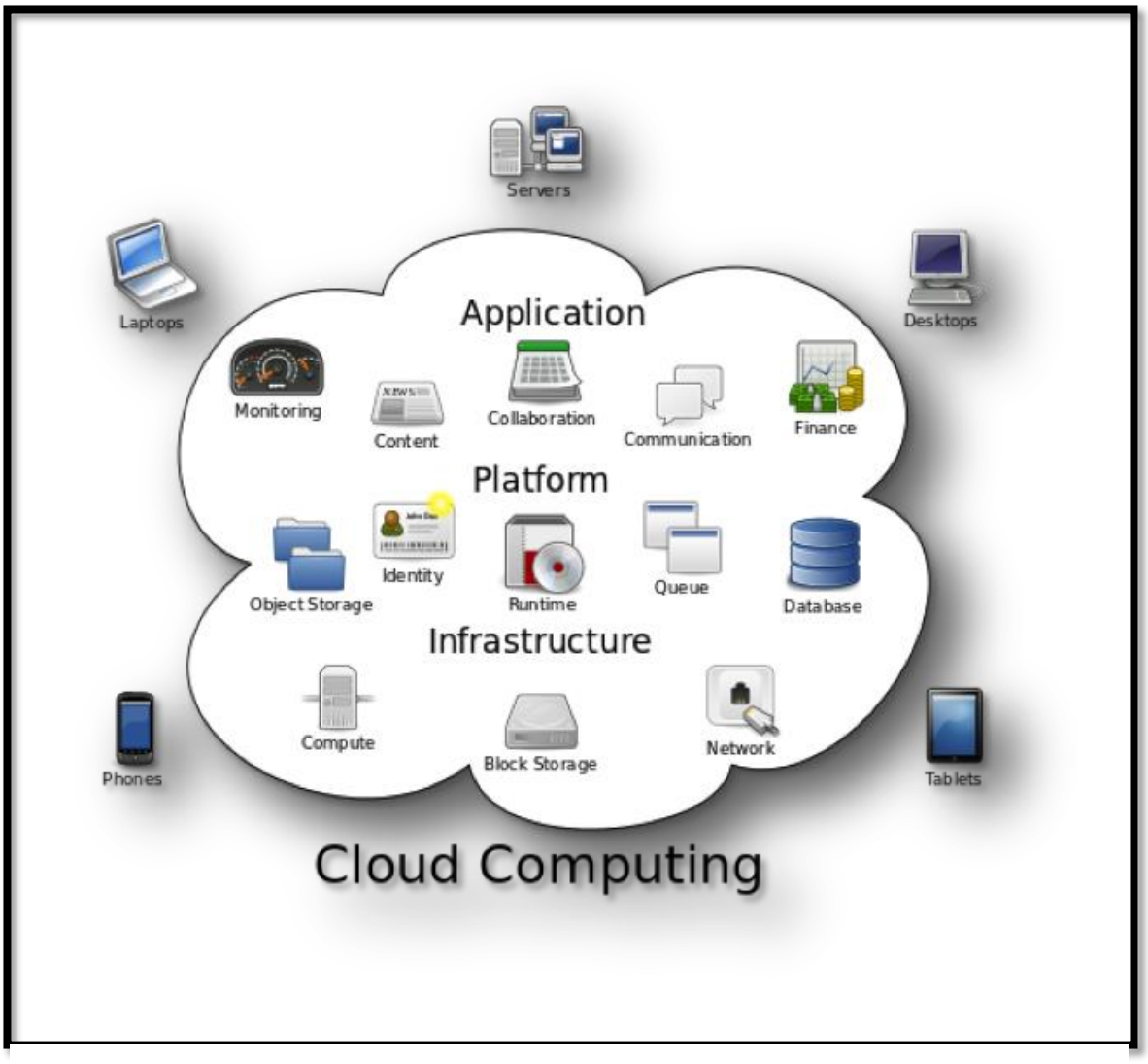
Cloud computing
από πλευρά ελέγχου

Φοίβος Άγγελος Χαραλαμπάκος (3170175), Κοκκίνη Μαρία Ελένη (3170070)
Επαλήθευση επικύρωση και συντήρησή λογισμικού
2019-2020



Περιεχόμενα:

Εισαγωγή: 2

Ορισμός cloud : 2

Αρχιτεκτονική του cloud: 3

Μορφές υπολογιστικού νέφους(Delivery models): 3

Οι τύποι του cloud (και περιβάλλοντα cloud testing): 5

Πλεονεκτήματα ύπαρξης του cloud computing: 6

Μειονεκτήματα του cloud: 6

Cloud testing: 8

Ανάγκη ύπαρξης του cloud testing (για το ίδιο το cloud): 8

Βασικά χαρακτηριστικά του cloud testing: 8

Αρχιτεκτονική ελέγχων cloud: 11

Μορφές ελέγχου cloud (για το ίδιο το cloud και για εφαρμογές): 14

Διάκριση ελέγχων εφαρμογών: 14

Σημαντικά στοιχεία κατά το cloud testing: 14

Περιεχόμενα του cloud testing: 15

Τύποι ελέγχου: 16

Κενά στους ελέγχους: 20

Βήματα για το cloud testing (για εφαρμογές του cloud): 21

Εργαλεία για έλεγχο cloud (για τις εφαρμογές του clouds): 21

Μετρικές για την αξιολόγηση του cloud: 22

Οφέλη από τον έλεγχο του Cloud (για τις εφαρμογές του cloud): 23

Μειονεκτήματα του cloud testing (για τις εφαρμογές του cloud): 25

Συμβουλές για ορθό cloud testing: 27

Σύγκριση ελέγχων στο cloud με τους συμβατικούς ελέγχους: 29

Ερωτήματα που μας βοηθούν τα προσδιορίσουμε τις κατάλληλες πολιτικές για το cloud testing: 30

Αναπάντητα ερευνητικά ερωτήματα σχετικά με το cloud testing: 31

Συμπεράσματα: 32

Bibliography.....32

Εισαγωγή:

Το cloud computing δημιουργήθηκε ως ένα νέο υπολογιστικό παράδειγμα που διευκολύνει την ανάπτυξη και χρήση ευέλικτων και ευπροσάρμοστων υπηρεσιών σε συνθήκες πραγματικού χρόνου. Αυτά τα χαρακτηριστικά ωθούν πολλούς οργανισμούς να μεταφέρουν την επιχείρηση τους σε μια πλατφόρμα νέφους. Ο έλεγχος λογισμικού αποτελεί για παράδειγμα δραστηριότητας που μεταφέρεται στο περιβάλλον του cloud. Διότι ο έλεγχος λογισμικού, συχνά απαιτεί ακριβούς διακομιστές, αποθηκευτικούς χώρους και δικτυακούς πόρους για συγκεκριμένο βέβαια διάστημα. Εάν δεν χρησιμοποιηθεί το περιβάλλον του cloud, αυτοί οι υπολογιστικοί πόροι μετά τον έλεγχο είτε δε χρησιμοποιούνται άλλο, είτε υπο-χρησιμοποιούνται με αποτέλεσμα να επιφέρουν επιπλέον κόστος στο συνολικό έργο.

Ορισμός cloud :

Το υπολογιστικό νέφος είναι η παροχή υπολογιστικών πόρων με τη βοήθεια του διαδικτύου από απομακρυσμένα κεντρικά συστήματα, που εξυπηρετούν τον τελικό χρήστη αυτοματοποιώντας διαδικασίες και προσφέροντας ευκολίες και ευελιξία κατά τη σύνδεση. [1]

Ο όρος cloud computing είναι σχετικά πρόσφατος. Περιγράφει ουσιαστικά ένα νέο τρόπο για παράδοση κάποιας υπηρεσίας που βασίζεται σε υπολογιστή. Επιτρέπει μια ευρεία, βολική και on-demand πρόσβαση σε ένα δίκτυο διαμοιραζόμενων πόρων(π.χ servers,applications).

Βασίζεται σε δύο αρχές. Η πρώτη είναι η είναι η αρχιτεκτονική του, η οποία είναι service-based, και επιτρέπει την παράδοση μιας σειράς λειτουργιών στον τελικό χρήστη. Η αρχιτεκτονική αυτή επιτρέπει στους χρήστες να ψάχνουν και να χρησιμοποιούν υπηρεσίες σε πραγματικό χρόνο και ποιοτικά.

Η δεύτερη αρχή στην οποία βασίζεται το cloud computing είναι η εικονικοποίηση. Η εικονικοποίηση επιτρέπει ένα είδος αφαίρεσης και

απομόνωσης των κατώτερων επιπέδων και του hardware και επιτρέπει επίσης μεταφερσιμότητα των υψηλότερων επιπέδων και των λειτουργιών που παρέχουν.

Οι πελάτες πληρώνουν μόνο για όσο χρόνο χρησιμοποιούν μια υπηρεσία, συνεπώς δεν χρειάζεται να επενδύσουν σε αχρείαστους υπολογιστικούς πόρους. Το cloud computing προσφέρει οφέλη και στους παρόχους καθώς μπορούν να χρησιμοποιήσουν τους πόρους τους για να εξυπηρετήσουν πολλούς πελάτες.

Σημαντικά χαρακτηριστικά του cloud: [5]

1) Ελαστικότητα
2) Επεκτασιμότητα
3) Multi-tenancy
4) Αυτοδιαχειριζόμενες λειτουργικές ικανότητες
5) Μέτρηση χρέωσης και χρήσης υπηρεσίες (προαναφέρθηκε)

Αρχιτεκτονική του cloud:

[1] Η αρχιτεκτονική του cloud αποτελείται από δύο βασικά στοιχεία τα οποία συνδέονται μεταξύ τους μέσω του διαδικτύου:

1) **Front end platform**: ονομάζονται επίσης πελάτες (cloud clients) και αλληλοεπιδρούν με την αποθήκευση δεδομένων cloud μέσω του middleware και μέσω κάποιου προγράμματος περιήγησης.
2) **Back end platform**: Είναι ένας ηλεκτρονικός αποθηκευτικός χώρος όπου κρατούνται τα δεδομένα και διατίθενται στους πελάτες.

Μορφές υπολογιστικού νέφους(Delivery models):

Το υπολογιστικό νέφος χωρίζεται σε τέσσερεις βασικές κατηγορίες μοντέλων: [1], [2]

1. **Software-as-a-Service (SaaS):** Προκειμένου να αποφευχθούν τυχόν επιβαρύνσεις από συχνές επιδιορθώσεις, νέες εκδόσεις κ.α., ο πελάτης αντί να εγκαταστήσει στο μηχάνημα του λογισμικό και εφαρμογές όπως το Word, CRM κτλ., αυτές διατίθενται μέσω του διαδικτύου για την κατανάλωση του τελικού χρήστη. Αυτές οι

εφαρμογές είναι προσβάσιμες από ποικίλες πλατφόρμες μέσω μιας απλής διεπαφής πελάτη, όπως είναι οι web browsers.

2. **Platform-as-a-Service (PaaS):** Ο τελικός χρήστης αντί να πρέπει να εξαγοράσει τις άδειες χρήσης του λογισμικού για τις διάφορες πλατφόρμες αλλά και τα λειτουργικά συστήματα, μπορεί να χρησιμοποιήσει την πλατφόρμα και τα εργαλεία για παράδειγμα Java, .Net και άλλα τέτοια εργαλεία. Έτσι οι πελάτες μπορούν να αναπτύξουν τις δικές τους "λύσεις" στο νέφος μέσω διάφορων πλατφόρμων όπως είναι οι διακομιστές εφαρμογών και οι υπηρεσίες των βάσεων δεδομένων που παρέχονται από το cloud.

