

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΡΗΤΗΣ

ΤΜΗΜΑ ΕΠΙΣΤΗΜΗΣ ΥΠΟΛΟΓΙΣΤΩΝ

ΠΑΡΟΥΣΙΑΣΗ / ΕΞΕΤΑΣΗ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Διάκος Ηρακλής

Μεταπτυχιακός Φοιτητής

Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης

Επόπτης Μεταπτ. Εργασίας: Καθηγητής Δ. Πλεξουσάκης

Πέμπτη, 30/6/2016, 11:00

Αίθουσα B108, Τμήμα Επιστήμης Υπολογιστών, Πανεπιστήμιο Κρήτης

" Σχεδιασμός και Υλοποίηση ενός Εποπτικού Συστήματος Εκτέλεσης Υπηρεσιών στο Νέφος"

ΠΕΡΙΛΗΨΗ

Στην εποχή του Big Data και του Cloud computing, με την εκθετική αύξηση της ροής της πληροφορίας, υπάρχει συνεχής απαίτηση από τους πελάτες των επιχειρήσεων να ικανοποιηθεί η ακόρεστη ανάγκη τους για κατανάλωση δεδομένων. Ωστόσο, οι μικρομεσαίες και οι μεγάλες επιχειρήσεις βρίσκονται αντιμέτωπες με ένα συνεχές δίλημμα στο πως να χτίσουν, διατηρήσουν, εποπτεύσουν και προσαρμόσουν μια υποδομή που να ανταποκρίνεται στις ανάγκες της επιχείρησης, μειώνοντας τις κεφαλαιουχικές δαπάνες (CAPEX) και τα λειτουργικά κόστη συντήρησης (OPEX) και εξυπηρετώντας τους πελάτες τους. Επίσης, όλο και περισσότερες επιχειρήσεις υιοθετούν την τεχνολογία νέφους με συνέπεια οι πελάτες τους να απαιτούν όχι μόνο οι συμφωνίες χρήσης σε επίπεδο υπηρεσίας (SLAs) να πληρούνται εις το ακέραιο αλλά και να έχουν τα μέσα για να επιβεβαιώσουν ότι αυτό συμβαίνει ως έχει. Για να το αντιμετωπίσουμε αυτό, ουσιαστικά, έχουμε υιοθετήσει το OpenStack, μια πλατφόρμα ανοικτού κώδικα ώστε να τεθούν τα θεμέλια της ανέγερσης μιας υποδομής cloud computing σε φυσικό επίπεδο και επίπεδο υλικού. Αρχικά, έχουμε αυτοματοποιήσει την όλη διαδικασία προκειμένου να παράγει μία βασική αλλά πλήρως λειτουργική και κλιμακώσιμη εικονοποιημένη υποδομή, καθώς με αυτόν τον τρόπο ελαχιστοποιούμε την προσπάθεια αναδόμησης ενός κέντρου δεδομένων, έτοιμο για την παραγωγή, από το σημείο μηδέν. Αυτό περιλαμβάνει τις υπηρεσίες ασφάλειας, τις υπολογιστικές υπηρεσίες, την

υπηρεσία δικτύωσης και τις υπηρεσίες εφήμερης/μόνιμης αποθήκευσης. Χρησιμοποιούμε το εργαλείο Packer για τη διαδικασία τυποποίησης και προτυποποίησης των εικονικών μηχανημάτων, το πρόγραμμα Vagrant για τη διαδικασία διοχέτευσης των εικονικών μηχανημάτων ενώ τα INI αρχεία εκκινούν τις διαδικασίες ρύθμισης των εικονικών μηχανημάτων. Έπειτα, αναδεικνύουμε τις προκλήσεις και τις λύσεις για τη ρύθμιση των υπηρεσιών συγχρονισμού χρόνου (NTP, Chrony), διότι είναι ζωτικής σημασίας για την ορθή λειτουργία του συνόλου της υποδομής μας. Επιπροσθέτως, προσπαθούμε να αυτοματοποιήσουμε την διαδικασία συλλογής των βασικών πληροφοριών των εικονικών μηχανημάτων μας μέσω ενός *nix δαίμονα ή μιας Windows υπηρεσίας. Αναδεικνύουμε τις εσωτερικές λειτουργίες και τις λεπτές λεπτομέρειες της συγγραφής μιας βιβλιοθήκης ανοιχτού κώδικα, η οποία ονομάζεται CRAPI (Common Range API) και μας βοήθησε στη συγγραφή κώδικα συμβατού με διάφορα λειτουργικά συστήματα με σκοπό τη δημιουργία και την εγκατάσταση Windows υπηρεσιών ή *nix δαιμόνων. Για τους δικούς μας σκοπούς, η παραγόμενη υπηρεσία λειτουργικού συστήματος περιέχει την βασική λογική που χειρίζεται τη διατήρηση και ανανέωση σημαντικών πληροφοριών σχετικά με τα εικονικά μας μηχανήματα. Επιπλέον, καταφεύγουμε στο Gnocci και το εγκαθιστούμε σε επίπεδο οικοδεσπότη (host), προκειμένου να μπορούμε να εποπτεύουμε τα εικονικά μας μηχανήματα και να παρακολουθούμε στιγμιότυπα τα οποία τρέχουν σε εικονικά μας μηχανήματα υπολογιστικής φύσεως. Το Gnocci επιτελεί ένα διττό ρόλο: από τη μία πλευρά, χρησιμοποιείται ως μία παθητική λύση παρακολούθησης όπως η συλλογή βασικών δεδομένων κατανάλωσης και χρήσης πόρων του συστήματος από την CPU / RAM / HDD / SSD / NIC. Από την άλλη, μας επιτρέπει να συγκεντρώσουμε επεξεργασμένα δεδομένα των εικονικών μας μηχανημάτων για λόγους τιμολόγησης και βραχυπρόθεσμης αρχειοθέτησης (WIP – εργασία σε εξέλιξη). Ως εκ τούτου εισάγουμε την έννοια των ενεργητικών καθυκόντων επόπτευσης συστημάτων. Για παράδειγμα, με τη χορήγηση μιας δευτερογενούς vNIC για την εξισορρόπηση της κίνησης του δικτύου ή την προμήθεια επιπρόσθετης vCPU ώστε να βοηθήσουμε ένα πρόγραμμα στις εργασίες του που εξαρτώνται αποκλειστικά από τον επεξεργαστή. Αυτοί οι μηχανισμοί μάς επιτρέπουν να αποφευχθεί η πιθανή παραβίαση ενός ή περισσότερων ρητρών των στόχων επιπέδου υπηρεσιών (SLOs) μιας SLA που έχει συναφθεί μεταξύ ενός διαχειριστή υπηρεσιών νέφους και της πελατειακής του βάσης εκ των προτέρων. Αυτός ο συνδυασμός της παθητικής/ενεργητικής εποπτικής λύσης συστημάτων υπηρεσιών αναφέρεται ως CMaaS (Συνεχής Επόπτευση σαν Υπηρεσία) και ολόκληρο το πλαίσιο παρακολούθησης που το αφορά ονομάζεται OSAUM (Αυτόματες Υπηρεσίες Διαχείρισης / Παρακολούθησης OpenStack). Τέλος, αναλύουμε τη θεωρία και διευκρινίζουμε τα οφέλη του συστήματος αποθήκευσης Ceph και το προτείνουμε ως μία εναλλακτική λύση έναντι του block συστήματος αποθήκευσης Cinder επειδή επιδεικνύει χαρακτηριστικά ανθεκτικότητας και ανοχής σε σφάλματα κατά τη διάρκεια αποτυχίας σκληρών δίσκων. Από την πλευρά της δικτύωσης, προτείνουμε την εγκαθίδρυση του εικονικού πρωτοκόλλου εναλλακτικής δρομολόγησης (VRRP) για την υπηρεσία Neutron με σκοπό την επίτευξη υψηλής διαθεσιμότητας (HA) και προτείνουμε το Swift ως μακροπρόθεσμη λύση αποθήκευσης και αρχειοθέτησης δεδομένων το οποίο είναι κατάλληλο για τις ανάγκες μας.

