτύου, ώστε να περιορίζετε η κίνηση στο δίκτυο. Για υλοποίηση του NaaS είναι απαραίτητη μια multi-tenant αρχιτεκτονική και απομόνωση των πόρων που παρέχονται στους χρήστες. Παρέχεται επίσης API για την εύκολη πρόσβαση του χρήστη στις διάφορες υπηρεσίες του NaaS παρόχου. Το NaaS αποτελεί αρκετά νέα τεχνολογία.

3.4 Cloud Computing types (Είδη)

Υπάρχουν τρία είδη cloud computing και κατηγοριοποιούνται ανάλογα με τον τρόπο που ελέγχονται από τους χρήστες αλλά και από τις υποδομές που χρησιμοποιούνται. Πιο συγκεκριμένα είναι το private cloud, το public cloud και το hybrid cloud (συνδυασμός των άλλων δύο). Στην παρακάτω εικόνα, με ένα απλό σχήμα διατυπώνεται η διαφορά των τριών ειδών του cloud computing.



Εικόνα 9. Είδη του cloud computing

Private Cloud (Ιδιωτικό νέφος)

Το ιδιωτικό νέφος είναι μια cloud υπηρεσία που λειτουργεί αποκλειστικά για μια οργάνωση και είτε διαχειρίζεται μέσα από την ίδια την οργάνωση, είτε είναι υπεύθυνοι εξωτερικοί συνεργάτες για την διαχείριση της cloud υπηρεσίας. Το να δημιουργηθεί ένα ιδιωτικό cloud απαιτεί μεγάλου βαθμού δέσμευση όπως και απαιτείται να επανεκτιμηθούν οι υπάρχοντες πόροι. Αν γίνει σωστά ένα ιδιωτικό νέφος μπορεί να βοηθήσει σημαντικά στην διαχείριση του οργανισμού ή της επιχείρησης αλλά πρέπει να υπάρχει μεγάλη προσοχή σε ότι αφορά την ασφάλεια για να αποφευχθούν τυχόν προβλήματα. Τα αυτό-διαχειριζόμενα ιδιωτικά νέφη θέλουν αρκετό κεφάλαιο για την δημιουργία τους και την διαχείριση τους αλλά και αρκετό χώρο, κάτι που έχει προσελκύσει επικρίσεις καθώς δεν αντιπροσωπεύει καθόλου το μοντέλο του cloud που είναι πιο οικονομικό αλλά και απαιτεί λιγότερα χέρια στην διαχείριση του. Παρόλα αυτά το private cloud είναι πολύ ενδιαφέρον σαν ιδέα. [17][18]

Public cloud (Δημόσιο νέφος)

πχ. Amazon AWS, Microsoft, Google.

Ένα νέφος ονομάζεται δημόσιο όταν οι υπηρεσίες του είναι προσβάσιμες από το ευρύ κοινό. Μερικά public clouds είναι δωρεάν και άλλα επί πληρωμή. Τεχνικά, από αρχιτεκτονικής άποψης μπορεί να μην υπάρχει καμία διαφορά μεταξύ ενός public και ενός private cloud, όμως από την πλευρά της ασφάλειας για τις υπηρεσίες μπορεί να υπάρχει και μάλιστα μεγάλη. Οι υπηρεσίες παρέχονται αποκλειστικά μέσω δικτύου.[15][22]

Hybrid cloud (Υβριδικό νέφος)

Τελευταίο διακρίνεται το υβριδικό νέφος. Το υβριδικό νέφος αποτελείται από τον συνδυασμό των public και private clouds και έτσι συνδυάζει και τα πλεονεκτήματα του κάθε είδους. Για την υλοποίηση απαιτείται και ιδιωτική αλλά και απομακρυσμένη υποδομή server και υπάρχει η δυνατότητα, ανάλογα τις ανάγκες, να λειτουργεί και τοπικά αλλά και με public clouding. [23]

3.5 Προβλήματα και περιορισμοί του Cloud Computing

Η χρήση του cloud computing έχει πολλά θετικά αλλά κίνδυνοι, όπως η παραβίαση των προσωπικών αρχείων που υπάρχουν σε cloud υπηρεσίες, είναι υπαρκτοί. Τέτοια ζητήματα θα παρουσιαστούν παρακάτω.

Ασφάλεια δεδομένων:

Η ασφάλεια είναι ένα από τα πιο σημαντικά θέματα του cloud computing. Επειδή βασίζεται εξ ολοκλήρου στο διαδίκτυο, καθίσταται αρκετά ευάλωτο σε επιθέσεις hacking. Το επίπεδο ευπάθειας είναι σχεδόν το ίδιο με οπουδήποτε αλλού όσο αφορά αυτές τις επιθέσεις, όμως λόγω του ότι στο cloud computing υπάρχουν κατανεμημένα δίκτυα και τα αρχεία είναι προστατευμένα και σε πολλαπλά σημεία αποθηκευμένα, οι εταιρείες εύκολα μπορούν να επανέλθουν από τυχόν ζημιά προερχόμενη από hacking. Γι' αυτό πριν χρησιμοποιήσουμε κάποια τέτοια υπηρεσία πρέπει να διαβάζουμε της πολιτικές ασφαλείας του κάθε παρόχου.

Προβλήματα συμβατότητας:

Ένα άλλο πρόβλημα με τις cloud υπηρεσίες είναι η συμβατότητα με όλα τα συστήματα πληροφορικής. Είναι παντού γνωστό ότι το cloud computing είναι το πιο αποδοτικό μέσο, ειδικά για εταιρείες. Παρόλα αυτά το πρόβλημα προκύπτει από το γεγονός πως η εταιρεία θα χρειαστεί να αντικαταστήσει πολλά από την υπάρχουσα υποδομή της, ώστε να αποκτήσει απόλυτα συμβατά συστήματα IT. Το hybrid cloud συνήθως μειώνει τα θέματα συμβατότητας κατά πολύ.

Κόστος:

Ενώ μακροπρόθεσμα, η φιλοξενία "σύννεφων" είναι πολύ φτηνότερη από τις παραδοσιακές τεχνολογίες, το γεγονός ότι είναι καινούργια και πρέπει να ερευνηθεί και να βελτιωθεί την καθιστά ακριβότερη.

Τα κέντρα δεδομένων πρέπει να αγοράσουν ή να αναπτύξουν το λογισμικό που θα τρέξει το "σύννεφο", θα πρέπει να ξανακάνουν εγκατάσταση τις μηχανές και έτσι μπορεί να δημιουργηθούν απρόβλεπτα προβλήματα (που πάντα προκύπτουν), άρα και επιπλέον κόστος.

Ευελιξία:

Αυτό είναι μόνο ένα προσωρινό πρόβλημα, αλλά οι τρέχουσες τεχνολογίες είναι ακόμα στα εξεταστικά στάδια, έτσι δεν προσφέρουν πραγματικά την ευελιξία που υπόσχονται. Φυσικά, αυτό θα αλλάξει στο μέλλον, αλλά μερικοί από τους τρέχοντες χρήστες πρέπει να ξέρουν ότι ο κεντρικός "υπολογιστής σύννεφων" τους είναι δύσκολο ή αδύνατο να αναβαθμιστεί χωρίς απώλεια μερικών στοιχείων.

4.Υλοποίηση περίπτωσης Virtualization & Cloud computing

Στα προηγούμενα κεφάλαια παρουσίασα τις τεχνολογίες virtualization και cloud computing. Σε αυτό το κεφάλαιο θα παρουσιάσω τα παραπάνω, με ένα παράδειγμα μίας περίπτωσης εφαρμογής τους. Θα δημιουργήσω μια χαρακτηριστική περίπτωση μεγάλης διαθεσιμότητας υπηρεσιών cloud χρησιμοποιώντας τεχνολογίες του virtualization σε όλη τη διαδικασία με απώτερο σκοπό: ανάπτυξη πλατφόρμας η οποία να επιτρέπει desktop virtualization με χρήση cloud computing, με σκοπό να παρέχει την δυνατότητα για λειτουργία από απόσταση εφαρμογών γραφείου σε κινητές συσκευές και ταμπλέτες.

4.1 Περιγραφή περίπτωσης

Θα παρουσιάσω μία περίπτωση high availability clustering σε LAN. Δηλαδή ομαδοποίηση κάποιων server (2 και πάνω) με απώτερο σκοπό την υψηλή διαθεσιμότητα των υπηρεσιών που διαθέτουν. Για την υλοποίηση αυτή έκανα την εγκατάσταση δύο Ubuntu server 14.04.1 LTS στο εικονικό περιβάλλον του VMware. Στη συνέχεια επέλεξα να εγκαταστήσω σαν κύρια υπηρεσία το Moodle και τέλος δημιούργησα υψηλή διαθεσιμότητα χρησιμοποιώντας το Heartbeat. Δηλαδή αυτό που κατάφερα μέσω τεχνολογιών που βασίζονται στο virtualization, είναι ότι όταν φορτώνω μία συγκεκριμένη IP (που έχω ορίσει εγώ) έχω άμεση πρόσβαση στην υπηρεσία του Moodle από υπολογιστές αλλά και κινητές συσκευές. Όταν υπάρξει πρόβλημα και ο ένας server δεν είναι προσβάσιμος τότε ο δεύτερος server λειτουργεί ως backup και το Moodle φορτώνεται πάλι επιτυχώς (high availability).