3. **Infrastructure-as-a-Service (IaaS):** Αυτό το κομμάτι αφορά τις βασικές υλικές συσκευές, για παράδειγμα Virtual Machines, μέσα αποθήκευσης, servers που βρίσκονται σε ένα κέντρο δεδομένων. Αυτές οι υλικές συσκευές μπορούν να προσπεραστούν αλλά και να χρησιμοποιηθούν από το διαδίκτυο με τη χρήση συστήματα ελέγχου ταυτότητας σύνδεσης σε συνδυασμό με τους κωδικούς πρόσβασης από κάθε συσκευή. Αυτό λοιπόν είναι το χαμηλότερο επίπεδο υπηρεσιών και σε αυτό παρέχονται στους πελάτες εργαλεία για να μπορούν να αναπτύξουν τις δικές τους υπηρεσίες και να αναπτύξουν τα δικά τους συστήματα και πιθανόν να αντιγράψουν τις δικές τους ήδη υπάρχουσες υποδομές. Το επίπεδο αυτό μπορεί ακόμη να ενεργοποιήσει τη συμβατότητα συστήματος και λογισμικού παλαιού τύπου.

4. **Desktop-as-a-Service (DaaS):** Το συγκεκριμένο μοντέλο χρησιμοποιεί μια συγκεκριμένη αρχιτεκτονική με την οποία μια μοναδική εμφάνιση κάποιας εφαρμογής εξυπηρετείται σε πολλούς χρήστες, τους "ενοικιαστές". Ο πάροχος του cloud πρέπει να διαχειρίζεται το νέφος και την υποκείμενη υποδομή και να φροντίζει το επίπεδο εξυπηρέτησης να διαφέρει ανάλογα με τις απαιτήσεις κάθε χρήστη. Συνεπώς οι χρήστες μπορούν να αποκτούν πρόσβαση στα δεδομένα και τις εφαρμογές τους οπουδήποτε και από οποιαδήποτε συσκευή.

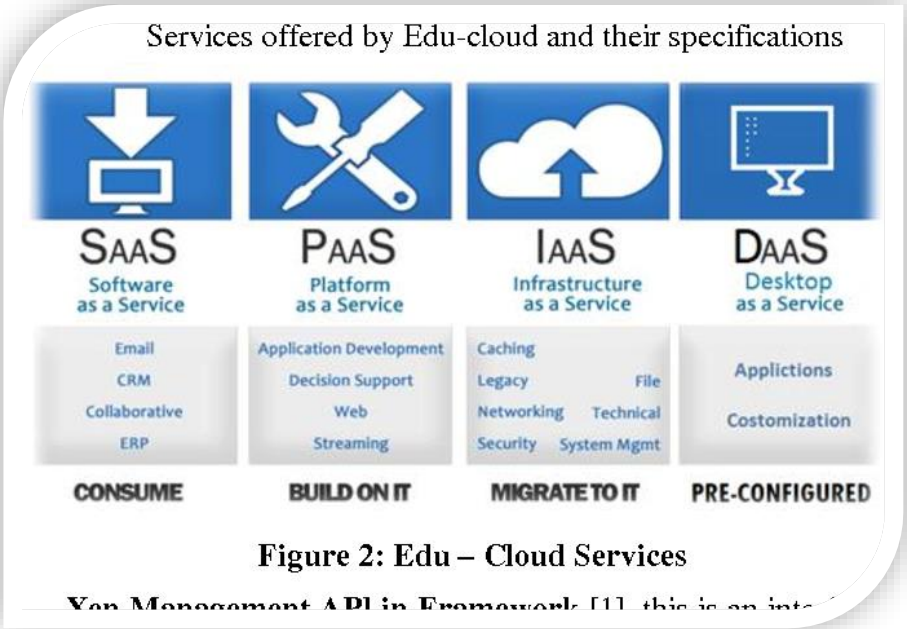


Figure 2: Edu – Cloud Services

Xen Management API in Framework [1], this is an inte

Οι τύποι του cloud (και περιβάλλοντα cloud testing):

Το Cloud μπορεί να εμπίπτει σε μία από τις εξής κατηγορίες ανάλογα με τους περιορισμούς που το διέπουν: [2]

1) **Public clouds**: οι υπηρεσίες που παρέχονται είναι διαθέσιμες σε όλους, οι πόροι διατίθενται και παρέχονται δυναμικά μετά από κάποιο αίτημα χρήστη.
2) **Private clouds**: οι υπηρεσίες διαχειρίζονται εντός του τείχους προστασίας ενός συγκεκριμένου οργανισμού και είναι διαθέσιμες μόνο στους χρήστες εντός αυτού του οργανισμού.
3) **Community Cloud:** Χρησιμοποιείται και ελέγχεται από μια ομάδα ανθρώπων, οι οποίοι έχουν κοινά ενδιαφέροντα.
4) **Hybrid clouds:** Είναι μια μίξη των κατηγοριών private cloud, public cloud. Οι επιχειρήσεις μπορούν να επιλέξουν ποιες υπηρεσίες θα διαθέσουν σε όλο το κοινό και ποιες υπηρεσίες θέλουν να παρέχουν μόνο στους χρήστες εντός του οργανισμού.

Πλεονεκτήματα ύπαρξης του cloud computing:

[1]

1) Είναι **οικονομικός τρόπος διαχείρισης** των δεδομένων του χρήστη. Το cloud είναι από τις πιο αποδοτικές μεθόδους για την επεξεργασία των πληροφοριών του χρήστη.
2) Παρέχεται **αρκετός διαθέσιμος χώρος για την αποθήκευση** πληροφοριών του χρήστη. Με ένα συμβολικό αντίτιμο, αυξάνεται και ο διαθέσιμος αποθηκευτικός χώρος.
3) Ευκολία **δημιουργίας backup και αποκατάστασης αρχείων**. Δεν χρειάζεται να αποθηκεύονται τα δεδομένα τοπικά, όπου υπάρχει και μεγαλύτερη πιθανότητα για την απώλεια τους αλλά αποθηκεύονται στο cloud και έτσι μπορώ να τα ανακτήσω εύκολα.
4) Το **λογισμικό ενσωματώνεται αυτόματα**, οι χρήστες δεν χρειάζεται να προσαρμόσουν τις εφαρμογές ανάλογα με τις προτιμήσεις του γιατί η εφαρμογή στο νέφος το τακτοποιεί από μόνη της.
5) **Εύκολη και γρήγορη πρόσβαση από παντού**. Δεν υπάρχει κανένας χρονικός ή τοπικός περιορισμός για την προσπέλαση των αποθηκευμένων πληροφοριών.
6) **Scalability**: νέοι χρήστες μπορούν εύκολα να χρησιμοποιήσουν το cloud και μάλιστα ανάλογα με τις απαιτήσεις τους.
   Συνεπώς για τη διατήρηση των παραπάνω χαρακτηριστικών, είναι ανάγκη να πληρούνται σωστοί και τακτικοί έλεγχοι. Ειδικά σήμερα που το Cloud γνωρίζει μεγάλη ακμή και άνοδο, εξυπηρετώντας όλο και περισσότερους χρήστες.

Μειονεκτήματα του cloud:

Η φιλοξενία στο cloud εισάγει ορισμένα ρίσκα στην αξιοπιστία των συστημάτων. Το software αναπτύσσεται απομακρυσμένα σε ένα εικονικό runtime περιβάλλον που χρησιμοποιεί κοινόχρηστο υλικό και λογισμικό και φιλοξενείται σε μια εξωτερική υποδομή. Η ποιότητα και η απόδοση του λογισμικού εξαρτάται σημαντικά από το runtime περιβάλλον εκτέλεσης που είναι συχνά εκτός του ελέγχου των χρηστών. [8] Μάλιστα κάποια από τα βασικότερα μειονεκτήματα σήμερα είναι τα εξής:

- **Εξαρτάται από τη σύνδεση του δικτύου**:
  Για να αποκομίσετε τα οφέλη του cloud computing, πρέπει πάντα να υπάρχει σύνδεση στο Διαδίκτυο. Δυστυχώς, δεν υπάρχει τρόπος να ξεπεραστεί αυτό το γεγονός. Χρειάζεστε ένα δίκτυο για να μπορείτε να χρησιμοποιείτε τις εικονικές σας μηχανές ακόμα κι αν επιλέξετε ένα IaaS. Εάν χάσετε τη σύνδεση δικτύου σας λόγω καταιγίδας ή διακοπής λειτουργίας, ενδέχεται να αντιμετωπίσετε κάποια διακοπή λειτουργίας.
- **Έχει περιορισμένες δυνατότητες**:
  Όταν χρησιμοποιείτε cloud computing για αποθήκευση και δημιουργία αντιγράφων ασφαλείας, ιδανικά θα πρέπει να συνεργάζεστε με έναν πάροχο που προσφέρει την αξία του απεριόριστου εύρους ζώνης. Ενδέχεται επίσης να αντιμετωπίσετε περιορισμένο χώρο αποθήκευσης ή περιορισμένη προσβασιμότητα. Οι προσφορές SaaS συνήθως ξεκινούν με ένα δωρεάν πακέτο, αλλά θα χρεωθείτε για προσφορές premium και επιπλέον χώρο. Η απάντηση στην ανησυχία των περιορισμένων δυνατοτήτων είναι να συνεργαστείτε με έναν πάροχο Hosted Services που μπορεί να καλύψει τις ανάγκες αποθήκευσης στο cloud, εικονικοποίηση και δημιουργία αντιγράφων ασφαλείας τόσο τώρα όσο και στο μέλλον.
- **Υπάρχει το πρόβλημα της απώλειας ελέγχου**: Ουσιαστικά οι πελάτες εμπιστεύονται κάποιον ‘τρίτο’ για να διαφυλάξει τα δεδομένα τους. Πρέπει κανείς να εμπιστευτεί τα κέντρα δεδομένων, τους διακομιστές και τους παρόχους αφήνοντας σε αυτούς τον έλεγχο για την επίβλεψη των δεδομένων του.
- **Το πρόβλημα της ασφάλειας**: Έχει διαπιστωθεί πως δεν παρέχουν όλοι οι πάροχοι υπηρεσιών του cloud τόσο ασφάλεια όσο υπόσχονται. Όμως οι εταιρείες δε μπορούν να ρισκάρουν να διαρρεύσουν τα ευαίσθητα προσωπικά δεδομένα των πελατών τους. Δυστυχώς δε μπορεί κάποιος να προβλέψει ποιους παρόχους να εμπιστευτεί. Το πρόβλημα είναι πιο αισθητό στους SaaS providers σε σύγκριση με τους hosted providers, διότι οι πρώτοι είναι πιο δημοφιλείς από τους δεύτερους και έτσι στοχοποιούνται πιο εύκολα.
- **Τεχνικά προβλήματα**: εάν κάποιος αντιμετωπίζει τεχνικά προβλήματα, δεν έχει άλλη επιλογή παρά να επικοινωνήσει με την τεχνική υποστήριξη του παρόχου. Δε μπορεί δηλαδή κάποιος να επιλύσει το πρόβλημα μόνος του και από την άλλη μεριά μπορεί η υποστήριξη δεν είναι πάντα διαθέσιμη. [9]

Cloud testing:

Υπάγεται ως κατηγορία του Software testing και ελέγχει τις υπηρεσίες που παρέχονται από το cloud computing. [2] Το cloud testing είναι μια μορφή ελέγχου κατά την οποία οι εφαρμογές του ιστού χρησιμοποιούν το περιβάλλον του υπολογιστικού νέφους, προκειμένου να προσομοιώσουν την πραγματική κίνηση των χρηστών χρησιμοποιώντας τεχνολογίες cloud. Το cloud testing παρέχει την ικανότητα ελέγχου στις υποδομές του νέφους όπως είναι το hardware. Αυτή η μορφή ελέγχου αναφέρεται στην επαλήθευση και επικύρωση των εφαρμογών και υποδομών που είναι διαθέσιμα σε πραγματικό χρόνο. Ο έλεγχος του νέφους ορίζεται ως έλεγχος υπηρεσιών (Testing-as-a-Service→TaaS). Το μοντέλο TaaS θεωρείται ένα καινοτόμο μοντέλο στο οποίο ο πάροχος του cloud αναλαμβάνει τις δραστηριότητές του software ελέγχου μιας εφαρμογής της υποδομής του Cloud. Το cloud testing εμπεριέχει τον έλεγχο της διαθεσιμότητας, της ασφάλειας, της απόδοσης, της διαλειτουργικότητας του συστήματος και τον έλεγχο ανάκαμψης μετά από καταστροφές. [3]