Diakos Iraklis
M.Sc. Thesis
Computer Science Department
University of Crete
Master's Thesis Supervisor: Professor D. Plexousakis

Thursday, 30/6/2016, 11:00
Room B108, Computer Science dept., University of Crete

“Design and Implementation of a Monitoring System of Services’ Execution in the Cloud”

ABSTRACT

In the time of Big Data and Cloud computing, with the exponential increase of the information flow, there is an ongoing demand to satisfy the relentless need for data consumption by customers. However, SMBs and enterprises are faced with an ongoing dilemma of how to build, maintain, monitor and adapt an infrastructure to suit their business needs, reduce their investment capital (CAPEX) and operating maintenance costs (OPEX) and service their customers. Also, as more and more businesses enter the cloud race, customers demand not only their Service Level Agreements (SLAs) to be fully respected but also have the means to verify that this is the case. In order to cope with this, effectively, we incorporate OpenStack as our open source platform for laying the foundations of building a cloud computing infrastructure at the bare metal level. At first, we automate the whole process to produce a basic but fully operational and scalable virtualized infrastructure because it minimizes the effort of building a production – ready data center from ground zero. This includes security, computational, networking and ephemeral/persistent storage services. Packer is used for the templating process of VMs, Vagrant is used for the provisioning process of VMs and INI files bootstrap the setup process of VMs. Then, we highlight the challenges and the solutions of setting up time synchronization services (NTP, Chrony) because it is critical for the correct operation of our whole infrastructure. In addition, we attempt to automate the collection of our guest VMs’ basic information via a Windows service or a *nix daemon. We demonstrate the inner workings and minute details of writing an open source library, called CRAPI (Common Range API), that helped us in writing interoperable OS code for creating and installing Windows services or *nix daemons. For our purposes, the generated OS service

contains the core logic that handles the maintenance of important information about our VM guests. Moreover, we resort to Gnocci and install it at the host level in order to monitor our VM guests and monitor our instances that run in our VM compute nodes. Gnocci serves a dual purpose; on the one hand, it is used as a passive monitoring solution such as gathering basic system data about CPU/RAM/HDD/SSD/NIC resource consumption and utilization. On the other hand, it allows us to aggregate consumed VM data for billing and short – term archiving purposes (WIP – Work in Progress). Therefore, we introduce the concept of active monitoring duties. For instance, allocating a secondary vNIC to load balance the network traffic or provisioning an additional vCPU instance to assist a program in CPU bound tasks. These mechanisms enable us to avoid triggering one or more potential clause violations of Service Level Objectives (SLOs) in a SLA which has been agreed between a cloud operator and its customer base beforehand. We refer to this combination of a passive/active monitoring solution of services as CMaaS (Continuous – Monitoring – as – a – Service) and the whole monitoring framework that pertains to it is called OSAUM (OpenStack Automatic Management/Monitoring Services). Finally, we analyze the theory and elucidate the benefits of Ceph storage and propose it as an alternative to Cinder block storage because it exhibits resilience and fault – tolerance characteristics during hard drive failures. On the networking side, we suggest the deployment of Virtual Router Redundancy Protocol (VRRP) in Neutron to achieve high availability (HA) and we suggest Swift as a long term storage and archival data solution that is suitable for our needs.