4.2 Εγκατάσταση των Virtual Servers

Για την εγκατάσταση των 2 εικονικών servers χρησιμοποίησα το VMware Workstation έκδοσης 10.0.1 build-1379776



The screenshot shows the 'Product Information' window of VMware Workstation. It contains the following details:

- Product Information:**
 - Product: VMware® Workstation
 - Version: 10.0.1 build-1379776
- License Information:**
 - Status: Licensed
 - Type: Volume
 - Expiration: No expiration
- Additional Information:**
 - Host name: SpeedDemon-PC
 - Memory: 16344 MB
 - Host OS version: Windows 7 Professional, 64-bit 6.1.7601, Service Pack 1
 - UI log file: C:\Users\SpeedDemon\AppData\Local\Temp\vmware-SpeedDemon\vmware-ui-5912.log

Copyright © 1998-2013 VMware, Inc. All rights reserved.
This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>.
VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions.
All other marks and names mentioned herein may be trademarks of their respective companies.

Στη συνέχεια πηγαίνοντας στο μενού File-> New virtual machine έκανα και τους 2 server εγκατάσταση, δίνοντας τους τα default χαρακτηριστικά που πρότεινε το VMware σε: CPU, RAM, Hard Disk εκτός από network adapter που αντί για την default επιλογή NAT, έβαλα BRIDGED. Έτσι ο εικονικός server συνδέεται σαν πραγματική μηχανή με δική του IP στο δίκτυο, αντίθετα με το NAT που παίρνει την ίδια IP με τον φυσικό υπολογιστή που τον φιλοξενεί μέσω του VMware.

Η έκδοση των Ubuntu servers είναι η 14.04.1 LTS.

4.3 Εγκατάσταση MOODLE

Το Moodle είναι μια πλατφόρμα εκμάθησης που έχει σχεδιαστεί για να παρέχει ένα ασφαλές και ολοκληρωμένο σύστημα σε εκπαιδευτικούς/διαχειριστές και μαθητευόμενους, για τη δημιουργία εξατομικευμένων περιβαλλόντων μάθησης. Στην ουσία είναι μια ιστοσελίδα που επέλεξα να χρησιμοποιήσω γιατί είναι καλοφτιαγμένη, ανοιχτού κώδικα και προσβάσιμη και από κινητές συσκευές όπως smartphones, tablets χωρίς προβλήματα συμβατότητας. Ακολουθούν τα βήματα εγκατάστασης.

4.3.1 Σύνδεση σαν Super User στους servers

Εκκίνηση του server και σύνδεση ως super user. Όταν συνδεθούμε σαν super user έχουμε την δυνατότητα για πρόσβαση οπουδήποτε θέλουμε στο σύστημα, για περαιτέρω επεξεργασία στην πορεία χωρίς περιορισμούς. Αυτό το καταφέρνουμε γράφοντας `sudo su` και τον απαραίτητο κωδικό.

```
Ubuntu 14.04.1 LTS ubuntu3 tty1
ubuntu3 login: mike
Password:
Last login: Sun Oct  5 10:12:55 PDT 2014 on tty1
Welcome to Ubuntu 14.04.1 LTS (GNU/Linux 3.13.0-32-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
mike@ubuntu3:~$ sudo su
sudo: unable to resolve host ubuntu3
[sudo] password for mike:
root@ubuntu3:/home/mike#
```

(-> Επαναλαμβάνουμε όλα τα παρακάτω βήματα και στον δεύτερο server κάνοντας τις απαραίτητες αλλαγές. <-)

4.3.2 Αλλαγή hostname στους servers

Στο συγκεκριμένο παράδειγμα έχω επιλέξει `ubuntu3` και `ubuntu4` αντίστοιχα.

Γράφοντας `vi /etc/hostname /etc/hosts` μπορούμε να επεξεργαστούμε και να αλλάξουμε το `hostname`. Γράφουμε `:wq!` για αποθήκευση και κλείσιμο του αρχείου που επεξεργαστήκαμε. Αυτό ισχύει και για οτιδήποτε επεξεργασία αρχείου κάνουμε με την εντολή `vi`.

```
root@ubuntu:/home/mike# vi /etc/hostname /etc/hosts
```

Σαν αποτέλεσμα μας δίνει το παρακάτω και εκεί θα κάνουμε την αλλαγή.

```
ubuntu3  
  
"/etc/hostname" 1 line, 7 characters
```

Στη συνέχεια κάνουμε επανεκκίνηση για να αποθηκευτούν οι αλλαγές με το service hostname restart.

```
root@ubuntu:/home/mike# service hostname restart
```

Και τέλος ελέγχουμε αν έγινε η αλλαγή σωστά γράφοντας hostname.

```
root@ubuntu:/home/mike# hostname  
ubuntu3
```

4.3.3 Static IP στους servers

Για να βάλουμε στατική IP στους server γράφουμε vi /etc/network/interfaces.

```
root@ubuntu:/home/mike# vi /etc/network/interfaces
```

Πρέπει να αλλάξουμε το «iface eth0 inet dhcp» σε «iface eth0 inet static» και μετά τα παρακάτω στοιχεία ανάλογα τις ανάγκες του κάθε δικτύου. (Στον 2^ο server έχω δώσει την 10.32.192.55)

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
    address 10.32.192.54
    netmask 255.255.255.128
    network 10.32.192.0
    broadcast 10.32.192.127
    gateway 10.32.192.1
    dns-nameservers 10.32.192.1
```

Στην συνέχεια κάνουμε μια επανεκκίνηση τον server για να μπουν σε εφαρμογή οι αλλαγές που κάναμε. Γράφουμε shutdown -r now.

```
root@ubuntu:/home/mike# shutdown -r now_
```

4.3.4 Εγκατάσταση των απαραίτητων στοιχείων Apache, MySQL, PHP

Σαν πρώτο βασικό βήμα θα κάνουμε update το app cache με την εντολή apt-get update.

```
root@ubuntu:/home/mike# apt-get update_
```

Το τελευταίο κομμάτι της ενημέρωσης πρέπει να έχει την εξής μορφή:

```
Get:43 http://us.archive.ubuntu.com trusty-updates/universe Sources [86.2 kB]
Get:44 http://us.archive.ubuntu.com trusty-updates/multiverse Sources [3,531 B]
Get:45 http://us.archive.ubuntu.com trusty-updates/main amd64 Packages [337 kB]
Get:46 http://us.archive.ubuntu.com trusty-updates/restricted amd64 Packages [5,820 B]
Get:47 http://us.archive.ubuntu.com trusty-updates/universe amd64 Packages [208 kB]
Get:48 http://us.archive.ubuntu.com trusty-updates/multiverse amd64 Packages [9,366 B]
Get:49 http://us.archive.ubuntu.com trusty-updates/main i386 Packages [331 kB]
Get:50 http://us.archive.ubuntu.com trusty-updates/restricted i386 Packages [5,820 B]
Get:51 http://us.archive.ubuntu.com trusty-updates/universe i386 Packages [209 kB]
Get:52 http://us.archive.ubuntu.com trusty-updates/multiverse i386 Packages [9,532 B]
Get:53 http://us.archive.ubuntu.com trusty-updates/main Translation-en [150 kB]
Get:54 http://us.archive.ubuntu.com trusty-updates/multiverse Translation-en [4,719 B]
Get:55 http://us.archive.ubuntu.com trusty-updates/restricted Translation-en [1,736 B]
Get:56 http://us.archive.ubuntu.com trusty-updates/universe Translation-en [105 kB]
Get:57 http://us.archive.ubuntu.com trusty-backports/main Sources [4,760 B]
Get:58 http://us.archive.ubuntu.com trusty-backports/restricted Sources [14 B]
Get:59 http://us.archive.ubuntu.com trusty-backports/universe Sources [14.1 kB]
Get:60 http://us.archive.ubuntu.com trusty-backports/multiverse Sources [1,315 B]
Get:61 http://us.archive.ubuntu.com trusty-backports/main amd64 Packages [6,356 B]
Get:62 http://us.archive.ubuntu.com trusty-backports/restricted amd64 Packages [14 B]
Get:63 http://us.archive.ubuntu.com trusty-backports/universe amd64 Packages [17.4 kB]
Get:64 http://us.archive.ubuntu.com trusty-backports/multiverse amd64 Packages [943 B]
Get:65 http://us.archive.ubuntu.com trusty-backports/main i386 Packages [6,379 B]
Get:66 http://us.archive.ubuntu.com trusty-backports/restricted i386 Packages [14 B]
Get:67 http://us.archive.ubuntu.com trusty-backports/universe i386 Packages [17.4 kB]
Get:68 http://us.archive.ubuntu.com trusty-backports/multiverse i386 Packages [945 B]
Get:69 http://us.archive.ubuntu.com trusty-backports/main Translation-en [4,216 B]
Get:70 http://us.archive.ubuntu.com trusty-backports/multiverse Translation-en [613 B]
Get:71 http://us.archive.ubuntu.com trusty-backports/restricted Translation-en [14 B]
Get:72 http://us.archive.ubuntu.com trusty-backports/universe Translation-en [14.6 kB]
Ign http://us.archive.ubuntu.com trusty/main Translation-en_US
Ign http://us.archive.ubuntu.com trusty/multiverse Translation-en_US
Ign http://us.archive.ubuntu.com trusty/restricted Translation-en_US
Ign http://us.archive.ubuntu.com trusty/universe Translation-en_US
Fetched 29.8 MB in 3min 14s (153 kB/s)
Reading package lists... Done
root@ubuntu3:/home/mike#
```

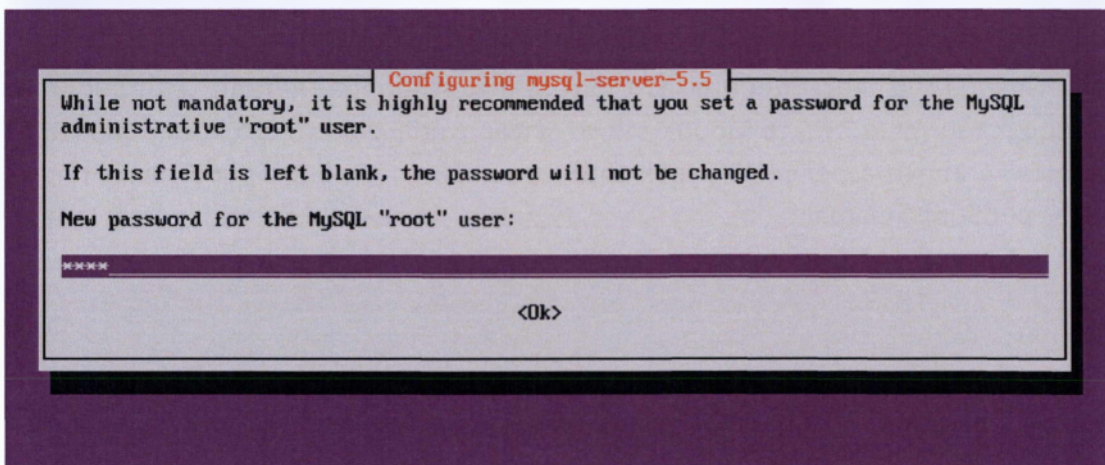
Όταν ολοκληρωθεί το παραπάνω, θα κάνουμε λήψη και εγκατάσταση των apache2, MySQL και php5 με μία εντολή: `sudo apt-get install apache2 mysql-client mysql-server php5`.