Ανάγκη ύπαρξης του cloud testing (για το ίδιο το cloud):

Όταν κάνουμε έλεγχο νέφους αναφερόμαστε στον έλεγχο των πόρων του, όπως είναι το hardware, το software και των επί μέρους στοιχείων που πρέπει να είναι διαθέσιμα σε συνθήκες πραγματικού χρόνου. Είναι λοιπόν σημαντικό όταν χρησιμοποιούμε το cloud για να κάνουμε ελέγχους (με την μορφή του cloud "as service") εντός του cloud, να εξασφαλίσουμε ότι οι υπηρεσίες του νέφους πληρούν τις λειτουργικές και μη λειτουργικές απαιτήσεις. [2]

Βασικά χαρακτηριστικά του cloud testing:

Το cloud testing διέπεται από τα παρακάτω χαρακτηριστικά: [8]

- ✓ **Embedded continuous testing for SaaS**: Συνήθως το λογισμικό τύπου SaaS χρησιμοποιεί MTA (message transfer agent→είναι ένα πρόγραμμα για να λαμβάνει τα εισερχόμενα e-mails και να προωθεί τα μηνύματα στους χρήστες) εντός του cloud. [10] Είναι πιθανό να ενσωματωθούν υπηρεσίες παρακολούθησης και ελέγχου στο cloud που εντοπίζει τις εισόδους και τις εξόδους των υπηρεσιών όποτε είναι

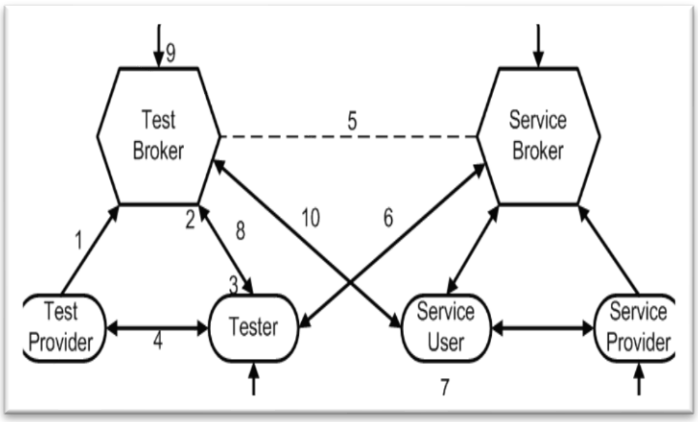
ενεργοποιημένες για εκτέλεση. Όταν συλλεχθούν επαρκείς είσοδοι και έξοδοι παράγονται τα στατιστικά.

- ✓ **Test cases generation from metadata**: Βασικό στοιχείο του PaaS, SaaS είναι ότι το σύστημα ελέγχεται από μεταδεδομένα (μια χρήση μεταδεδομένων είναι για παράδειγμα η συντήρηση και ο εντοπισμός της κατάστασης του συστήματος). Επιπλέον η αποτυχία κόμβων που διαχειρίζονται μεταδεδομένα μπορεί να επιφέρει καταστροφικά αποτελέσματα στο σύστημα. Τα μεταδεδομένα, χρησιμοποιούνται επιπλέον για να παράξουμε περιπτώσεις ελέγχου. Αυτές οι περιπτώσεις παράγονται εξετάζοντας τα μεταδεδομένα.
- ✓ **Policy enforcement as testing**: οι πολιτικές είναι προσανατολισμένες στις υπηρεσίες. Οι πολιτικές μπορούν να εφαρμοστούν κατά το χρόνο εκτέλεσης στο Cloud. Οι πολιτικές μπορούν επίσης να χρησιμοποιηθούν για την αυτόματη παραγωγή μεταδεδομένων. Είναι άρρηκτα συνδεδεμένο κομμάτι με τους ελέγχους και την αξιολόγηση των εφαρμογών του νέφους.
- ✓ **Scalability metrics**: Στο cloud computing χρειάζεται ο ορισμός νέων μετρικών και όχι οι παραδοσιακές μετρικές επεκτασιμότητας. Οι νέες αυτές μετρικές λαμβάνουν υπόψιν τους και τις επιδόσεις κέρδους συγκριτικά με τις ανάγκες πόρων.
    - Η μετρική T επεξεργάζεται τον χρόνο και αντανακλά την παραδοσιακή επιτάχυνση.
    - R^*T_r είναι η κατανάλωση πόρων και αντιπροσωπεύει την χρήση πόρων στο σύστημα.
    - PRR είναι η αναλογία πόρων ως προς την απόδοση και αντικατοπτρίζει τη σχέση μεταξύ απόδοσης και των πόρων που χρησιμοποιούνται.
    - Η μετρική της διακύμανση, όπως η διακύμανση της ταχύτητας, η διακύμανση της κατανάλωσης πόρων και η διακύμανση του PRR.
- ✓ **Multi-layer testing**: Ενδέχεται να υπάρχουν σφάλματα σε διάφορα στοιχεία του cloud, όπως υλικό, δίκτυο, διαχείριση εικονικοποίησης και σύστημα αποθήκευσης. Ως ένα κατανεμημένο σύστημα μεγάλης κλίμακας, ένα cloud μπορεί επίσης να έχει πολύπλοκους μηχανισμούς αντοχής σε σφάλματα και αποκατάστασης αστοχίας. Είναι δύσκολο να εντοπίσετε σφάλματα σε περίπτωση αποτυχίας μιας εφαρμογής. Για διεξοδική ανάλυση, πρέπει να πραγματοποιούνται δοκιμές σε κάθε στοιχείο σε όλα αυτά τα επίπεδα. Κάθε στρώμα απαιτεί διαφορετικές δοκιμές και τεχνικές.

- ✓ **SLA-based testing**: Για συμβατικό λογισμικό, ο έλεγχος βασίζεται στον πηγαίο κώδικα ή τις προδιαγραφές λογισμικού που περιγράφουν την αναμενόμενη συμπεριφορά λογισμικού χρησιμοποιώντας φυσική γλώσσα ή επίσημα μοντέλα. Για λογισμικό που αναπτύσσεται στο cloud, ο πηγαίος κώδικας ενδέχεται να μην είναι διαθέσιμος. Αντίθετα, η SLA (service-level agreement) είναι διαπραγματευόμενη μεταξύ παρόχων λογισμικού και υποδομής, συμπεριλαμβανομένων λειτουργιών. Η SLA παρέχει έτσι τη βάση,όχι μόνο για την παροχή cloud, αλλά και για το σχεδιασμό, την εκτέλεση και την αξιολόγηση δοκιμών.
- ✓ **Large-scale simulation**: Ο έλεγχος πρέπει να προσομοιώνει διάφορες εισόδους και σενάρια. Ως ανοιχτή πλατφόρμα, ένα δημόσιο cloud επιτρέπει ευέλικτη πρόσβαση και λειτουργία. Ο αριθμός πιθανών σεναρίων χρήσης είναι τεράστιος. Το φορτίο είναι υψηλό και απρόσμενο και μπορεί να προκύψουν μεγάλες διακυμάνσεις. Για να ελεγχθεί η λειτουργικότητα και η απόδοση τέτοιων πολύπλοκων συστημάτων, απαιτείται προσομοίωση μεγάλης κλίμακας. Στην πραγματικότητα, πρέπει να προσομοιώσει όχι μόνο τη χρήση του συστήματος, αλλά και τις αλλαγές στο περιβάλλον, όπως διαμορφώσεις υποδομής.
- ✓ **On-demand test environment**: Οι δοκιμές πρέπει να ενεργοποιούνται στο διαδίκτυο κάθε φορά που συμβαίνει μια αλλαγή στο cloud, συμπεριλαμβανομένης της εφαρμογής, του περιβάλλοντος εκτέλεσης και της υποδομής. Χρειάζεται συνήθως προσπάθεια για τη δημιουργία περιβάλλοντος δοκιμών και τη διατήρησή του για regression testing. Για να δοκιμαστεί το cloud κάτω από διάφορα σενάρια χρήσης, είναι απαραίτητο να σχεδιαστούν μηχανισμοί δοκιμών έτσι ώστε διάφορα περιβάλλοντα δοκιμών να μπορούν να αναπτυχθούν ή να χρησιμοποιηθούν όταν χρειαστεί.

Αρχιτεκτονική ελέγχων cloud:

- **<u>Συνεργατική επαλήθευση και επικύρωση</u>**: όπως σε ένα σύστημα προσανατολισμένο στις υπηρεσίες, οι εφαρμογές στο νέφος αποτελούνται από υπηρεσίες ανεπτυγμένες από διαφορετικά μέλη. Για να δουλέψει μια εφαρμογή, κάθε μέλος χρειάζεται να στείλει τα κομμάτια του για να γίνει η ένωση τους επιτυχώς. Για αυτό για την επαλήθευση και επικύρωση της εφαρμογής χρειάζεται να γίνεται συνεισφορά από όλα τα μέλη με ένα συνεργατικό τρόπο. Οι πιο ισχυροί έλεγχοι θα χρησιμοποιηθούν αρχικά για το νέο λογισμικό έτσι ώστε να μειωθεί ο φόρτος του ελέγχου. Οι κλασσικές ενδιάμεσες διαδικασίες μπορούν να επεκταθούν με τις διαδικασίες ελέγχου "check-in", "check-out". Οι check-in" έλεγχοι επιβεβαιώνουν ότι μόνο οι κατάλληλοι έλεγχοι θα είναι αποδεκτοί για δημοσιοποίηση, ενώ οι "check-out" έλεγχοι στις υποψήφιες υπηρεσίες αξιολογούν αιτήματα εάν οι υπηρεσίες έχουν αλλαχθεί μετά την δημοσίευση τους. Οι test brokers ενεργοποιούν επεκτάσιμες και ευέλικτες συνεργασίες ανάμεσα στους συμμετέχοντες που κάνουν τους ελέγχους. Ένας ελεγκτής γνωρίζει τα δημοσιευμένα σενάρια ελέγχου και προσομοιώνει ελέγχους στις υπηρεσίες που χρειάζονται. Οι συμφωνίες απαρτίζονται από δύο μέλη: A)TSC (Testing Service Contracts), B)TCC (Test Collaboration Contracts). Τα TSC είναι η επικοινωνία μεταξύ των εξεταζόμενων components και των SUTs (service under test). Τα TCC καθορίζουν τον τρόπο που τα ελεγχόμενα components συνεργατικά σχεδιάζουν τα σενάρια ελέγχων, εκτελούν το πλάνο ελέγχου και αξιολογούν τα αποτελέσματα ελέγχου. [8]



- **<u>Testing as a Service (TaaS):</u>** Για να μειωθεί το κόστος του σχεδιασμού, της εκτέλεσης και της συντήρησης, προτάθηκε η ιδέα TaaS (Testing as-a- Service) για τη δημιουργία ενός ενιαίου πλαισίου βασισμένου στην υπηρεσία για την προώθηση της επαναχρησιμοποίησης όλων των αντικειμένων ελέγχου. Παρέχει στατικές / δυναμικές υπηρεσίες δοκιμών σε πραγματικό χρόνο in / on / over cloud για τους πελάτες ανά πάσα

