**תרגיל בית 2**ב – עבודה ביחידים

**סטודנטית**: נוי ממן 313594434

**שם המאמר:** OpenStack: Toward an Open-Source Solution for Cloud Computing

**מטרת המאמר:** מחקר השוואתי של שלושה פתרונות שונים לפריסת קוד ענן פתוח. מחקר זה בא לפרט ולעזור לבחור את הפתרון המתאים ביותר, על הפתרון לענות כמה שיותר על הקריטריונים שלנו הכוללים פופולריות, קהילה, שימוש, מודולריות, פתיחות, קוד פתוח וחדשנות. מאמר זה אף מפרט עד לפרטים הקטנים על ארכיטקטורת הפתרון וכל חלקיו ויתרונותיו.

**תיאור טכנולוגיות\ מחקרים במאמר:** שלושה פתרונות שונים לפריסת קוד ענן:

1. Eucalyptus - אקליפטוס הוא פתרון המאפשר התקנת בית פרטי ותשתית ענן היברידית. זה כתוב בשפת Java, C ופייתון, עם בקר אחסון ראשי ובקרים בכל צומת. הרשת מנוהלת על ידי רכיב בקר הענן וכל בקר מאומת על ידי SSH-קבצי מפתח והרשאה לאמת עסקאות. קוד המקור של חלק מהמודולים שלו סגור, זו הסיבה שהוא פחות נפוץ מפתרונות אחרים.
2. OpenNebula - הוא פתרון ענן נוסף, תחת קוד פתוח, כתוב ב- C ++, Ruby ו- Shell. הרשת מנוהלת באופן ידני על ידי מנהל המערכת. וניתן לגשת לעותק ברירת המחדל של המכונות הווירטואליות באמצעות SSH. זהו פתרון קל לשימוש עבור מרכזי נתונים וענן פרטי.
3. OpenStack - גמיש, מוקדש לספק לתעשיית המחשבים את ההזדמנות לבנות אדריכלות מסיבית בעלת יכולת הרחבה וקוד פתוח לחלוטין. כל הקוד יכול להיות מותאם לפי הצורך. מאפשר ניהול גדול של מכונות וירטואליות המתוכנן לטווח ארוך ונפחים גדולים זה נראה כי הפתרון הטוב ביותר.

OpenStack Imaging Service - שירות הדמיה המספק שירותי אחסון. הקלטה והפצה של התמונות לדיסקי מכונה וירטואלית. הוא גם מספק API לביצוע שאילתות למידע על התמונות שמתארחות במערכות אחסון שונות.

OpenStack Object Storage - משמש ליצירת שטח אחסון שניתן להרחבה. שמערכת שתוכננה לטווח ארוך תוכננה במיוחד לטווח ארוך.

Other components - תכונות המתווספות לשיפור השירות:

- שירות זהות עבור OpenStack. רכיב החדש המטפל בבעיית האימות של OpenStack.

- Quantum, רשתות וירטואליות OpenStack ואיזון עומסים של OpenStack.

**סיכום המאמר:** מחשוב ענן הוא מושג חדש למדי הקשור לרוב לאספקת מנגנונים חדשים המאפשרים לספקים לתת למשתמשים גישה למספר בלתי מוגבל כמעט של משאבים. מחשוב ענן מורכב משלוש רמות של הצעות:

1. תשתיות כשירות (IaaS)- הציוד מסופק בצורה של מכונות וירטואליות. הלקוח מתחזק את היישומים, זמני ההפעלה, מאגרי מידע, תוכנת ושרתים בעוד שהספק שומר על שרת חומרה, אחסון, רשתות.
2. פלטפורמה כשירות (PaaS)- אתה יכול לפתח ולהחזיק יישומים באמצעות השירותים הניתנים. הלקוח מתחזק רק את היישומים האלה בזמן שהספק מתחזק את ענני הריצה, אינטגרציה של SOA, מסדי נתונים, תוכנת שרת, חומרת שרת ורשתות אחסון.
3. תוכנה כשירות (SaaS)- כל היישומים זמינים מרחוק, אפליקציות.

קיימים פתרונות שונים לפריסת קוד ענן פתוח, נראה מי הבחירה הטובה ביותר:

Eucalyptus- מדרגיות מוגבלת בהשוואה לקנה מידה מסיבי וקוד המקור של חלק מהמודולים שלו סגור. זו הסיבה שהוא פחות יעיל כפתרון. אקליפטוס הוא הפלטפורמה בשימוש על ידי אמזון EC2.

OpenNebula - הוא פתרון ענן תחת קוד פתוח כתוב ב- C ++, Ruby ו- Shell. הרשת מנוהלת באופן ידני על ידי מנהל המערכת. זהו פתרון קל לשימוש עבור מרכזי נתונים וענן פרטי.

OpenStack - תומך ברוב פתרונות הוירטואליזציה בשוק, גמיש, מוקדש לספק לתעשיית המחשבים את ההזדמנות לבנות אדריכלות מסיבית בעלת יכולת הרחבה וקוד פתוח לחלוטין. כל הקוד יכול להיות מותאם לפי הצורך ומאפשר ניהול גדול של מכונות וירטואליות המתוכנן לטווח ארוך ונפחים גדולים זה נראה כי הפתרון הטוב ביותר. הפתרון הטוב ביותר צריך לענות על שתי נקודות חשובות מאוד: מקור פתוח ותמיכה בתקנים חדשים, לכן רכיב זה הוא הטוב מכולם, הוא גם בעל קוד פתוח וגם מקיים מגמות המוכתבות על ידי צרכי המשתמשים ועומד בתקנים הפתוחים החדשים.

**האם מטרת המאמר הושגה?:** מטרת המאמר אכן הושגה. ניתן לראות כי במאמר באים להראות לנו פתרונות שונים לפריסת ענן פתוח. מאמר זה מפרט על כל פתרון את יתרונותיו וחסרונותיו ביחס לאחרים מביא דוגמאות ומציג את החברות המשתמשות בכל פתרון בתעשייה. לפי המאמר ניתן לראות כי עומד ברוב הקריטריונים וכי הוא הכי פופולרי ונפוץ מהשאר. מצוינים גם הטכנולוגיות הכלולות בתוך פתרון זה על מנת להבין את הארכיטקטורה העומדת מאחוריו ואת הסיבה שהוא הנבחר.

**הצעות נוספות שלי:**

קוד סגור המשומש כפתוח- לדוגמה Eucalyptus שזהו קוד סגור, על מנת להגדיל את מספר החברות המשתמשות בשירות ענו זה לאפשר לשירות להיות קוד פתוח על ידי השמת מנהל השירות שמעדכן/מאשר/מוסיף לשירות זה מה שמתבקש כאילו זה היה קוד פתוח בא ופן מבוקר ומאושר.

מימוש פתרון שירות ענן אישי- אפשרות פתרון ענן בו הלקוח הוא זה שמכתיב איך/מה/כמה לאחראיים על הנושא בשירות זה עם עלויות בהתאם. כך למעשה הלקוח לא צריך להתפשר/להתעכב בחיפוש שירות הענן האולטימטיבי בשבילו אלא הוא ישלם על שירות "מותאם אישית" שהוא מגדיר אך ורק על ידי בקשה שמותאמת לו.

בהצלחה!