```
root@ubuntu3:/home/mike# apt-get install apache2 mysql-client mysql-server php5
```


Αφού ολοκληρωθεί η λήψη θα χρειαστεί να κάνουμε επιβεβαίωση:

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  apache2-bin apache2-data libaio1 libapache2-mod-php5 libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libdbd-mysql-perl libdbi-perl
  libhtml-template-perl libmysqlclient18 libterm-readkey-perl libwrap0
  mysql-client-5.5 mysql-client-core-5.5 mysql-common mysql-server-5.5
  mysql-server-core-5.5 php5-cli php5-common php5-json php5-readline ssl-cert
  tcpd
Suggested packages:
  www-browser apache2-doc apache2-suexec-pristine apache2-suexec-custom
  apache2-utils php-pear libclone-perl libmldbm-perl libnet-daemon-perl
  libplrpc-perl libsql-statement-perl libipc-sharedcache-perl tinyca mailx
  php5-user-cache openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data libaio1 libapache2-mod-php5 libapr1
  libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap libdbd-mysql-perl
  libdbi-perl libhtml-template-perl libmysqlclient18 libterm-readkey-perl
  libwrap0 mysql-client mysql-client-5.5 mysql-client-core-5.5 mysql-common
  mysql-server mysql-server-5.5 mysql-server-core-5.5 php5 php5-cli
  php5-common php5-json php5-readline ssl-cert tcpd
0 upgraded, 29 newly installed, 0 to remove and 45 not upgraded.
Need to get 15.2 MB of archives.
After this operation, 123 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Έτσι θα ξεκινήσει η εγκατάσταση και κατά την διάρκεια της θα ορίσουμε και κωδικό root user στο mysql.



Στην ολοκλήρωση πρέπει να φτάσουμε σε αυτή τη μορφή.

```
Creating config file /etc/php5/cli/php.ini with new version
php5_invoke pdo: already enabled for cli SAPI
php5_invoke opcache: already enabled for cli SAPI
Setting up php5-readline (5.5.9+dfsg-1ubuntu4.4) ...

Creating config file /etc/php5/mods-available/readline.ini with new version
php5_invoke: Enable module readline for cli SAPI
php5_invoke: Enable module readline for apache2 SAPI
Setting up php5-json (1.3.2-2build1) ...
php5_invoke: Enable module json for cli SAPI
php5_invoke: Enable module json for apache2 SAPI
Processing triggers for ufw (0.34~rc-0ubuntu2) ...
Setting up libapache2-mod-php5 (5.5.9+dfsg-1ubuntu4.4) ...

Creating config file /etc/php5/apache2/php.ini with new version
php5_invoke pdo: already enabled for apache2 SAPI
php5_invoke readline: already enabled for apache2 SAPI
php5_invoke json: already enabled for apache2 SAPI
php5_invoke opcache: already enabled for apache2 SAPI
Module mpm_event disabled.
Enabling module mpm_prefork.
apache2_switch_mpm Switch to prefork
 * Restarting web server apache2
AH00557: apache2: apr_sockaddr_info_get() failed for ubuntu3
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0
.0.1. Set the 'ServerName' directive globally to suppress this message
[ OK ]

apache2_invoke: Enable module php5
 * Restarting web server apache2
AH00557: apache2: apr_sockaddr_info_get() failed for ubuntu3
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0
.0.1. Set the 'ServerName' directive globally to suppress this message
[ OK ]

Setting up php5 (5.5.9+dfsg-1ubuntu4.4) ...
Processing triggers for libc-bin (2.19-0ubuntu6) ...
root@ubuntu3:/home/mike#
```

Τώρα πρέπει να κατεβάσουμε και να εγκαταστήσουμε κάποια επιπρόσθετα λογισμικά που χρειάζεται το Moodle για να λειτουργήσει σωστά. Γράφουμε `sudo apt-get install graphviz aspell php5-pspell php5-curl php5-gd php5-intl php5-mysql php5-xmlrpc php5-ldap clamav`.

```
root@ubuntu3:/home/mike# apt-get install graphviz aspell php5-pspell php5-curl php5-gd php5-intl php5-mysql php5-xmlrpc php5-ldap clamav
```

πού ολοκληρωθεί η λήψη θα πρέπει να γίνει επιβεβαίωση τ

```
Setting up php5 (5.5.9+dfsg-1ubuntu4.4) ...
Processing triggers for libc-bin (2.19-0ubuntu6) ...
root@ubuntu3:/home/nike# apt-get install graphviz aspell php5-pspell php5-curl php5-gd php5-intl php
5-mysql php5-xmllrpc php5-ldap clamav
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
aspell-en clamav-base clamav-freshclam dictionaries-common fontconfig
fontconfig-config fonts-dejavu-core fonts-liberation libaspell15 libcairo2
libcdt5 libcgraph6 libclamav6 libcurl3 libdatrie1 libfontconfig1 libgd3
libgraphite2-3 libgvc6 libgvr2 libharfbuzz0b libice6 libicu52 libjbig0
libjpeg-turbo8 libjpeg8 libltdl7 libpango-1.0-0 libpangocairo-1.0-0
libpangofc2-1.0-0 libpathplan4 libpixmap-1-0 libsm6 libthai-data libthai0
libtiff5 libvpx1 libxaw7 libxcb-render0 libxcb-shm0 libxmu6 libxpm4
libxrender1 libxt6 x11-common
Suggested packages:
aspell-doc spellutils clamav-docs emacs-common jed-extra gsfonts
graphviz-doc libclamunrar6 libgd-tools ttf-baekmuk ttf-arphic-gbsn00lp
ttf-arphic-bsmi00lp ttf-arphic-gkai00mp ttf-arphic-bkai00mp
The following NEW packages will be installed:
aspell aspell-en clamav clamav-base clamav-freshclam dictionaries-common
fontconfig fontconfig-config fonts-dejavu-core fonts-liberation graphviz
libaspell15 libcairo2 libcdt5 libcgraph6 libclamav6 libcurl3 libdatrie1
libfontconfig1 libgd3 libgraphite2-3 libgvc6 libgvr2 libharfbuzz0b libice6
libicu52 libjbig0 libjpeg-turbo8 libjpeg8 libltdl7 libpango-1.0-0
libpangocairo-1.0-0 libpangofc2-1.0-0 libpathplan4 libpixmap-1-0 libsm6
libthai-data libthai0 libtiff5 libvpx1 libxaw7 libxcb-render0 libxcb-shm0
libxmu6 libxpm4 libxrender1 libxt6 php5-curl php5-gd php5-intl php5-ldap
php5-mysql php5-pspell php5-xmllrpc x11-common
0 upgraded, 55 newly installed, 0 to remove and 45 not upgraded.
Need to get 18.5 MB of archives.
After this operation, 68.5 MB of additional disk space will be used.
Do you want to continue? [Y/n] _
```

παρακάτω μορφή μας δείχνει πως ολοκληρώνεται η εγκατάσταση των επιπλέον απρόσθετων στοιχείων.