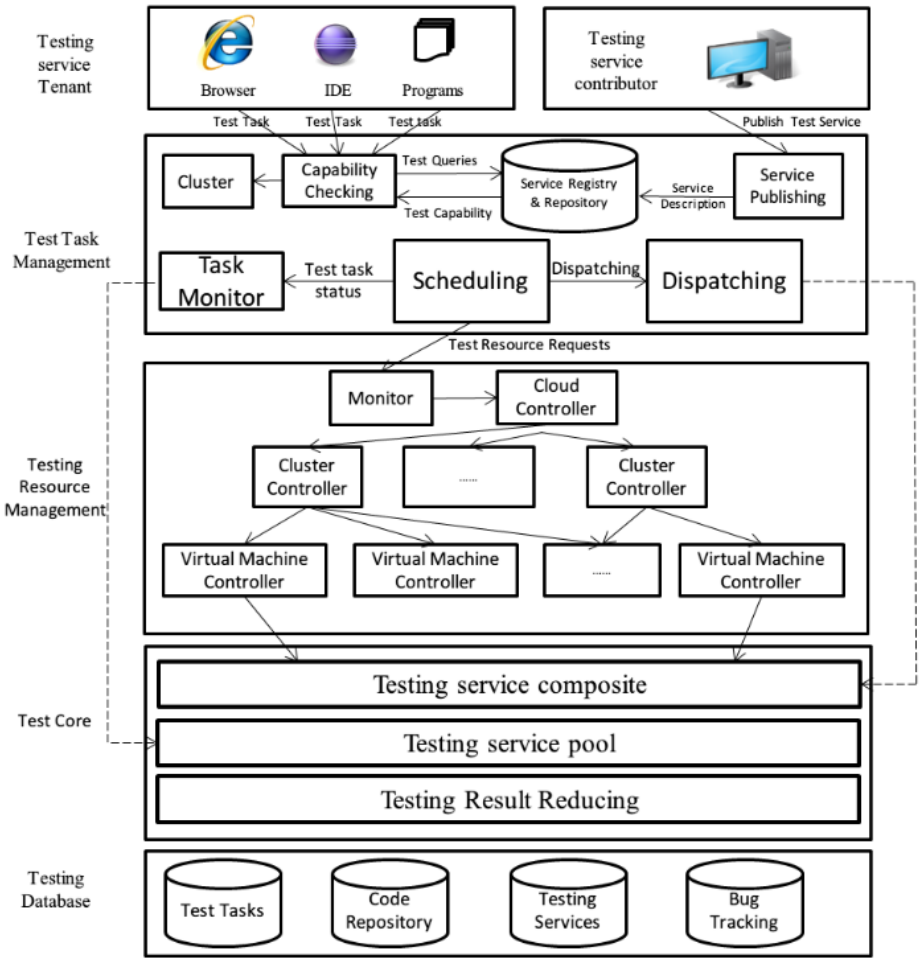
στιγμή. Η ιδέα του TaaS παρουσιάστηκε αρχικά από την Tieto στη Δανία το 2009. Το TaaS έχει λάβει μεγάλη προσοχή λόγω των επεκτάσιμων περιβαλλόντων δοκιμών, της μείωσης του κόστους, των μοντέλων υπηρεσιών που βασίζονται σε βοηθητικά προγράμματα και των υπηρεσιών δοκιμών σε πραγματικό χρόνο.

Έχουν οριστεί 5 layers στο TaaS:

1. Test service tenant and contributor layer: Αυτό το επίπεδο παρέχει λειτουργικότητα που υποστηρίζει τον ενοικιαστή υπηρεσιών ελέγχων για αλληλεπίδραση με το TaaS.
2. Test task management layer: Αυτό το επίπεδο είναι επίπεδο middleware, υποστηρίζοντας το μητρώο υπηρεσιών και το αποθετήριο, τον προγραμματισμό και την αποστολή εργασιών ελέγχων.
3. Testing resource management layer: Αυτό το επίπεδο λειτουργεί ως υποδομή του cloud, αναλαμβάνοντας την ευθύνη της διαχείρισης πόρων και της παρακολούθησης.
4. Test layer: Αυτό το στρώμα είναι ο πυρήνας της πλατφόρμας, που αποτελείται από σύνθεση υπηρεσίας, συνένωση υπηρεσιών και το test-reduce sublayer.
5. Testing database layer: Αυτό το επίπεδο χρησιμοποιείται για την αποθήκευση της εργασίας των ενοικιαστών που βρίσκεται υπό έλεγχο και των αποτελεσμάτων παρακολούθησης σφαλμάτων.

Επίσης, έχουν οριστεί 3 κατηγορίες υπηρεσιών ελέγχου:

1. $TaaS_D$ για προγραμματιστές.
2. $TaaS_H$ για τελικούς χρήστες.
3. $TaaS_C$ ως υπηρεσία πιστοποίησης.



- **<u>Υποστήριξη ελέγχου as-a-Service (TSaaS):</u>** Για να βελτιωθεί η ελεγξιμότητα των αυτόματων υπηρεσιών, TSaaS προτάθηκε έτσι ώστε κάθε υπηρεσία να παρέχει παραγωγικότητα και περιβάλλον ελέγχου στους εξωτερικούς χρήστες. Οι συναρτήσεις ελέγχων παρέχονται σαν API υπηρεσιών. Η τεχνική autonomic self-testing μεταφέρθηκε στην πλατφόρμα του cloud και ονομάστηκε TSaaS. Οι λειτουργίες ελέγχου υποστηρίζουν υπηρεσίες συμπεριλαμβανομένου του ελέγχου εκκίνησης, ελέγχου εισαγωγής κτλ. Αυτές οι υπηρεσίες παρέχονται στους εταίρους του cloud κατά την ανάπτυξη, τον έλεγχο και τη συντήρηση του νέφους προσαρμοσμένες στις εφαρμογές και τις υπηρεσίες του cloud.

Μορφές ελέγχου cloud (για το ίδιο το cloud και για εφαρμογές):

Ο έλεγχος του cloud μπορεί να διαμοιραστεί σε τέσσερεις βασικές κατηγορίες, ανάλογα με το στόχο που υπάρχει κάθε φορά: [2]

1) **Testing of the whole cloud:** Εδώ το νέφος αντιμετωπίζεται σαν μία συνολική οντότητα που βασίζεται στα χαρακτηριστικά του και οι έλεγχοι που πραγματοποιούνται βασίζονται επίσης σε αυτό.
2) **Testing within a cloud:** Αυτός είναι ο έλεγχος που διεξάγεται μέσα στο νέφος, ελέγχοντας καθ' ένα από τα εσωτερικά χαρακτηριστικά του.
3) **Testing across the clouds:** Σύμφωνα με τις προδιαγραφές, αυτοί οι έλεγχοι διεξάγονται σε διαφορετικούς τύπους cloud όπως public, private, hybrid clouds.
4) **SaaS Testing in cloud:** Ο λειτουργικός και μη λειτουργικός έλεγχος πραγματοποιείται βάση των απαιτήσεων.

Διάκριση ελέγχων εφαρμογών:

Υπάρχουν δύο βασικές κατηγορίες για τον έλεγχο των εφαρμογών στο cloud.

1) **Testing for the cloud**: Έτσι ορίζουμε την διαδικασία ελέγχου εφαρμογών που είναι προορισμένες για να τρέξουν σε cloud πλατφόρμα. Συνήθως πρόκειται για multi-threaded εφαρμογές.
2) **Testing on the cloud:** Διαχωρίζουμε τις τεχνικές ελέγχου που πραγματοποιούνται εντός των εφαρμογών ως "testing on the cloud". Σε αυτή την κατηγορία υπηρεσιών το σύστημα υπό έλεγχο εξετάζεται είτε εντός της επιχείρησης είτε στο cloud για σκοπούς ελέγχου, αλλά αναπτύσσονται σε πλατφόρμα εκτός του cloud. [6]

Σημαντικά στοιχεία κατά το cloud testing:

Κατά το cloud testing επικεντρωνόμαστε στα εξής βασικά στοιχεία:

1) **Τις εφαρμογές:** εδώ καλύπτονται οι έλεγχοι των συναρτήσεων, των δεδομένων ασφάλειας, της συμβατότητας των μηχανών αναζήτησης κλπ.

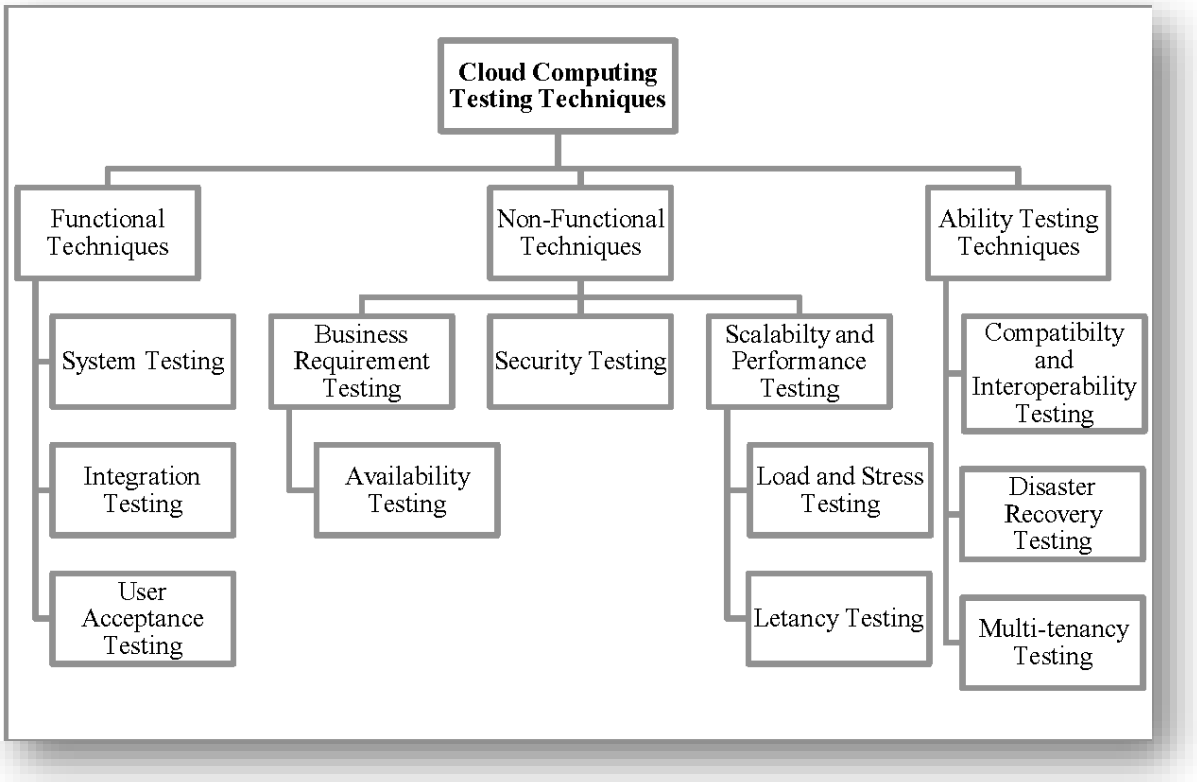
2) **Το δίκτυο:** εδώ καλύπτονται έλεγχοι που αφορούν χαρακτηριστικά όπως τα εξής: το εύρος του δικτύου, διαδικτυακά πρωτόκολλά, μεταφορές δεδομένων πάνω από το δίκτυο κλπ.
3) **Την υποδομή:** εδώ καλύπτονται έλεγχοι για ανάκαμψη από καταστροφές, πολιτικές αποθήκευσης, backups, ασφαλούς σύνδεσης κτλ. Οι υποδομές χρειάζονται να πιστοποιηθούν από κατάλληλους οργανισμούς και να συμμορφώνονται με τα κατάλληλα πρότυπα. [6]

Περιεχόμενα του cloud testing:

Παρακάτω παρατίθενται κάποιοι τύποι του cloud testing σε αντιστοιχία με τις εργασίες που εκτελούνται σε αυτούς:

1) **Έλεγχοι προσανατολισμένοι στο cloud:** Πρωταρχικός στόχος είναι να επιβεβαιωθεί η ποιότητα των παρεχόμενων υπηρεσιών και λειτουργιών που παρέχονται στο cloud ή στα Saas. Οι έλεγχοι που εκτελούνται σε αυτό το περιβάλλον είναι για παράδειγμα οι έλεγχοι ασφάλειας, λειτουργικότητας, απόδοσης, κλπ.
2) **Online εφαρμογές που ελέγχονται στο cloud**: Οι πωλητές των online εφαρμογών εκτελούν ελέγχους που ελέγχουν την επίδοση του συστήματος και λειτουργικούς ελέγχους που βασίζονται στο cloud. Όταν οι εφαρμογές συνδέονται σε "legacy systems", η ποιότητα της συνδεσιμότητας μεταξύ αυτών των συστημάτων και της εφαρμογής που ελέγχεται επικυρώνεται.
3) **Εφαρμογές για το cloud που ελέγχονται στο cloud**: Ελέγχεται η ποιότητα της εφαρμογής πάνω σε διαφορετικούς τύπους cloud. [6]

Τύποι ελέγχου:



Όπως βλέπουμε και από το σχήμα οι έλεγχοι του cloud χωρίζονται σε 3 βασικές κατηγορίες. Τον λειτουργικό έλεγχο, μη λειτουργικό έλεγχο και τον έλεγχο δυνατοτήτων. Αυτοί οι έλεγχοι χωρίζονται περεταίρω σε άλλες κατηγορίες που θα δούμε παρακάτω.

1) **Λειτουργικός έλεγχος:** Αφορά τοπικές αλλά και απομακρυσμένες εφαρμογές. Είναι η μορφή ελέγχου που γίνεται για όλα τα χαρακτηριστικά και τις συναρτήσεις του συστήματος που περιλαμβάνουν όχι μόνο το hardware αλλά και το software. Διεξάγεται σε μια ολοκληρωμένη και ενσωματωμένη πλατφόρμα ελέγχου με σκοπό να ελέγξει τη συμμόρφωσή του νέφους με τις απαιτήσεις. Σε αυτή την κατηγορία ελέγχου, η διαδικασία της επαλήθευσης πραγματοποιείται στο Cloud και όχι στα θεμέλια του software. Ο λειτουργικός έλεγχος δεν είναι "έξυπνος" αρκετά για να αναγνωρίσει όλους τους δοθέντες συνδυασμούς που θα υποβληθεί μια ιστοσελίδα και ποιες θα είναι επιδόσεις τις υπό συνθήκες έντονης συμφόρησης.

Έτσι διασφαλίζεται ότι το λογισμικό πληροί τις λειτουργικές απαιτήσεις. Ο λειτουργικός έλεγχος χωρίζεται σε τρείς υποκατηγορίες:

i. *Τον έλεγχο συστήματος*: Αυτές οι τεχνικές ελέγχου χρησιμοποιούνται για να δοκιμαστεί η συμπεριφορά του συστήματος εντός των δικών του ορίων. Είναι σημαντικό να αποδειχθεί ότι οι συναρτήσεις του συστήματος όπως έχουν σχεδιαστεί κατά τη δημιουργία του συστήματος μπορούν να λειτουργήσουν μαζί, ότι τα δεδομένα εισόδου και εξόδου είναι τα προσδοκόμενα και ότι το συνολικό αποτέλεσμα του συστήματος είναι υψηλής ποιότητας για το cloud.
ii. *Τον έλεγχο ενσωμάτωσης*: Αυτές οι τεχνικές επιτρέπουν στις επιχειρήσεις να επιβεβαιώσουν ότι η χρήση του cloud θα λειτουργήσει μέσα στις δικές τους υποδομές και τα δικά τους περιβάλλοντα και ότι η υλοποίηση του cloud δε θα επηρεάσει τα ήδη υπάρχοντα συστήματα. Επιπλέον οι απαιτήσεις των επιχειρήσεων πρέπει να επικυρώνονται και να επαληθεύονται προκειμένου να αποδειχθεί ότι το αποτέλεσμα από τη χρήση του cloud θα ικανοποιήσει τις προσδοκίες της επιχείρησης.
iii. *Τον έλεγχο αποδοχής χρηστών*: Αυτές οι τεχνικές ελέγχου πραγματοποιούνται προκειμένου να αποδειχθεί ότι η χρήση του cloud ικανοποιεί της απαιτήσεις τις επιχείρησης, έτσι ώστε οι χρήστες να αποδέχονται την ανάπτυξη των μεθόδων και των λύσεων που παρέχονται μέσω του cloud. Αυτός ο έλεγχος δεν πραγματοποιείται μόνο εντός της εταιρείας αλλά και στους απομακρυσμένους κεντρικούς διακομιστές. [3]
iv. *Τον έλεγχο μονάδας*: Χρησιμοποιείται για τον έλεγχο συγκεκριμένης μονάδας ή ομάδας από συσχετιζόμενες μονάδες. Μπορεί να περιγραφεί και ως ο έλεγχος μιας λειτουργίας, μονάδας ή ενός αντικειμένου ξεχωριστά από το υπόλοιπο πρόγραμμα.[5]
v. *Τον έλεγχο παραγωγής*: Λαμβάνει μέρος όταν το λογισμικό εγκαταθίσταται στο μηχάνημα του πελάτη για πραγματική χρήση.[5]

2) **Μη-λειτουργικός έλεγχος:** Γίνεται για να διαπιστωθεί ότι μια web εφαρμογή πληροί ορισμένα κριτήρια απόδοσης. Ο μη-λειτουργικός έλεγχος είναι γνωστός και ως έλεγχος απόδοσης. Στο cloud, το εύρος του scalability των εφαρμογών είναι πολύ πιο ευρύ σε σχέση. Με αυτούς τους ελέγχους συνεπώς διασφαλίζεται η ποιότητα των

υπηρεσιών.[4] Ο μη-λειτουργικός έλεγχος χωρίζεται σε έξι υποκατηγορίες:

i. *Τον έλεγχο επεκτασιμότητας και απόδοσης*: Είναι μια περιοχή που χρειάζεται ακόμη αρκετό έλεγχο. Οι λύσεις που προσφέρονται από το cloud computing υποτίθεται πως είναι επεκτάσιμες και σε πραγματικό χρόνο. Οι συνθήκες υπερφόρτωσης μπορούν να χρησιμοποιηθούν για να αποδειχθεί ότι η ανάπτυξη του cloud μπορεί να είναι όσο επεκτάσιμη χρειάζεται με τη βοήθεια εργαλείων software. Συνεπώς το σύστημα του Cloud μπορεί να μετρηθεί επακριβώς και να επαληθευτεί η χωρητικότητα του. Οι τεχνικές που μελετούν την απόδοση μας επιτρέπουν να μετρήσουμε την απόδοση του συστήματος με ακρίβεια. Οι έλεγχοι απόδοσης σε συνδυασμό με τις συνθήκες ελέγχου του υποκείμενου φόρτου, μας επιτρέπουν να καταλάβουμε με σαφήνεια τις ικανότητες της χρήσης του Cloud. Η απόδοση γενικά είναι πλήρως συνυφασμένη με τις δυνατότητες του συστήματος εντός της υποδομής του Cloud. Η εύρεση των ορίων και των σημείων συμφόρησής είναι και αυτό κομμάτι του ελέγχου απόδοσης. Για αυτό είναι απαραίτητος ο έλεγχος της απόδοσης κάτω από συγκεκριμένο χώρο εργασίας αλλά σε ποικίλες περιπτώσεις κίνησης του διαδικτύου σε πραγματικό χρόνο.
ii. *Τον έλεγχο που αφορά τις επιχειρησιακές απαιτήσεις*: Οι επιχειρησιακές απαιτήσεις είναι το θεμέλιο για την δημιουργία ενός cloud συστήματος. Οι απαιτήσεις αυτές μπορούν να ικανοποιηθούν μέσω συναντήσεων με τους πελάτες, reviews κ.ά.
iii. *Τον έλεγχο καθυστέρησης*: Ο έλεγχος αυτός πραγματοποιείται για να υπολογίζει το μέτρο της καθυστέρησης μεταξύ ενός αιτήματος στο σύστημα του cloud και της αντίστοιχης απάντησης του για κάθε εφαρμογή μετά την ανάπτυξη της στο περιβάλλον του cloud.
iv. *Τον έλεγχο διαθεσιμότητας του cloud*: Το cloud πρέπει να είναι συνεχώς διαθέσιμο. Πρέπει να υπάρχει η βεβαιότητα ότι δεν θα υπάρξει απότομη διακοπή λειτουργίας.
v. *Τον έλεγχο ασφάλειας*: Απαραίτητο κομμάτι του ελέγχου κάθε είδους εφαρμογής. Μπορεί να επιτευχθεί χρησιμοποιώντας τεχνικές white hacking.

vi. *Τον έλεγχο φόρτου και "stress":* Η σταθερότητα των εφαρμογών είναι ένας σημαντικός παράγοντας καθώς ο αριθμός των χρηστών συνεχώς αυξάνεται. Ο έλεγχος φόρτου περιλαμβάνει την δημιουργία βαριάς χρήσης και μέτρησης της απόκρισης. Υπάρχει επίσης η ανάγκη για αύξηση της απόδοσης των εφαρμογών. Είναι σημαντική η αναγνώριση σφαλμάτων όταν το σύστημα ελέγχεται στην μέγιστη "χωρητικότητά" του ή εκτός ορίων της προβλεπόμενης χρήσης.
Ο έλεγχος "stress" χρησιμοποιείται για να καθοριστέι η ικανότητα που έχει μια εφαρμογή να διατηρεί την αποτελεσματικότητά της όταν υπάρχει εξαιρετικά μεγάλος αριθμός χρηστών. Είναι αρκετά σημαντικό μια εφαρμογή να δουλεύει ακόμα και με μεγάλες χρηστικές απαιτήσεις διατηρώντας την σταθερότητά της.
Ο έλεγχος αυτός δυστυχώς είναι αρκετά δαπανηρός. [3]

- *Διαπίστωση*: Οι λειτουργικοί έλεγχοι απαιτούν σημαντική χρήση του hardware και του software προκειμένου να προσομοιώσουν τη δραστηριότητα του χρήστη. Αντίθετα, οι μη λειτουργικοί έλεγχοι ενεργοποιούν τα μη λειτουργικά χαρακτηριστικά του software. [5]

3) **Έλεγχος δυνατοτήτων**: Αυτός ο τύπος ελέγχων πραγματοποιείται για να βεβαιώσουμε ότι το περιβάλλον του cloud παρέχει στους χρήστες τις υπηρεσίες του σε συνθήκες πραγματικού χρόνου. Σε αυτή την κατηγορία υπάγονται οι έλεγχοι συμβατότητας, διαλειτουργικότητας, ικανότητας ενός στιγμιότυπου software και των αντίστοιχων υποδομών του, να υποστηρίζουν πολλούς πελάτες (multi-tenancy). Αυτή η κατηγορία λοιπόν δείχνει πότε οι χρήστες θα λάβουν υπηρεσίες εφαρμογών από το νέφος σε συνθήκες πραγματικού χρόνου. [4]
   i. *Έλεγχος ανάκαμψης από καταστροφές*: Κάθε πάροχος υπηρεσιών του cloud επιθυμεί οι υπηρεσίες που προσφέρονται από το Cloud να είναι διαθέσιμες συνεχώς στους τελικούς χρήστες, δυστυχώς αυτό δεν είναι εφικτό. Υπάρχει πιθανότητα να προκληθεί κάποια απώλεια όποτε ο χρόνος ανάκαμψης από την καταστροφή πρέπει να είναι χαμηλός. Η επαλήθευση του cloud πρέπει να γίνει για να επιβεβαιωθεί ότι οι υπηρεσίες που παρέχονται θα ναι πάλι διαθέσιμες online με την μικρότερη δυνατή επίπτωση στην επιχείρηση.