```
Setting up libpathplan4 (2.36.0-0ubuntu3) ...
Setting up libgvc6 (2.36.0-0ubuntu3) ...
Setting up libgvr2 (2.36.0-0ubuntu3) ...
Processing triggers for ureadahead (0.100.0-16) ...
Setting up clamav (0.98.1+dfsg-4ubuntu1.1) ...
Setting up libice6:amd64 (2:1.0.8-2) ...
Setting up libsm6:amd64 (2:1.2.1-2) ...
Setting up libxt6:amd64 (1:1.1.4-1) ...
Setting up libxmu6:amd64 (2:1.1.1-1) ...
Setting up libxaw7:amd64 (2:1.0.12-1) ...
Setting up graphviz (2.36.0-0ubuntu3) ...
Processing triggers for dictionaries-common (1.20.5) ...
aspell-autobuildhash: processing: en [en-common]
aspell-autobuildhash: processing: en [en-variant_0]
aspell-autobuildhash: processing: en [en-variant_1]
aspell-autobuildhash: processing: en [en-variant_2]
aspell-autobuildhash: processing: en [en-u_accents-only]
aspell-autobuildhash: processing: en [en-wo_accents-only]
aspell-autobuildhash: processing: en [en_CA-variant_0]
aspell-autobuildhash: processing: en [en_CA-variant_1]
aspell-autobuildhash: processing: en [en_CA-u_accents-only]
aspell-autobuildhash: processing: en [en_CA-wo_accents-only]
aspell-autobuildhash: processing: en [en_GB-ise-u_accents-only]
aspell-autobuildhash: processing: en [en_GB-ise-wo_accents-only]
aspell-autobuildhash: processing: en [en_GB-ize-u_accents-only]
aspell-autobuildhash: processing: en [en_GB-ize-wo_accents-only]
aspell-autobuildhash: processing: en [en_GB-variant_0]
aspell-autobuildhash: processing: en [en_GB-variant_1]
aspell-autobuildhash: processing: en [en_US-u_accents-only]
aspell-autobuildhash: processing: en [en_US-wo_accents-only]
Setting up aspell (0.60.7~20110707-1ubuntu1) ...
Processing triggers for dictionaries-common (1.20.5) ...
Setting up aspell-en (7.1-0-1) ...
Processing triggers for libc-bin (2.19-0ubuntu6) ...
Processing triggers for libapache2-mod-php5 (5.5.9+dfsg-1ubuntu4.4) ...
Processing triggers for dictionaries-common (1.20.5) ...
root@ubuntu3:/home/nike# _
```

Πρέπει να γίνει επανεκκίνηση του apache ώστε όλα να δουλεύουν σωστά: service apache2 restart

```
root@ubuntu3:/home/mike# service apache2 restart
```

Σε αυτό το σημείο θα εγκαταστήσουμε και το GIT που με αυτό θα γίνεται εγκατάσταση ή αναβάθμιση το Moodle. Γράφουμε : apt-get install git-core

```
root@ubuntu3:/home/mike# apt-get install git-core_
```

Επιβεβαιώνουμε την εγκατάσταση μετά την λήψη:

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  git git-man liberror-perl patch
Suggested packages:
  git-daemon-run git-daemon-sysvinit git-doc git-el git-email git-gui gitk
  gitweb git-arch git-bzr git-cvs git-mediawiki git-svn diffutils-doc
The following NEW packages will be installed:
  git git-core git-man liberror-perl patch
0 upgraded, 5 newly installed, 0 to remove and 45 not upgraded.
Need to get 3,360 kB of archives.
After this operation, 21.8 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Και το τελευταίο κομμάτι της εγκατάστασης του GIT πρέπει να είναι έτσι:

```
git-daemon-run git-daemon-sysvinit git-doc git-el git-email git-gui gitk
gitweb git-arch git-bzr git-cvs git-mediawiki git-svn diffutils-doc
The following NEW packages will be installed:
  git git-core git-man liberror-perl patch
0 upgraded, 5 newly installed, 0 to remove and 45 not upgraded.
Need to get 3,360 kB of archives.
After this operation, 21.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us.archive.ubuntu.com/ubuntu/ trusty/main liberror-perl all 0.17-1.1 [21.1 kB]
Get:2 http://us.archive.ubuntu.com/ubuntu/ trusty/main git-man all 1:1.9.1-1 [698 kB]
Get:3 http://us.archive.ubuntu.com/ubuntu/ trusty/main git amd64 1:1.9.1-1 [2,555 kB]
Get:4 http://us.archive.ubuntu.com/ubuntu/ trusty/main git-core all 1:1.9.1-1 [1,452 B]
Get:5 http://us.archive.ubuntu.com/ubuntu/ trusty-updates/main patch amd64 2.7.1-4ubuntu1 [84.4 kB]
Fetched 3,360 kB in 1min 16s (44.1 kB/s)
Selecting previously unselected package liberror-perl.
(Reading database ... 50831 files and directories currently installed.)
Preparing to unpack .../liberror-perl_0.17-1.1_all.deb ...
Unpacking liberror-perl (0.17-1.1) ...
Selecting previously unselected package git-man.
Preparing to unpack .../git-man_1:1.9.1-1_all.deb ...
Unpacking git-man (1:1.9.1-1) ...
Selecting previously unselected package git.
Preparing to unpack .../git_1:1.9.1-1_amd64.deb ...
Unpacking git (1:1.9.1-1) ...
Selecting previously unselected package git-core.
Preparing to unpack .../git-core_1:1.9.1-1_all.deb ...
Unpacking git-core (1:1.9.1-1) ...
Selecting previously unselected package patch.
Preparing to unpack .../patch_2.7.1-4ubuntu1_amd64.deb ...
Unpacking patch (2.7.1-4ubuntu1) ...
Processing triggers for man-db (2.6.7.1-1) ...
Setting up liberror-perl (0.17-1.1) ...
Setting up git-man (1:1.9.1-1) ...
Setting up git (1:1.9.1-1) ...
Setting up git-core (1:1.9.1-1) ...
Setting up patch (2.7.1-4ubuntu1) ...
root@ubuntu3:/home/mike#
```

4.3.5 Λήψη του Moodle

Πηγαίνουμε στον φάκελο opt γράφοντας : cd /opt

```
root@ubuntu3:/home/mike# cd /opt
root@ubuntu3:/opt# _
```

Κάνουμε λήψη του πυρήνα της εφαρμογής του Moodle : git clone git://git.moodle.org/moodle.git

```
root@ubuntu3:/opt# git clone git://git.moodle.org/moodle.git _
```

Το τελευταίο κομμάτι της εγκατάστασης πρέπει να είναι αυτής της μορφής:

```
Cloning into 'git.moodle'...
fatal: No path specified. See 'man git-pull' for valid url syntax
root@ubuntu3:/opt# git clone git://git.moodle.org/moodle.git
Cloning into 'moodle'...
remote: Counting objects: 780454, done.
remote: Compressing objects: 100% (190232/190232), done.
remote: Total 780454 (delta 578948), reused 722591 (delta 571141)
Receiving objects: 100% (780454/780454), 277.60 MiB | 515.00 KiB/s, done.
Resolving deltas: 100% (578948/578948), done.
Checking connectivity... done.
Checking out files: 100% (13683/13683), done.
root@ubuntu3:/opt#
```

Στην συνέχεια μπαίνουμε στον φάκελο του Moodle: cd moodle

```
root@ubuntu3:/opt# cd moodle
root@ubuntu3:/opt/moodle# _
```

Κάνουμε ανάκτηση των διαθέσιμων εκδόσεων: git branch -a

```
root@ubuntu3:/opt/moodle# git branch -a
* master
remotes/origin/HEAD -> origin/master
remotes/origin/MOODLE_13_STABLE
remotes/origin/MOODLE_14_STABLE
remotes/origin/MOODLE_15_STABLE
remotes/origin/MOODLE_16_STABLE
remotes/origin/MOODLE_17_STABLE
remotes/origin/MOODLE_18_STABLE
remotes/origin/MOODLE_19_STABLE
remotes/origin/MOODLE_20_STABLE
remotes/origin/MOODLE_21_STABLE
remotes/origin/MOODLE_22_STABLE
remotes/origin/MOODLE_23_STABLE
remotes/origin/MOODLE_24_STABLE
remotes/origin/MOODLE_25_STABLE
remotes/origin/MOODLE_26_STABLE
remotes/origin/MOODLE_27_STABLE
remotes/origin/master
root@ubuntu3:/opt/moodle#
```

Συνεχίζουμε δείχνοντας στο GIT ποια έκδοση να χρησιμοποιήσει αλλά και να ελέγχει για πιθανές αλλαγές, όπως αναβαθμίσεις : `git branch --track MOODLE_27_STABLE origin/MOODLE_27_STABLE`

```
root@ubuntu3:/opt/moodle# git branch --track MOODLE_27_STABLE origin/MOODLE_27_STABLE
Branch MOODLE_27_STABLE set up to track remote branch MOODLE_27_STABLE from origin.
```

Τέλος ελέγχουμε ποια έκδοση επιλέξαμε: `git checkout MOODLE_27_STABLE`

```
root@ubuntu3:/opt/moodle# git checkout MOODLE_27_STABLE
Switched to branch 'MOODLE_27_STABLE'
Your branch is up-to-date with 'origin/MOODLE_27_STABLE'.
root@ubuntu3:/opt/moodle#
```

4.3.6 Αντιγραφή του /opt/moodle στο /var/www/html

Πάμε μία θέση πίσω και αντιγράφουμε τον φάκελο /opt/moodle στο /var/www/html: `cp -R /opt/moodle /var/www/html/`; `cd -` και μετά `cp -R /opt/moodle /var/www/html/`

```
root@ubuntu3:/opt/moodle# cd -
/opt
root@ubuntu3:/opt# cp -R /opt/moodle /var/www/html/
root@ubuntu3:/opt#
```

Στη συνέχεια φτιάχνουμε τον φάκελο που το moodle θα αποθηκεύει τα δεδομένα του : `mkdir /var/moodldata`