ii. *Τον έλεγχο συμβατότητας και διαλειτουργικότητας*: Στο περιβάλλον cloud χρησιμοποιούνται πολλά διαφορετικά λειτουργικά συστήματα και διαφορετικό λογισμικό, συνεπώς ο έλεγχος αυτός καθίσταται απαραίτητος. Μια cloud εφαρμογή πρέπει να λειτουργεί σε πολλές πλατφόρμες.

iii. *Τον έλεγχο ικανότητας ενός στιγμιότυπου software και των αντίστοιχων υποδομών του να υποστηρίζουν πολλούς πελάτες (multi-tenancy)*: Ο έλεγχος multi-tenancy διασφαλίζει ότι πολλαπλοί πελάτες και οργανισμοί χρησιμοποιούν υπηρεσίες πραγματικού χρόνου, ενεργοποιημένες σε μια συγκεκριμένη χρονική στιγμή. Οι υπηρεσίες του cloud πρέπει να μπορούν να εξατομικευτούν για κάθε πελάτη και να του παρέχουν τα δεδομένα του και ένα επίπεδο ασφάλειας που θα αποφύγει προβλήματα προσπέλασης. [3]

Κενά στους ελέγχους:

Καθώς ο έλεγχος με τη βοήθεια του cloud είναι μια σχετικά καινούργια έννοια, έχει παρατηρηθεί σε διάφορα πεδία έλλειψη στην έρευνα για συγκεκριμένες πτυχές. Μερικά ενδεικτικά παραδείγματα αναφέρονται στη συνέχεια.

1) Οι έλεγχοι διαλειτουργικότητας παρουσιάζουν ανάγκες για περαιτέρω έρευνα
2) Οι έλεγχοι αποδοχής επίσης δεν έχουν μελετηθεί διεξοδικά.
3) Οι αυτοματοποιημένοι έλεγχοι στο cloud είναι ακόμη ένα θέμα που δεν έχει μελετηθεί προσεκτικά.

Ωστόσο όλο και περισσότερες υπηρεσίες μεταφέρονται στο cloud για έλεγχο και έτσι η ερευνητική κοινότητα δίνει μεγαλύτερη έμφαση στον χώρο. [3]

Βήματα για το cloud testing (για εφαρμογές του cloud):

Οι εταιρείες προσομοιώνουν τους πραγματικούς χρήστες του διαδικτύου χρησιμοποιώντας τις cloud υπηρεσίες ελέγχου που προσφέρονται από συγκεκριμένους παρόχους όπως οι "HP", "KEYNOTE SYSTEMS" κλπ. Όταν τα σενάρια χρηστών έχουν αναπτυχθεί και έχει σχεδιαστεί και ο κατάλληλος έλεγχος, αυτοί οι πάροχοι αξιοποιούν τις υπηρεσίες του Cloud προκειμένου να παράξουν διαδικτυακή κίνηση. Όταν ολοκληρωθεί ο έλεγχος, οι πάροχοι των υπηρεσιών νέφους μεταφέρουν τα αποτελέσματα στους ειδικούς, για να ολοκληρωθεί η ανάλυση σχετικά με τον τρόπο που οι εφαρμογές τους και το διαδίκτυο θα λειτουργήσουν σε μεγάλες συμφορήσεις. [12]

Εργαλεία για έλεγχο cloud (για τις εφαρμογές του clouds):

Μερικοί από τους πιο γνωστούς παρόχους για έλεγχο νέφους είναι οι εξής: [1]

1. Amazon
2. Advaltis
3. 3-terra
4. Microsoft
5. Skytap
6. HP
7. Soasta
8. Intel
9. Yahoo

Κάποιοι από τους παραπάνω παρόχους συνεργάζονται προκειμένου να δημιουργήσουν τα λεγόμενα "test beds", τα οποία αποτελούνται από εκατοντάδες επεξεργαστές που δουλεύουν μαζί για την υλοποίηση ενός υπολογιστικού συστήματος. Αυτά τα "test beds" επιτρέπουν στους χρήστες να δοκιμάζουν τις εφαρμογές cloud σε κλίμακα διαδικτύου και επίσης να κατανοήσουν πώς συμπεριφέρονται τα συστήματα και το λογισμικό τους

στο cloud. Αυτά τα εργαλεία είναι ιδανικά για μη λειτουργικούς και για αυτοματοποιημένους ελέγχους. Μάλιστα κάποια από αυτά τα εργαλεία μπορούν να χρησιμοποιηθούν στο cloud προκειμένου να πραγματοποιηθούν αυτοματοποιημένοι έλεγχοι όσον αφορά το "regression test". [5]

Κάποια από τα εργαλεία χρησιμοποιούνται σε διαφορετικά είδη ελέγχων που πραγματοποιούνται στο νέφος. Πολλά από τα εργαλεία χρησιμοποιούνται για έλεγχο απόδοσης, φόρτου και στρες. Τα επόμενα εργαλεία χρησιμοποιούνται κυρίως λειτουργικούς ελέγχους: [2]

1. SOASTA CloudTest
2. LoadStorm
3. CloudTestGo
4. AppPerfect.
5. Jmeter
6. Cloudslueth
7. CloudTestGo
8. AppPerfect

Επίσης τα πιο συχνά χρησιμοποιούμενα εργαλεία ελέγχου που σχετίζονται με την ασφάλεια, είναι τα εξής:

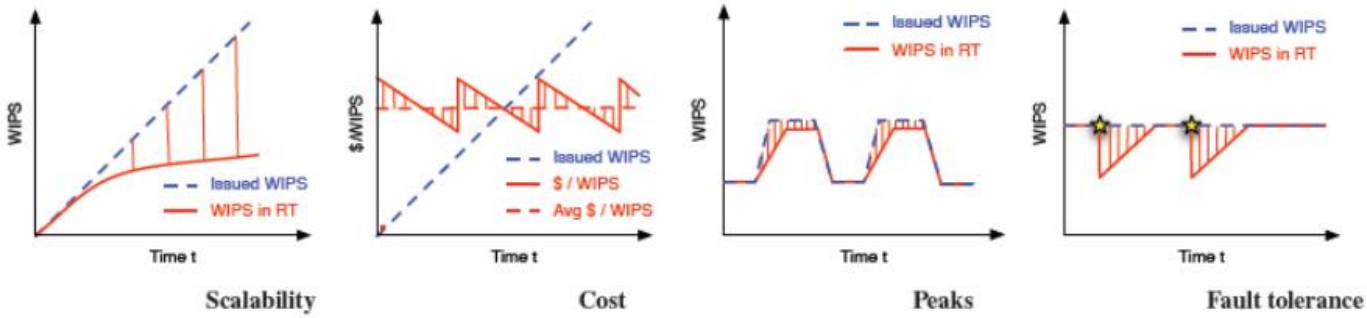
1. Nessus
2. Wireshark
3. Nmap

Μετρικές για την αξιολόγηση του cloud:

Οι ακόλουθες μετρικές χρησιμοποιούται για την αξιολόγηση της αποθηκευτικής ικανότητας του cloud[8]:

1. **Scalability:** Οι υπηρεσίες του νέφους αναμένεται να κλιμακωθούν γραμμικά έχοντας όμως σταθερό κόστος. Γίνεται μέτρηση των αποκλίσεων του του χρόνου απόκρισης στην τέλεια γραμμική κλίμακα χρησιμοποιώντας τον συντελεστή συσχέτισης R^2 ή ή καθορίζοντας τις παραμέτρους μιας συνάρτησης της μορφής $f(x) = x^b$.

2. **Cost:** Η οικονομία του νέφους μετριέται από τον τύπο $/WIPS όπου WIPS = web interaction per second. Επιπλέον, μετρά την τυπική απόκλιση του κόστους κατά την κλιμάκωση,
3. **Peaks:** Η μετρική δείχνει πόσο καλά ένα σύννεφο μπορεί να προσαρμοστεί στο άνω όριο φόρτου τόσο στην κλιμάκωση όσο και στην αποκλιμάκωση. Η προσαρμοστικότητα ορίζεται ως η αναλογία του WIPS με το RT(real-time).
4. **Fault tolerance:** Η υποδομή του νέφους στηρίζεται σε έναν μεγάλο αριθμό από commodity hardware. Οι αποτυχίες του υλικού είναι αρκετά συχνές στην υποδομή. Έτσι, η μετρική μετράει τις ικανότητες ανεκτικότητας του νέφους στις αποτυχίες. Δεδομένων κάποιων αποτυχιών σε συγκεκριμένη χρονική περίοδο, η δυνατότητα ανάκαμψης ορίζεται επίσης ως την αναλογία του WIPS με το RT(real-time).



Οφέλη από τον έλεγχο του Cloud (για τις εφαρμογές του cloud):

1. Το χαμηλό κόστος και η προσβασιμότητα των υπέρογκων υπολογιστικών πόρων του cloud παρέχουν την δυνατότητα να αντιγραφεί η πραγματική χρήση των συστημάτων από γεωγραφικά διαμοιρασμένους χρήστες, που εκτελούν ποικίλα σενάρια χρήσης και σε κλίμακα που δεν είναι εφικτή σε ένα παραδοσιακό περιβάλλον ελέγχου. [1] Πιο συγκεκριμένα τα κόστη περιορίζονται αρκετά. Καθώς στο cloud computing πληρώνει κανείς μόνο για τους πόρους που χρησιμοποιεί, έτσι δε χρειάζεται να επενδύσουν σε ακριβό εξοπλισμό και να σπαταλήσουν χρήματα συντηρώντας τον και αναβαθμίζοντας τον. Μπορεί λοιπόν κανείς να χρησιμοποιήσει το περιβάλλον ελέγχου

που χρειάζεται αυτή τη στιγμή και να πληρώσει μόνο για αυτό το περιβάλλον. [4]

2. Μερικά από τα πιο σημαντικά πλεονεκτήματα παρατάσσονται στη συνέχεια. Το cloud είναι πως αποτελεί ένα περιβάλλον ελέγχου δυναμικά διαθέσιμο, σε αντίθεση με τους συνηθισμένους ελέγχους που γίνονται εντός των οργανισμών. Το νέφος είναι εύκολο να αναπαράγει το περιβάλλον του χρήστη και να βρίσκει τα προβλήματα εγκαίρως.
3. Επιπλέον είναι εύκολα εξατομικεύσιμο, καθώς οι επιχειρήσεις μπορούν εύκολα να προσομοιώσουν ένα περιβάλλον προσανατολισμένο στους τελικούς χρήστες προσαρμόζοντας το ανά χρήση. Οι ομάδες ελέγχων μπορούν λοιπόν εύκολα να δημιουργήσουν διάφορους συνδυασμούς σεναρίων ελέγχου απόδοσης και φόρτου.
4. Τέλος είναι εύκολα επεκτάσιμο, αυτό συμφέρει πολύ στις περιπτώσεις που οι επιχειρήσεις αλλάζουν τις απαιτήσεις τους με μεγάλη συχνότητα. [2]
5. Τα πλεονεκτήματα όμως του cloud testing είναι αρκετά ακόμη. Για παράδειγμα σώζει σημαντικό χρόνο, καθώς με τη χρήση ελέγχων που βασίζονται στο νέφος η εφαρμογή μπορεί στιγμιαία να τρέξει σε διαφορετικά hardware και έτσι οι testers μπορούν να αφιερώσουν περισσότερο χρόνο στη διόρθωση λαθών.
6. Είναι ένα εργαλείο που είναι κατάλληλα διαμορφωμένο. Συνήθως χρειάζεται αρκετός χρόνος και κόπος προκειμένου να στηθεί το περιβάλλον ελέγχου σε πολλές συσκευές. Χρειάζεται επίσης αρκετός χρόνος για να ανακάμψουμε από λάθη που μπορεί να εμφανίζονται σε όλες τις συσκευές. Με τη βοήθεια όμως του cloud testing μπορεί να αποφευχθεί όλος αυτός ο κόπος, διότι υπάρχουν εργαλεία που έχουν προσχεδιαστεί από τους παρόχους με αυτόν τον σκοπό σώζοντας χρήμα και χρόνο.
7. Το cloud testing εξασφαλίζει ολοκληρωμένες δοκιμές. Κανονικά για να διεξαχθούν ολοκληρωμένοι έλεγχοι χρειάζονται ομάδες ελέγχου οι οποίες θα τρέξουν την εφαρμογή σε όλες τις πιθανές συσκευές που υποστηρίζουν διαφορετικές πλατφόρμες, λειτουργικά συστήματα και προγράμματα περιήγησης. Όμως οι έλεγχοι που βασίζονται στο cloud παρέχουν όλα αυτά και περιορίζουν την ανάγκη να αγοραστούν.
8. Επιπλέον το cloud testing επιταχύνει τους ελέγχους, γιατί τα εργαλεία ελέγχου στο νέφος εξασφαλίζουν αυτοματοποιημένους ελέγχους.

9. Οι έλεγχοι στο νέφος χαρακτηρίζονται ως διαρκώς διαθέσιμοι. Αυτό σημαίνει πως οι έλεγχοι στο νέφος είναι διαρκώς διαθέσιμοι στους ελεγκτές κάθε στιγμή. Οι έλεγχοι μπορούν να πραγματοποιηθούν οπουδήποτε και οποιαδήποτε στιγμή, με αποτέλεσμα οι ελεγκτές να μπορούν να επιταχύνουν την ανάπτυξη λογισμικού και τους ελέγχους.
10. Το cloud testing είναι ιδανικό για να επιτυγχάνεται καλύτερη συνεργασία μεταξύ των προγραμματιστών και τον ελεγκτών. [4]

Μειονεκτήματα του cloud testing (για τις εφαρμογές του cloud):

1. Το αρχικό κόστος για τη μεταφορά των ελέγχων στο cloud είναι πολύ υψηλό καθώς εμπεριέχει τροποποιήσεις κάποιων από τις περιπτώσεις χρήσης για να ταιριάζουν στο περιβάλλον του cloud. Για τον λόγο αυτόν, δεν είναι απαραίτητα η καλύτερη λύση να ελέγχουμε στο cloud όλα τα προβλήματα. Τα συστήματα και οι υπηρεσίες πρέπει λοιπόν να τροποποιηθούν για να ελεγχθούν στο cloud. Η χρήση ειδικών διεπαφών ίσως αποτελεί λύση για το προαναφερθέν πρόβλημα.
2. Ένα ακόμη πρόβλημα είναι πως όπως και οι άλλες υπηρεσίες του νέφους, ο έλεγχος του cloud μπορεί να αντιμετωπίζει θέματα ασφάλειας. Η ασφάλεια στο νέφος δημιουργεί πολλούς προβληματισμούς, καθώς οι τεχνικές κρυπτογράφησης έχουν διάφορα θέματα. Παρόλο που μέσω του cloud οι πάροχοι μπορούν κάνουν ελέγχους, ειδικά στα private clouds, υπάρχουν ακόμη αμφιβολίες σχετικά με την ασφάλεια των δεδομένων τα οποία είναι αποθηκευμένα σε απομακρυσμένες τοποθεσίες μακριά από τη δικαιοδοσία της εταιρείας. Εφόσον όλα είναι διαθέσιμα σε πραγματικό χρόνο και μάλιστα σε κάθε χρήστη, η ασφάλεια είναι ένα πρωταρχικό ζήτημα για κάθε οργανισμό καθώς ακόμη πραγματοποιείται αρκετή έρευνα και συζήτηση σχετικά με τα πρότυπα ασφάλειας. Έτσι η προστασία προσωπικών δικαιωμάτων των χρηστών, τα πρωτόκολλα ασφάλειας του νέφους και η ασφάλεια των εφαρμογών που τρέχουν στο cloud είναι μερικά από τα πιο σοβαρά θέματα που πρέπει να διευθετηθούν.
3. Επιπλέον, τα αποτελέσματα των ελέγχων ενδεχομένως να μην είναι ακριβή εξαιτίας της απόδοσης των παρόχων των υπηρεσιών του δικτύου και του διαδικτύου. [1]

4. Ένα ακόμη πρόβλημα είναι η επίδοση των εφαρμογών στο cloud και ιδιαίτερα στα private clouds. Επειδή αυτή η εφαρμογή θα διαμοιραστεί ανάμεσα σε πολλούς χρήστες είναι λογικό να δημιουργεί καθυστερήσεις. Εάν μάλιστα πραγματοποιούνται συντηρήσεις και εξωγενείς δραστηριότητες, το εύρος του cloud ίσως κριθεί ανεπαρκές.
5. Για τους σκοπούς του ελέγχου, χρειάζονται συγκεκριμένες προδιαγραφές στους αντίστοιχους διακομιστές, ο αποθηκευτικός χώρος και το δίκτυο ίσως να μην υποστηρίζονται από τον συγκεκριμένο πάροχο του νέφους. Αυτό όμως καμία φορά δεν είναι εύκολο να προσομοιωθεί στο περιβάλλον του πελάτη.
6. Ένα ακόμη πρόβλημα είναι πως λείπουν πρότυπα. Δηλαδή, δεν υπάρχουν ακόμη διεθνή πρότυπα που να ορίζουν πως οι ελεγκτές μπορούν να ενσωματώνουν εσωτερικούς πόρους των εταιρειών τους με τους δημόσιους πόρους του νέφους. Εφόσον οι πάροχοι public cloud αναπτύσσουν δικές τους αρχιτεκτονικές και μοντέλα, αυτές οι υπηρεσίες του νέφους έχουν μικρή ικανότητα διαλειτουργικότητας. Έτσι, οι ελεγκτές μπορεί να αντιμετωπίζουν δυσκολίες εάν θέλουν να αλλάξουν πάροχο, καθώς όλοι οι πάροχοι δεν προσφέρουν τις ίδιες υπηρεσίες.[4]
7. Επιπλέον υπάρχουν διάφορα προβλήματα διαθεσιμότητας, παρόλο που οι πάροχοι εγγυόνται συνεχή διαθεσιμότητα των υπηρεσιών τους ακόμη και ο παραμικρός χρόνος που μπορεί να πέσει το cloud μπορεί να δημιουργήσει τρομερές συνέπειες στη διαδικασία ελέγχου.
8. Υπάρχουν επίσης προβλήματα υποδομής. Είναι δύσκολο να προσομοιωθεί το περιβάλλον του πελάτη εάν διαπιστωθεί πως κάποιες ρυθμίσεις δεν υποστηρίζονται από τον πάροχο. Επιπλέον, η δημιουργία περιβάλλοντος ελέγχου που συμπεριλαμβάνει όλες τις απαραίτητες ρυθμίσεις και τα αναγκαία δεδομένα μπορεί να είναι αρκετά αργό για τους ελεγκτές.
9. Επιπλέον αναδύονται κρυφά κόστη. Παρόλο που οι πωλητές ενημερώνουν τους καταναλωτές για τις τιμές των υπηρεσιών τους, η ακατάλληλη χρήση των περιβαλλόντων ελέγχου μπορεί να αυξήσουν κατακόρυφα τα κόστη. Προκειμένου να αποφευχθούν τα κρυφά κόστη, οι ελεγκτές πρέπει να οργανώσουν προσεκτικά το περιβάλλον ελέγχου τους, λαμβάνοντας υπόψιν τους επιπλέον κόστη εξαρχής.
10. Προβλήματα δημιουργούνται στο integration testing όπου ελέγχονται δίκτυα, βάσεις δεδομένων κ.ά. γιατί ο ελεγκτής δεν θα έχει πρόσβαση στο περιβάλλον αυτό. Το πρόβλημα μεγιστοποιείται όταν αυτά τα

στοιχεία πρέπει να αλληλοεπιδράσουν διότι θα υπάρχει ο κίνδυνος για server crashes κ.ά. [2]

11. Υπάρχουν προβλήματα σύντομης ειδοποίησης, δηλαδή ο πάροχος του cloud στέλνει μια σύντομη ειδοποίηση -διαστήματος μια με δύο εβδομάδες- στους πελάτες σχετικά με την ύπαρξη αναβαθμίσεων. Αυτό μπορεί να είναι μεγάλο θέμα όταν επικυρώνονται χειρωνακτικά οι αλλαγές σε μια SaaS εφαρμογή
12. Προβλήματα εμφανίζονται και κατά την επικύρωση της συμβατότητας της διεπαφής, με την αναβάθμιση μιας υπηρεσίας στο cloud κάποιες φορές η εξωτερική διεπαφή αναβαθμίζεται και αυτή με αποτέλεσμα να δημιουργεί προβλήματα σε κάποιους χρήστες που χρησιμοποιούν παλιότερες διεπαφές. Για αυτό οι συνδρομητές χρειάζεται να εξασφαλίσουν ότι οι χρήστες μπορούν να επιλέξουν όποια έκδοση θέλουν να δουλέψουν.
13. Η μεταφορά δεδομένων από έναν πάροχο σε έναν άλλο αποτελεί τεράστια πρόκληση, καθώς οι δύο πάροχοι ενδέχεται να έχουν διαφορετικά σχήματα βάσης δεδομένων και απαιτεί πολλή προσπάθεια για να κατανοήσουν τα πεδία δεδομένων, τις σχέσεις και πώς χαρτογραφούνται σε όλη την εφαρμογή SaaS.
14. Η ενσωμάτωση εταιρικών εφαρμογών απαιτεί επικύρωση δεδομένων τόσο των εξερχόμενων όσο και των εισερχόμενων δεδομένων, από το δίκτυο πελατών έως την εφαρμογή SaaS και αντίστροφα. Το απόρρητο των δεδομένων απαιτεί μια ολοκληρωμένη επικύρωση, προκειμένου να διασφαλιστούν οι συνδρομητές σχετικά με την ασφάλεια και το απόρρητο των δεδομένων.
15. Η μεγαλύτερη πρόκληση του Cloud testing είναι να διασφαλιστεί ότι οι αναβαθμίσεις δεν επηρεάζουν τους υπάρχοντες συνδεδεμένους χρήστες.

Συμβουλές για ορθό cloud testing:

Ο έλεγχος στο cloud μπορεί να είναι πιο αποδοτικός για τις επιχειρήσεις συγκριτικά με τον έλεγχο εντός του περιβάλλοντος της εταιρείας. Κάποιες συμβουλές για την αξιοποίηση των ελέγχων του νέφους παρατάσσονται στη συνέχεια: [4]

1) Ο ελεγκτής πρέπει να έχει πλήρη κατανόηση των αναγκών τις εταιρείας. Ο έλεγχος στο νέφος, απαιτεί μια στενή συνεργασία

ανάμεσα σε ελεγκτές και προγραμματιστές προκειμένου να μπορούν να διεξαχθούν όλοι οι απαραίτητοι έλεγχοι προσεκτικά και καθόλη τη διάρκεια ζωής του λογισμικού.