```
root@ubuntu3:/opt# mkdir /var/moodldata
```

Μετά αλλάζουμε το ποιος θα έχει την κυριότητα αυτού του φακέλου: `chown -R www-data /var/moodldata`

```
root@ubuntu3:/opt# chown -R www-data /var/moodldata
```

Επίσης αλλάζουμε και τα δικαιώματα που θα έχει ο ιδιοκτήτης του φακέλου: `chmod -R 777 /var/moodldata`

```
root@ubuntu3:/opt# chmod -R 777 /var/moodldata
```

Τέλος αλλάζουμε τα δικαιώματα στον κεντρικό φάκελο του moodle: `chmod -R 0755 /var/www/html/moodle`

```
root@ubuntu3:/opt# chmod -R 0755 /var/www/html/moodle
```

4.3.7 Ρύθμιση του MySQL server

Τώρα θα ορίσουμε ποια θα είναι η προεπιλεγμένη μηχανή αποθήκευσης για το mysql. Αφού έχουμε πρόσβαση στο αρχείο ρυθμίσεων του mysql: vi /etc/mysql/my.cnf, θα προσθέσουμε στον τομέα του mysqld κάτω από τα basic settings το εξής: default-storage-engine = innodb

```
root@ubuntu3:/opt# vi /etc/mysql/my.cnf
```

```
[mysqld]
#
# * Basic Settings
#
user                = mysql
pid-file            = /var/run/mysqld/mysqld.pid
socket              = /var/run/mysqld/mysqld.sock
port                = 3306
basedir             = /usr
datadir             = /var/lib/mysql
tmpdir              = /tmp
lc-messages-dir    = /usr/share/mysql
skip-external-locking
default-storage-engine = innodb
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address        = 127.0.0.1
#
# * Fine Tuning
#
key_buffer           = 16M
max_allowed_packet  = 16M
thread_stack         = 192K
thread_cache_size   = 8
# This replaces the startup script and checks MyISAM tables if needed
# the first time they are touched
myisam-recover      = BACKUP
#max_connections    = 100
#table_cache        = 64
#thread_concurrency = 10
#
# * Query Cache Configuration
#
query_cache_limit   = 1M
query_cache_size    = 16M
:uq!_
```

Στην συνέχεια θα κάνουμε επανεκκίνηση την υπηρεσία του mysql: service mysql restart

```
root@ubuntu3:/opt# service mysql restart
mysql stop/waiting
mysql start/running, process 1907
root@ubuntu3:/opt#
```

Τέλος, θα δημιουργήσουμε την βάση δεδομένων του moodle αλλά και τον χρήστη του mysql του moodle και θα θέσουμε τα κατάλληλα δικαιώματα. Γράφουμε «mysql -u root -p» και συνεχίζουμε όπως παρακάτω. (Όπου mike = το όνομα που έβαλε ο κάθε χρήστης, όπου 1234 = ο κωδικός που όρισε ο χρήστης στην ενότητα 4.3.4)

```
root@ubuntu3:/opt# mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 36
Server version: 5.5.38-0ubuntu0.14.04.1 (Ubuntu)

Copyright (c) 2000, 2014, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE moodle DEFAULT CHARACTER SET utf8 COLLATE utf8_unicode_ci;
Query OK, 1 row affected (0.03 sec)

mysql> create user 'mike'@'localhost' IDENTIFIED BY '1234';
Query OK, 0 rows affected (0.01 sec)

mysql> GRANT SELECT,INSERT,UPDATE,DELETE,CREATE,CREATE TEMPORARY TABLES,DROP,INDEX,ALTER ON moodle.*
  TO mike@localhost IDENTIFIED BY '1234';
Query OK, 0 rows affected (0.01 sec)

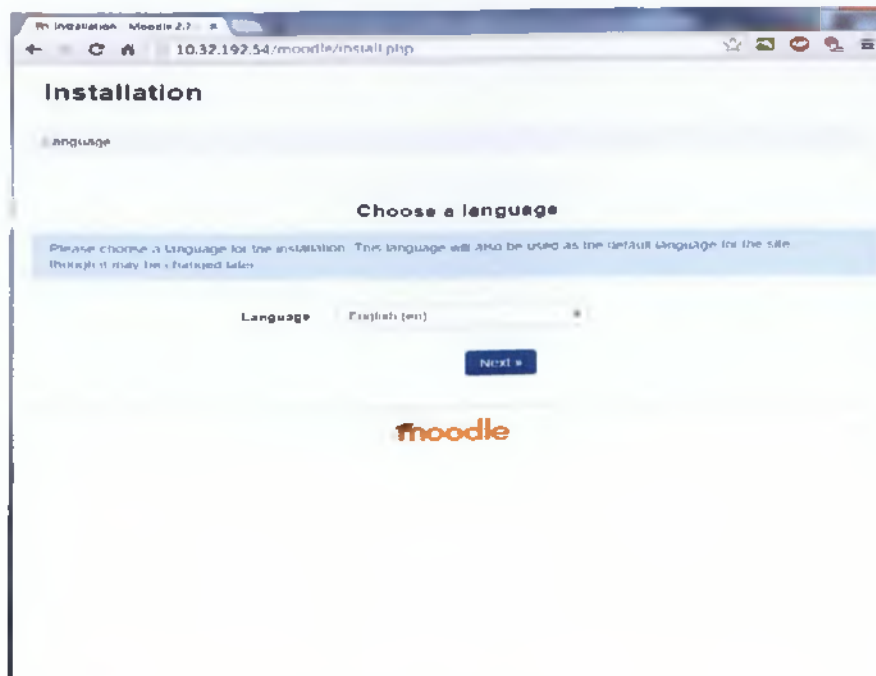
mysql> quit;
Bye
```

4.3.8 Ολοκλήρωση ρυθμίσεων του Moodle

Πρώτο βήμα. Δίνουμε τέτοια δικαιώματα στο directory του moodle ώστε να μπορούμε να γράψουμε σε αυτό με ευκολία : `chmod -R 777 /var/www/html/moodle`

```
root@ubuntu3:/opt# cd
root@ubuntu3:~# chmod -R 777 /var/www/html/moodle
```

Τώρα θα φορτώσουμε την IP του server μας / moodle. Στην συγκεκριμένη περίπτωση θα φορτώσω σε έναν browser <http://10.32.192.54/moodle> , στον δεύτερο server θα φορτώσω την 10.32.192.55. Το setup ρυθμίσεων που θα εμφανιστεί θα είναι όπως παρακάτω:



Στο επόμενο βήμα αλλάζουμε το data directory σε /var/moodledata

Confirm paths

Web address

The full address where Moodle will be accessed i.e. the address that users will enter into the address bar of their browser to access Moodle

It is not possible to access Moodle using multiple addresses. If your site is accessible via multiple addresses then choose the easiest one and set up a permanent redirect for each of the other addresses

If your site is accessible both from the Internet, and from an internal network (sometimes called an Intranet), then use the public address here

If the current address is not correct, please change the URL in your browser's address bar and restart the installation

Moodle directory

The full path to the directory containing the Moodle code

Data directory

A directory where Moodle will store all file content uploaded by users.

This directory should be both readable and writeable by the web server user (usually 'www-data' 'nobody', or 'apache')

It must not be directly accessible over the web

If the directory does not currently exist, the installation process will attempt to create it

Web address

Moodle directory

Data directory

◀ Previous **Next >**

Στην συνέχεια κάνουμε τις παρακάτω ρυθμίσεις κάνοντας τις απαραίτητες αλλαγές ανάλογα τις ανάγκες του κάθε χρήστη:

Choose database driver

Moodle supports several types of database servers. Please contact server administrator if you do not know which type to use

Type

[« Previous](#) [Next »](#)



Database settings

Improved MySQL (native/mysqli)

The database is where most of the Moodle settings and data are stored and must be configured here

The database name, username, and password are required fields, table prefix is optional

If the database currently does not exist, and the user you specify has permission, Moodle will attempt to create a new database with the correct permissions and settings

Database host	<input type="text" value="localhost"/>
Database name	<input type="text" value="moodle"/>
Database user	<input type="text" value="mike"/>
Database password	<input type="text" value="1234"/>
Tables prefix	<input type="text" value="mdl_"/>
Database port	<input type="text"/>
Unix socket	<input type="text"/>

[« Previous](#) [Next »](#)



Server checks

Name	Information	Report	Plugin	Status
unicode		① must be installed and enabled		OK
database	mysql (5.5.38-0ubuntu0.14.04.1)	① version 5.5.31 is required and you are running 5.5.38.0.0.14.04.1		OK
php		① version 5.4.4 is required and you are running 5.5.9.1.4.4		OK
pcrcnucode		① should be installed and enabled for best results		OK
php_extension	iconv	① must be installed and enabled		OK
php_extension	mbstring	① should be installed and enabled for best results		OK
php_extension	curl	① must be installed and enabled		OK
php_extension	openssl	① should be installed and enabled for best results		OK
php_extension	tokenizer	① should be installed and enabled for best results		OK
php_extension	intl	① should be installed and enabled for best results		OK
php_extension	soap	① should be installed and enabled for best results		OK
php_extension	ctype	① must be installed and enabled		OK
php_extension	zip	① must be installed and enabled		OK
php_extension	zlib	① must be installed and enabled		OK
php_extension	gd	① must be installed and enabled		OK
php_extension	simplexml	① must be installed and enabled		OK
php_extension	spl	① must be installed and enabled		OK
php_extension	pcr	① must be installed and enabled		OK
php_extension	dom	① must be installed and enabled		OK
php_extension	xml	① must be installed and enabled		OK
php_extension	enl	① should be installed and enabled for best results		OK
php_extension	json	① must be installed and enabled		OK
php_extension	hash	① must be installed and enabled		OK
php_setting	memory_limit	① recommended setting detected		OK
php_setting	file_uploads	① recommended setting detected		OK
php_setting	opcache.enable	① recommended setting detected		OK

Your server environment meets all minimum requirements

Continue

Αφού συνεχίσουμε στα επόμενα βήματα και συμπληρώσουμε ότι χρειαστεί στην δημιουργία κωδικού κλπ το setup θα ολοκληρωθεί. Στην συνέχεια θα φορτώσουμε πάλι την 10.32.192.54/moodle και πηγαίνοντας στο Site Administration > Server > System Paths, πρέπει να ορίσουμε τις διευθύνσεις φακέλων του moodle

Changes saved

System paths

Path to du ✓ Default: Empty
pathidu

Path to du. Probably something like /usr/bin/du. If you enter this, pages that display directory contents will run much faster for directories with a lot of files

Path to aspell ✓ Default: Empty
aspellpath

To use spell-checking within the editor, you MUST have **aspell 0.60** or later installed on your server, and you must specify the correct path to access the aspell binary. On Unix/Linux systems, this path is usually /usr/bin/aspell, but it might be something else.

Path to dot ✓ Default: Empty
pathdot

Path to dot. Probably something like /usr/bin/dot. To be able to generate graphics from DOT files, you must have installed the dot executable and point to it here. Note that, for now, this only used by the profiling features (Development->Profiling) built into Moodle.

Save changes

Στο τελευταίο βήμα θα αναιρέσουμε το πρώτο βήμα, δηλαδή θα επαναφέρουμε τα δικαιώματα στο webroot: `chmod -R 0755 /var/www/html/moodle`