2) Απαιτείται καθορισμός της στρατηγικής που θα ακολουθηθεί. Πρώτού μεταφέρουμε το project στο νέφος, χρειάζεται να αποφασίσουμε ποια είδη ελέγχων μας ενδιαφέρει να διεξάγουμε, πόσο χρόνο θέλουμε να αφιερώσουμε στους ελέγχους και τι ρίσκα θα πάρουμε. Έτσι θα υπολογίσουμε καλύτερα το κόστος του ελέγχου και θα αποφύγουμε επιπλέον χρεώσεις.
3) Χρειάζεται κατανόηση της υποδομής. Είναι απαραίτητο να μπορεί ο ελεγκτής να εντοπίσει τις απαιτήσεις του περιβάλλοντος ελέγχου. Πρέπει να διασφαλίσει ότι οι υπηρεσίες που βασίζονται στο cloud παρέχουν τα απαραίτητα εργαλεία, λογισμικό, υλικό και εύρος. Επίσης χρειάζεται να καθοριστεί για πόσο διάστημα θα χρησιμοποιηθεί τον περιβάλλον ελέγχου του νέφους.
4) Απαιτείται η σωστή επιλογή παρόχου. Όταν ψάχνουμε για τεχνικές ελέγχου cloud, μας ενδιαφέρει να επιλέξουμε έναν πάροχο ο οποίος να παρέχει ασφάλεια, ποιότητα και αξιοπιστία υπηρεσιών.
5) Καλό θα ήταν να γίνεται ανάλυση των αποτελεσμάτων σε πραγματικό χρόνο, καθώς έτσι θα μπορεί να γίνει έγκαιρη επέμβαση σε θέματα που αφορούν θέματα απόδοσης.
6) Μπορεί να γίνει δοκιμή κάποιων free trials τα οποία παρέχονται από τους παρόχους και έτσι οι ελεγκτές μπορούν να πειραματιστούν με τα εργαλεία που προσφέρει ο εκάστοτε πάροχος και να μάθουν τα πλεονεκτήματα και τα μειονεκτήματα κάθε υπηρεσίας που παρέχεται.
7) Η πρόσβαση στην υπηρεσία του cloud πρέπει να διανεμηθεί σε αρκετούς ανθρώπους ώστε να αποφευχθεί το ρίσκο λαθών(που υπάρχει στην περίπτωση ενός ελεγκτή).

Σύγκριση ελέγχων στο cloud με τους συμβατικούς ελέγχους:

Παράμετροι ελέγχου	Συμβατικοί έλεγχοι	Cloud Testing
Κύριο αντικείμενο του ελέγχου	• Έλεγχος διαλειτουργικότητας, συμβατότητας, χρηστικότητας. • Επαλήθευση της ποιότητας της, λειτουργίας και της απόδοσης του συστήματος με βάση τις συγκεκριμένες προδιαγραφές	Επαλήθευση της ποιότητας της απόδοσης και των λειτουργιών των SaaS, clouds και εφαρμογών, αξιοποιώντας ένα περιβάλλον cloud
Έλεγχοι περιβάλλοντος	Χρήση ενός προκαθορισμένου και διαμορφωμένου περιβάλλοντος δοκιμών σε ένα εργαστήριο δοκιμών	Χρήση ενός ανοιχτού και δημόσιου περιβάλλοντος ελέγχου με ποικιλία υπολογιστικών πόρων
Έλεγχοι ενσωμάτωσης	Οι έλεγχοι βασίζονται στα συστατικά, την αρχιτεκτονική και τις διάφορες λειτουργίες	Οι έλεγχοι βασίζονται στην τεχνολογία του SaaS
Έλεγχοι ασφάλειας	Τα χαρακτηριστικά του ελέγχου ασφάλειας βασίζονται σε διαδικασίες, σε διακομιστές και γίνονται με βάση το απόρρητο.	Τα χαρακτηριστικά του ελέγχου βασίζονται στο νέφος, την τεχνολογία του SaaS και σε ελέγχους πραγματικού χρόνου
Έλεγχοι απόδοσης και επεκτασιμότητας	Πραγματοποιούνται σε προδιαγεγραμμένο περιβάλλον	Μπορούν να εφαρμοστούν είτε σε συνθήκες

		πραγματικού χρόνου είτε σε εικονικούς online ελέγχους δεδομένων.
Κόστος δοκιμών	Το κόστος παραμένει υψηλό λόγω των απαιτήσεων υλικού και λογισμικού	Πρέπει να πληρωθούν μόνο για λειτουργικές χρεώσεις. Πληρώστε μόνο ό, τι χρησιμοποιείτε.
Δοκιμή προσομοίωσης	• Προσομοιωμένα δεδομένα κίνησης στο διαδίκτυο • Προσομοιωμένη πρόσβαση χρηστών στο διαδίκτυο	• Προσομοίωση διαδικτυακών δεδομένων κίνησης • Προσομοίωση διαδικτυακής πρόσβασης χρηστών
Λειτουργικός έλεγχος	Επικύρωση λειτουργιών (μονάδα και σύστημα) καθώς και τα χαρακτηριστικά του	Έλεγχος λειτουργίας εφαρμογής από άκρο σε άκρο σε SaaS ή Cloud

[6]

Ερωτήματα που μας βοηθούν τα προσδιορίσουμε τις κατάλληλες πολιτικές για το cloud testing:

Παρακάτω παρατίθενται τα ερωτήματα που μας βοηθούν να επιλέξουμε κατάλληλες πολιτικές για τους ελέγχους μας: [7]

- Τι κριτήριο μέτρησης να χρησιμοποιήσουμε για να αξιολογήσουμε την ποιότητα του συστήματος;
- Ποια είναι τα επαρκή "inputs" ελέγχου που πρέπει να παραχθούν για να αποκτήσουμε την ζητούμενη κάλυψη;
- Με ποια κριτήρια θα κριθεί εάν το αποτέλεσμα του ελέγχου είναι αποδεκτό ή όχι;
- Τι προβλήματα δημιουργούνται κατά τους ελέγχους της ασφάλειας υπηρεσιών; Με ποια κριτήρια θα περιοριστούν;

- Με βάση τα αποτελέσματα, πώς θα αντιληφθούν οι προγραμματιστές από πού πηγάζει το πρόβλημα;
- Πώς θα πρέπει να διαφέρει η δομή των cloud υπηρεσιών ασφάλειας από ένα παραδοσιακό σύστημα δικτύου;
- Ποιες μπορεί να είναι οι πολιτικές ασφάλειας των υπηρεσιών;
- Πώς αυτές οι πολιτικές ασφαλείας θα μοντελοποιηθούν για να λάβουν αποτελέσματα στη βάση των προδιαγραφών δοκιμές;
- Πώς οι ομάδες δοκιμών ελέγχουν διεξοδικά μια υπηρεσία έναντι των πολιτικών ασφαλείας της;
- Ποιο είναι το κριτήριο πλήρους κάλυψης χρηστικότητας;
- Η επαρκής κάλυψη θα ταιριάζει περισσότερο από την εκτεταμένη κάλυψη χρηστικότητας υπηρεσιών;
- Ποιο είναι το κριτήριο επάρκειας;

Αναπάντητα ερευνητικά ερωτήματα σχετικά με το cloud testing:

Κάποια ερωτήματα που έχουν απασχολήσει την ερευνητική κοινότητα και δεν έχει βρεθεί ακόμη μια σαφής απάντηση είναι τα εξής:

- Ποια είναι η ταξινόμηση μεταξύ εσωτερικών και εξωτερικών χρηστών σε συγκεκριμένες ομάδες, ποιο μπορεί να είναι το επίπεδο εμπιστοσύνης;
- Ποια μπορεί να είναι η κλίμακα μέτρησης για την παρακολούθηση της αξιοπιστίας των ομάδων χρηστών;
- Ποιο είναι το κριτήριο για να διαμορφωθούν οι υπηρεσίες που σχετίζονται με τον χρήστη για την βοήθεια της ολοκλήρωσης του ελέγχου, που όμως είναι ασφαλή και εξασφαλίζουν την απόκρυψη των δεδομένων των χρηστών;
- Ποια κομμάτια κώδικα και υπηρεσιών πρέπει να διαμοιραστούν στις ομάδες ελέγχων;
- Πώς ορίζεται ένας «χρήστης» σε μια διαδικασία δοκιμής;
- Ποια είναι τα χαρακτηριστικά του χρήστη και πώς μπορούμε να αντιστοιχίσουμε αυτά τα χαρακτηριστικά στη διαδικασία δοκιμής;
- Ποιες πληροφορίες σχετικά με τα στοιχεία υπηρεσίας πρέπει να παρέχονται στους χρήστες εσωτερικού ελέγχου και τι πρέπει να κρύβεται από αυτούς;

- Ποιες πληροφορίες σχετικά με τα στοιχεία υπηρεσίας πρέπει να παρέχονται στους εξωτερικούς χρήστες δοκιμών και τι πρέπει να κρύβεται από αυτούς;
- Με την εξασφάλιση της προστασίας των κρυφών πληροφοριών, πώς να παρέχονται επαρκείς πληροφορίες ώστε να διευκολύνονται οι δοκιμές;

Συμπεράσματα:

Το cloud είναι ένα εργαλείο το οποίο γίνεται όλο και πιο διαδεδομένο με την πάροδο του χρόνου. Σε αυτό οφείλονται φυσικά οι ιδιότητες του(όπως η ευελιξία του, η επεκτασιμότητα του και τα μειωμένα κόστη). Η χρήση του νέφους για ελέγχους βοηθάει τους οργανισμούς ώστε να αποκτήσουν τα απαιτούμενα εργαλεία, τις άδειες λογισμικού και τις κατάλληλες υποδομές με πολύ μικρό κόστος. Παράλληλα, οι έλεγχοι που βασίζονται στο νέφος επιτρέπει στους οργανισμούς να μειώσουν σημαντικά τα κόστη τους και το χρόνο που απαιτείται για αυτούς τους ελέγχους. Από την άλλη μεριά όμως εμπεριέχει και διάφορα ρίσκα και μειονεκτήματα που πρέπει να ληφθούν υπόψιν πρώτού αναπτυχθεί το λογισμικό στο cloud. Ένα βασικό χαρακτηριστικό των ελέγχων είναι πως είναι μια περιοδική δραστηριότητα και δημιουργούνται νέες απαιτήσεις και περιορισμοί για κάθε νέο έργο.

Bibliography

[1] https://el.wikipedia.org/wiki/%CE%A5%CF%80%CE%BF%CE%BB%CE%BF%CE%B3%CE%B9%CF%83%CF%84%CE%B9%CE%BA%CF%8C_%CE%BD%CE%AD%CF%86%CE%BF%CF%82

[2] https://www.softwaretestinghelp.com/getting-started-with-cloud-testing/

[3] Testing Techniques and its Challenges in a Cloud Computing Environment, by Dr. Rahul Malhotra & Prince Jain

[4] https://www.apriorit.com/dev-blog/548-cloud-based-testing

[5] Survey on software testing techniques in cloud computing, by V.Priyadharshini, Dr. A. Malathi

[6] https://www.guru99.com/cloud-testing-tutorial-with-saas-testing-primer.html

[7] Cloud Services Testing: An Understanding, by Atif Farid Mohammada, Hamid Mcheickb (The 2nd International Conference on Ambient Systems, Networks and Technologies)

[8] Cloud Testing Tools, Xiaoying Bai, Muyang Li, Bin Chen, Wei-Tek Tsai, Jerry Gao

[9] https://centretechnologies.com/5-cloud-computing-disadvantages/

[10] https://www.webopedia.com/TERM/M/MTA.html

[11]https://www.researchgate.net/figure/Bai-et-als-test-broker-architecture-7_fig5_228920258

[12] https://en.wikipedia.org/wiki/Cloud_testing#Steps