```
root@ubuntu3:~# chmod -R 0755 /var/www/html/moodle
```

Η εγκατάσταση του moodle στην ουσία έχει τελειώσει. Αλλά θα κάνουμε κάποιες προτεινόμενες αλλαγές.

```
root@ubuntu3:/etc/php5# vi /etc/php5/apache2/conf.d/05-opcache.ini
```

Μέσα σε αυτό το αρχείο γράφουμε : `opcache.enable = 1`

```
; configuration for php ZendOpcache module
; priority=05
zend_extension=opcache.so
opcache.enable = 1
```

Κάνουμε και επανεκκίνηση την υπηρεσία του apache: `service apache2 restart`

```
root@ubuntu3:/etc/php5# service apache2 restart
* Restarting web server apache2
AH00557: apache2: apr_sockaddr_info_get() failed for ubuntu3
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0
.0.1. Set the 'ServerName' directive globally to suppress this message
```

[OK]

Ανοίγουμε το αρχείο στο οποίο υπάρχουν ρυθμίσεις για τα sites που σηκώνει ο apache: vi /etc/apache2/sites-available/000-default.conf

```
root@ubuntu3:~# vi /etc/apache2/sites-available/000-default.conf
```

Εδώ θα αλλάξουμε τη διαδρομή των αρχείων. Όπου DocumentRoot /var/www/html βάζουμε DocumentRoot /var/www/html/moodle

```
<VirtualHost *:80>
# The ServerName directive sets the request scheme, hostname and port that
# the server uses to identify itself. This is used when creating
# redirection URLs. In the context of virtual hosts, the ServerName
# specifies what hostname must appear in the request's Host: header to
# match this virtual host. For the default virtual host (this file) this
# value is not decisive as it is used as a last resort host regardless.
# However, you must set it for any further virtual host explicitly.
#ServerName www.example.com

ServerAdmin webmaster@localhost
DocumentRoot /var/www/html/moodle

# Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
# error, crit, alert, emerg.
# It is also possible to configure the loglevel for particular
# modules, e.g.
#LogLevel info ssl:warn

ErrorLog ${APACHE_LOG_DIR}/error.log
CustomLog ${APACHE_LOG_DIR}/access.log combined

# For most configuration files from conf-available/, which are
# enabled or disabled at a global level, it is possible to
# include a line for only one particular virtual host. For example the
# following line enables the CGI configuration for this host only
# after it has been globally disabled with "a2disconf".
#Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

"/etc/apache2/sites-available/000-default.conf" 31 lines, 1332 characters
```

Επανεκκίνηση το apache

```
root@ubuntu3:~# service apache2 restart
 * Restarting web server apache2
AH00557: apache2: apr_sockaddr_info_get() failed for ubuntu3
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0
.0.1. Set the 'ServerName' directive globally to suppress this message

[ OK ]
```

Τελευταίο βήμα σε όλη αυτή τη διαδικασία

```
root@ubuntu3:~# vi /var/www/html/moodle/config.php_
```

Μέσα σε αυτό το αρχείο αλλάζουμε τη διεύθυνση που με αυτήν θα έχουμε πρόσβαση στο moodle και αφήνουμε την IP του server μας ΜΟΝΟ. Όπως βλέπουμε παρακάτω:

```
<?php // Moodle configuration file

unset($CFG);
global $CFG;
$CFG = new stdClass();

$CFG->dbtype      = 'mysqli';
$CFG->dblibrary   = 'native';
$CFG->dbhost      = 'localhost';
$CFG->dbname      = 'moodle';
$CFG->dbuser      = 'mike';
$CFG->dbpass      = '1234';
$CFG->prefix      = 'mdl_';
$CFG->dboptions   = array (
    'dbpersist' => 0,
    'dbport'    => '',
    'dbsocket'  => '',
);

$CFG->wwwroot     = 'http://10.32.192.54';
$CFG->dataroot    = '/var/moodldata';
$CFG->admin       = 'admin';

$CFG->directorypermissions = 0777;

require_once(dirname(__FILE__) . '/lib/setup.php');

// There is no php closing tag in this file,
// it is intentional because it prevents trailing whitespace problems!

:ωq!_
```

Επίσης όπως μετά από τις περισσότερες αλλαγές κάνουμε επανεκκίνηση στο apache ώστε οι αλλαγές να γίνουν ενεργές.

```
root@ubuntu3:~# service apache2 restart
* Restarting web server apache2
AH00557: apache2: apr_sockaddr_info_get() failed for ubuntu3
AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0
.0.1. Set the 'ServerName' directive globally to suppress this message
[ OK ]
```

4.4 Servers υψηλής διαθεσιμότητας με το Heartbeat

Σε αυτή την ενότητα θα παρουσιάσω πώς να εγκαταστήσουμε και να ρυθμίσουμε το Heartbeat σε δύο Ubuntu 14.04.1 LTS servers. Δηλαδή θα χρησιμοποιήσω τους 2 server που έφτιαξα νωρίτερα (με υπηρεσία το moodle) ώστε να δημιουργήσω το λεγόμενο high availability clustering. Αυτό θα πραγματοποιηθεί χρησιμοποιώντας μία cluster-IP που θα μας δρομολογεί είτε στον κύριο server είτε στον δευτερεύον, όταν ο κύριος έχει 'πέσει' λόγω βλάβης. Έτσι ο χρήστης θα έχει πρόσβαση και πάλι στην υπηρεσία που θέλει. Ακολουθούν τα βασικά βήματα.

4.4.1 Επίλυση των hostnames μεταξύ των 2 servers

Σε αυτό το στάδιο θα βεβαιωθούμε ότι και οι δύο servers μπορούν να επιλύσουν τα hostnames. Για να το επιτύχουμε αυτό θα επεξεργαστούμε το αρχείο των hosts στον κάθε server χωριστά. Θα ορίσουμε μέσα σε αυτό το αρχείο τα εξής:

1) το localhost

2) τα δύο nodes (δηλαδή τους δύο server με τα hostname που τους δώσαμε στην ενότητα 4.3.2)

3) την cluster IP

Γράφουμε: `vi /etc/hosts` και κάνουμε την επεξεργασία όπως παρακάτω.

```
root@ubuntu1:~/home/nike# vi /etc/hosts
```

```
127.0.0.1    localhost
10.32.192.54 ubuntu3
10.32.192.55 ubuntu4
10.32.192.56 cluster

# The following lines are desirable for IPv6 capable hosts
::1        localhost ip6-localhost ip6-loopback
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters
```

Η cluster ip που ορίζουμε είναι υπεύθυνη να μας δρομολογεί προς τον κύριο server και όταν εκείνος δεν είναι ενεργός να μας πηγαίνει στον δεύτερο server.

4.4.2 Εγκατάσταση του Heartbeat

Κάνουμε εγκατάσταση του heartbeat και στους δύο servers με την εντολή: apt-get install heartbeat

```
root@ubuntu3:/home/mike# apt-get install heartbeat
```

Επιβεβαιώνουμε την εγκατάσταση μετά την λήψη


```

Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  cluster-glue crmsh gawk iproute libcfgy6 libcid3 libcmpp4 libcorosync-common4
  libcpq4 libcrmcluster4 libcrncommon3 libcrnservice1 libesntp6 libheartbeat2
  liblrn2 liblrnd1 libnet1 libopenhpi2 libopenipmi0 libpe-rules2 libpe-status4
  libpengine4 libperl5.18 libpils2 libplumb2 libplumbyp12 libqb0 libqorum5
  libsensors4 libsigsegv2 libsnmp-base libsnmp30 libstonith1 libstonith2
  libtransitioner2 libxml2-utils libxslt1.1 openhpid pacemaker
  pacemaker-cli-utils python-lxml resource-agents
Suggested packages:
  gawk-doc lm-sensors snmp-mibs-downloader python-lxml-dbg
The following NEW packages will be installed:
  cluster-glue crmsh gawk heartbeat iproute libcfgy6 libcid3 libcmpp4
  libcorosync-common4 libcpq4 libcrmcluster4 libcrncommon3 libcrnservice1
  libesntp6 libheartbeat2 liblrn2 liblrnd1 libnet1 libopenhpi2 libopenipmi0
  libpe-rules2 libpe-status4 libpengine4 libperl5.18 libpils2 libplumb2
  libplumbyp12 libqb0 libqorum5 libsensors4 libsigsegv2 libsnmp-base
  libsnmp30 libstonith1 libstonith2 libtransitioner2 libxml2-utils libxslt1.1
  openhpid pacemaker pacemaker-cli-utils python-lxml resource-agents
0 upgraded, 43 newly installed, 0 to remove and 45 not upgraded.
Need to get 5,878 kB of archives.
After this operation, 24.8 MB of additional disk space will be used.
Do you want to continue? [Y/n]

```

Το τελευταίο κομμάτι της εγκατάστασης πρέπει να είναι αυτής της μορφής:

```

Setting up libheartbeat2 (1:3.0.5-3.2) ...
Setting up iproute (1:3.12.0-2) ...
Setting up libxml2-utils (2.9.1+dfsg1-3ubuntu4.3) ...
Setting up libcorosync-common4 (2.3.3-1ubuntu1) ...
Setting up libqb0 (0.16.0-real-1ubuntu3) ...
Setting up libcfgy6 (2.3.3-1ubuntu1) ...
Setting up libcrncommon3 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up libpe-rules2 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up libcid3 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up libcmpp4 (2.3.3-1ubuntu1) ...
Setting up libcpq4 (2.3.3-1ubuntu1) ...
Setting up libqorum5 (2.3.3-1ubuntu1) ...
Setting up libstonith2 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up libcrmcluster4 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up libcrnservice1 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up libesntp6 (1.0.6-1ubuntu2) ...
Setting up liblrnd1 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up libpe-status4 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up libpengine4 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up libtransitioner2 (1.1.10+git20130802-1ubuntu2.1) ...
Setting up openhpid (2.14.1-1-1ubuntu2)
Starting openhpid: * success
Setting up python-lxml (3.3.3-1ubuntu0.1) ...
Setting up crmsh (1.2.5+log1034-1ubuntu4) ...
Setting up pacemaker-cli-utils (1.1.10+git20130802-1ubuntu2.1) ...
Processing triggers for ureadahead (0.100.0-16) ...
Setting up resource-agents (1:3.9.3+git20121009-3ubuntu2) ...
Setting up heartbeat (1:3.0.5-3.2) ...
Heartbeat not configured: /etc/ha.d/ha.cf not found. Heartbeat failure (cc=1). Failed.
Processing triggers for ureadahead (0.100.0-16) ...
Setting up pacemaker (1.1.10+git20130802-1ubuntu2.1) ...
addgroup: The group 'haclient' already exists as a system group. Exiting.
Warning: The home dir /var/lib/heartbeat you specified already exists.
The system user 'hacluster' already exists. Exiting.
Processing triggers for libc-bin (2.19-0ubuntu6) ...
Processing triggers for ureadahead (0.100.0-16) ...
root@ubuntu3:~/home/wike#

```


Στη συνέχεια θα επεξεργαστούμε το `/etc/ha.d/ha.cf` και στους δύο servers με τη μόνη διαφορά ότι στο σημείο που γράφει `ping` γράφουμε στον έναν server την IP του άλλου. Σε αυτό το παράδειγμα γράφω την IP του `ubuntu4` καθώς βρίσκομαι στο `ubuntu3`. : `vi /etc/ha.d/ha.cf`

```
root@ubuntu3:/home/mike# vi /etc/ha.d/ha.cf
```

```
logfacility local0
keepalive 2
deadtime 5
ping 10.32.192.55
udpport 694
bcast eth0
node ubuntu3
node ubuntu4
auto_failback on
```

Τέλος, δείχνουμε στο heartbeat ποιος είναι ο κύριος server, ποια είναι η cluster IP και ποια υπηρεσία θέλουμε να γίνεται cluster (επεξεργαζόμαστε αυτό το αρχείο και στους δύο servers). : vi /etc/ha.d/haresources

```
root@ubuntu3:/home/mike# vi /etc/ha.d/haresources_
```

```
ubuntu3 IPaddr::10.32.192.56/25/eth0 apache2
```

```
:wq!_
```

Τώρα ξεκινάμε την υπηρεσία του heartbeat: /etc/init.d/heartbeat start

```
root@ubuntu3:/home/mike# /etc/init.d/heartbeat start
Starting High-Availability services: INFO: Resource is stopped
Done.
```

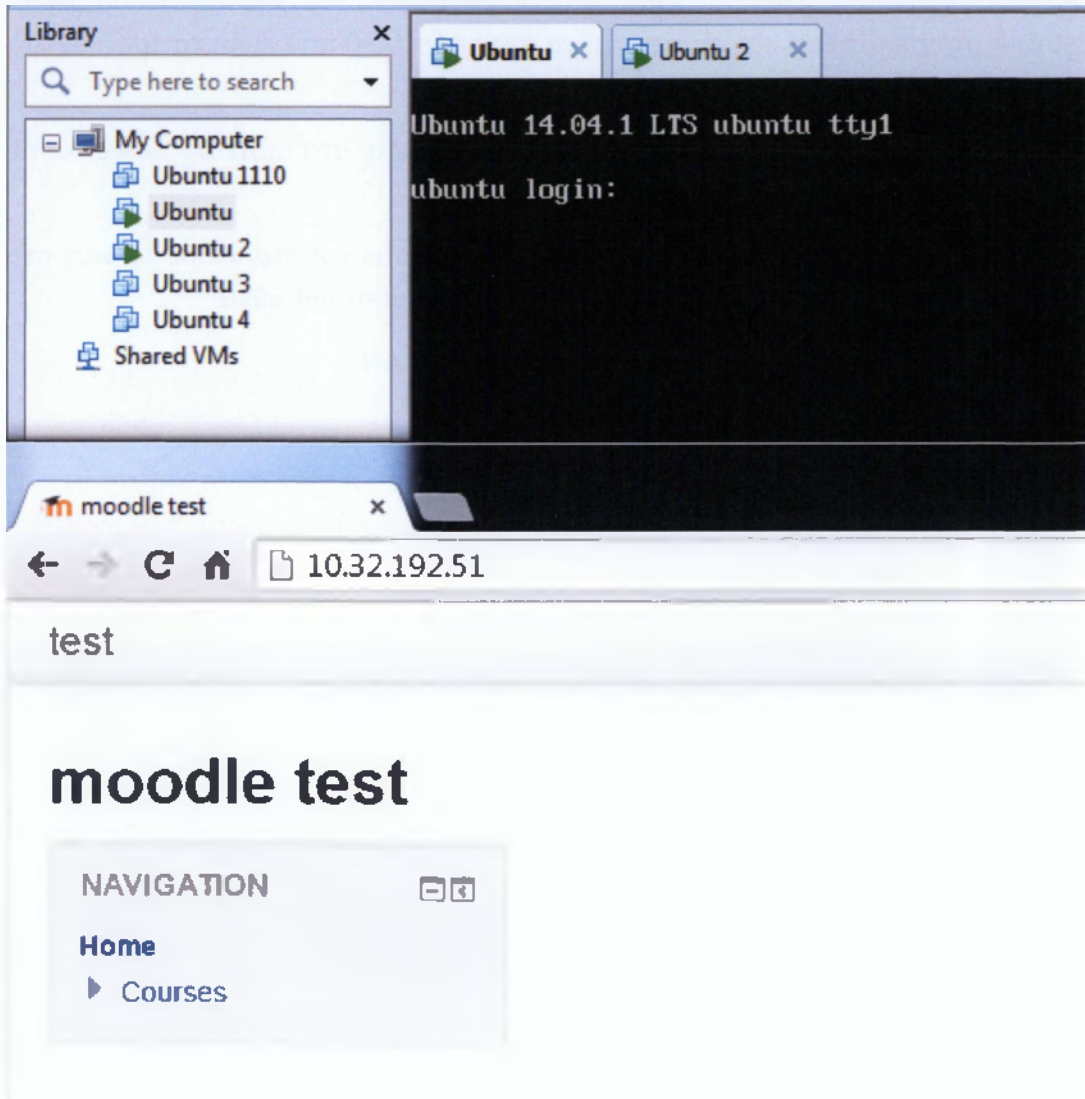
4.5 Αλλαγές που χρειάζονται για να λειτουργήσει το HA clustering του moodle μέσω internet.

- 1) Αρχικά πήρα ένα hostname από το <http://www.noip.com>
 - 2) Ενεργοποίησα το Dynamic DNS στο router μου και έκανα τις απαραίτητες ρυθμίσεις ώστε να έχω service provider το “noip” με hostname το “mike91.ddns.net”
 - 3) Ρύθμισα στο router πάλι κάποιες δρομολογήσεις...Δηλαδή όταν κάποιος χτυπήσει την πόρτα 8080 έξω από το LAN να τον στέλνει στην 10.32.192.56 (που έχω ορίσει ως cluster IP στο heartbeat) και συγκεκριμένα στην πόρτα 80 που είναι το apache2 και στους 2 servers.
 - 4)Με vi /etc/apache2/apache2.conf προσθέτω κάπου στο αρχείο “ServerName ubuntu3” (αντίστοιχα στον άλλο server)
 - 5)Με vi /var/www/html/moodle/config.php αντικαθιστώ και στους δύο servers στο σημείο που γράφεται η IP του καθένα, με το “mike91.ddns.net:8080”
- Έτοιμο και δουλεύει κανονικά το clustering και εκτός LAN...

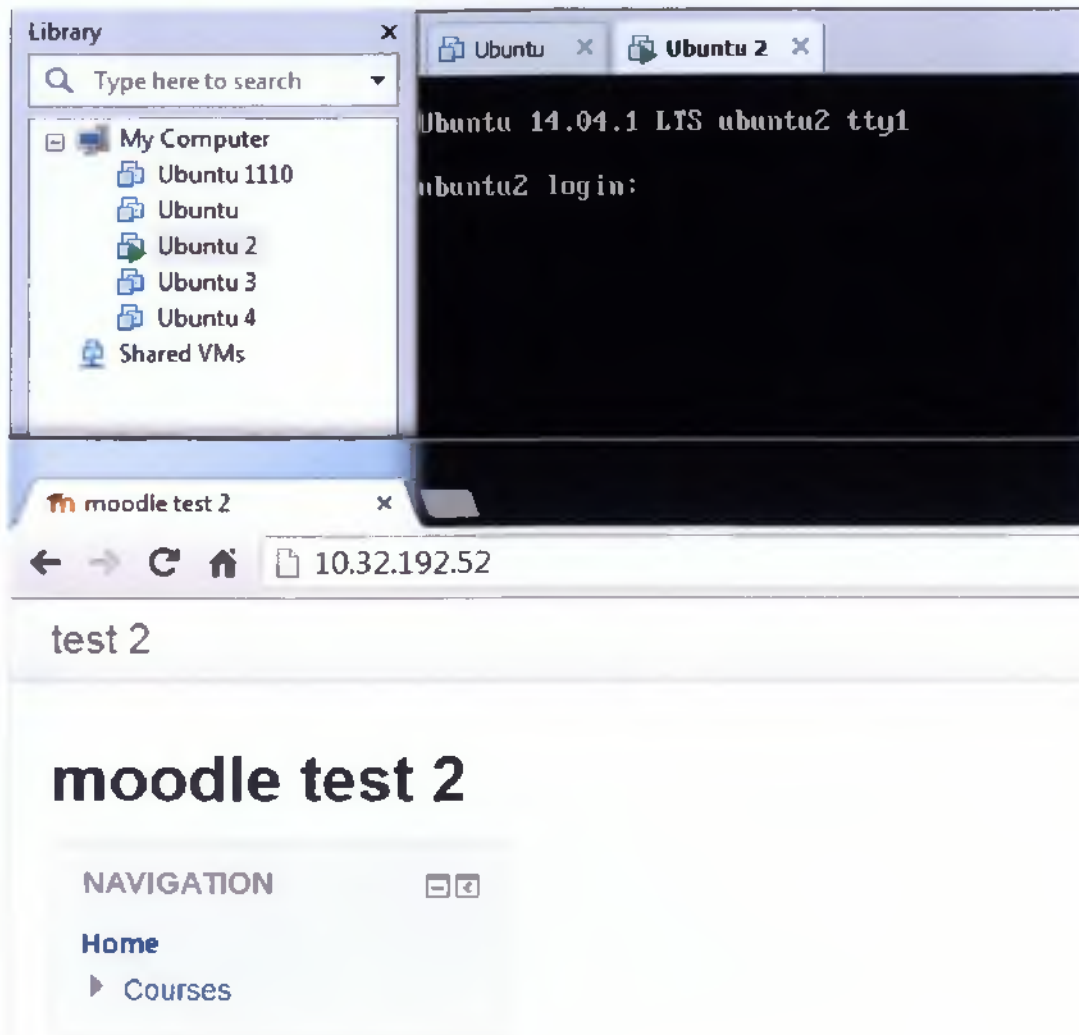
4.6 Παράδειγμα εφαρμογής του HA clustering στο LAN

Σε αυτό το παράδειγμα χρησιμοποιώ τους Ubuntu (10.32.192.51) και ubuntu2 (10.32.192.52) που είναι ρυθμισμένοι με clustering (cluster IP 10.32.192.53) τοπικά στο LAN.

Όταν είναι και οι δύο ανοιχτοί (ubuntu & ubuntu2) και φορτώσω την cluster IP στον browser λαμβάνω ένα μήνυμα redirect και ανοίγει η 10.32.192.51 που είναι και η IP του ορισμένου ως κύριου server (ubuntu)

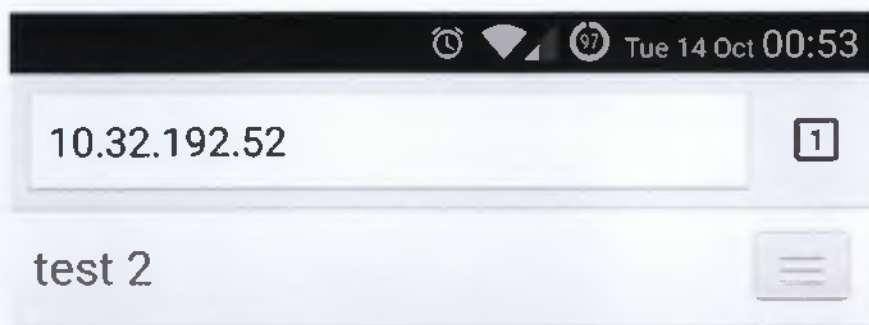


Μόλις κλείσω τον κύριο server (ubuntu) και φορτώσω την cluster IP στον browser τότε:



Όπως και έπρεπε, από την στιγμή που έκλεισα τον κύριο server έγινε redirect στον δεύτερο server. Βέβαια μόλις ο ubuntu server έρθει ξανά σε λειτουργία και φορτώσω την cluster IP ξανά πάλι εκείνος, ως ο κύριος server, θα επανέλθει σε λειτουργία από το redirect.

Πρόσβαση έχουν επίσης όλες οι κινητές συσκευές. Ενδεικτικό παράδειγμα από smartphone παρακάτω:



moodle test 2

NAVIGATION



Home

▶ Courses

CALENDAR



← October 2014 →

<u>Sun</u>	<u>Mon</u>	<u>Tue</u>	<u>Wed</u>	<u>Thu</u>	<u>Fri</u>	<u>Sat</u>
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Βιβλιογραφία

- [1]- Virtualization: A long brief history, URL: <http://www.servethehome.com/virtualization-long-history>
- [2]- An overview of virtualization methods, architectures and implementations, URL: <http://www.ibm.com/developerworks/library/l-linuxvirt/>
- [3]- Matthew Portnoy (2012). Virtualization Essentials, John Wiley & Sons Inc, pp 19-25.
- [4]- John Hoopes (2009), Virtualization for Security Including Sandboxing, Disaster Recovery, High Availability, Forensic Analysis, and Honeypotting. Syngress Publishing, Inc. pp 20-40.
- [5]- Matthew Portnoy (2012). Virtualization Essentials
- [6]- <http://searchservvirtualization.techtarget.com/definition/virtual-machine>
- [7]- Shankar Kolosu, Virtualization of the Desktop, White Paper. (2010)
<http://www.slideshare.net/Cameroon45/microsoft-word-whitepaper-desktop-virtualization>
- [8]- Paulus Kampert (2010), a taxonomy of virtualization Technologies. Master Thesis August 2010 Delft University of Technology .pp 10-15
- [9]- Ruben Spruijt, Desktop virtualization and the power of App-V and Windows 7, Article (2010), <http://www.brianmadden.com/blogs/rubenspruijt/archive/2010/02/22/desktop-virtualization-and-the-power-of-windows-7.aspx>
- [10]- <http://searchitchannel.techtarget.com/feature/Storage-virtualization-solutions-explained>
- [12]- Wikipedia. Application virtualization URL:
http://en.wikipedia.org/wiki/Application_virtualization#Benefits_of_application_virtualization
- [13]- "Defining and Measuring Cloud Elasticity". KIT Software Quality Department
- [14]- "Cloud Computing: Small Companies Take Flight". Bloomberg BusinessWeek.
- [15]- "Defining 'Cloud Services' and 'Cloud Computing'". IDC. 2008-09-23
- [16]- <http://www.businessweek.com/stories/2006-11-12/jeff-bezos-risky-bet>
- [17]- <http://cloudandcompute.com/private-cloud/private-cloud-more-secure/>
- [18]- <http://www.govconnection.com/technology-services/self-run-private-cloud>
- [19]- <http://www.informationweek.com/private-clouds-take-shape/d/d-id/1070793?>
- [20]- <http://www.cnet.com/news/just-dont-call-them-private-clouds/>
- [21]- <http://www.networkcomputing.com/cloud-infrastructure/theres-no-such-thing-as-a-private-cloud/a/d-id/1075465?>
- [22]- <http://searchcloudcomputing.techtarget.com/definition/public-cloud>
- [23]- Metzler, Jim; Taylor, Steve. (2010-08-23) "Cloud computing: Reality vs. fiction," Network World.