סיכום AWS

# IAM - Identity and Access Management

* משתמש IAM הוא משתמש מערכת רשום, להבדיל ממשתמש ROOT שהוא המשתמש שהקים את המערכת ויש לו הרשאות בלתי מוגבלות, ולא מומלץ להשתמש בו.
* מטרת ה-IAM – לנהל משתמשים שונים לכל אחד, והרשאות הגישה המתאימות לכל אחד.
  + Identity information assurance – תיעוד בקשות של המשתמשים מה-AWS
  + Eventually consistent – המידע משוכפל על פני השרתים בכל העולם
* מונחים:
  + Principle – משתמש או תוכנה המזדהים כחשבון או role
  + Authentication – הזדהות לחשבון
  + Request – בקשה לביצוע פעולה, מכיל את הפעולה והמשאב עליו מבצעים את הפעולה
  + Authorization – הרשאה שחשבון מכיל. ההרשאות מסודרות ב-JSON. משתמש ללא הגדרה נחשב משתמש ללא כל הרשאה
  + Action/operation – הפעולה שתתבצע על המשאב, לאחר אימות ההרשאה
  + Resources – משאב זה אחד האובייקטים הקיימים בשירות ה-AWS
* IAM מכיל משתמשים, קבוצות והרשאות
* משתמשים:
  + משתמשים יכולים להזדהות לפי שמם, ה-ARN (amazon resource name, מזהה ייחודי בכל רחבי AWS) או ID (עבור ה-CLI/API)
  + משתמשים יכולים מאמתים זהות לפי סיסמה, access key, SSH או תעודה
  + עד 5000 משתמשים
  + המשתמשים מקבלים את ההרשאות לפי הקבוצות שהם בהם או ישירות
  + על כל או קבוצה ניתן לראות את ה- Identity information assurance עד 400 ימים
  + בדף הגישה להרשאות ניתן לראות איזה policy העניקה את ההרשאה למשאב המסוים, ולאתר את הקבוצה דרכה ניתן ה-policy ב-permission
* קבוצות
  + קבוצות לא יכולות להכיל קבוצות אחרות, אך משתמשים יכולים להכיל מספר קבוצות
* Policy – אוסף הרשאות הניתנות לשיוך לקבוצה או למשתמש
  + Policy לרוב בנוי מ-JSON, ומכיל 4 הגדרות: השירות המבוקש, הפעולות על השירות, המשאבים שיש לשירות, הגבלות ותנאים (כגון הגבלת זמן הפוליסה).
  + ניתן לראות משם מי מהמשתמשים ניגש לכל הרשאה.
  + סוגי הרשאה:
    - Inedtity-based – הרשאות הניתנות לקבוצה/משתמש
    - Resource-based (inline policy) – הרשאה המיוחדת למשתמש אחד בלבד, ומהווה חלק מהמשתמש
    - Permission Boundaries
    - Organization SCPs
    - Access Control List (ACL)
    - Session
* Role – מקבץ הרשאות הניתנות לשיוך באופן זמני
  + ה-role יכול להיקבע עבור אחד מהבאים: משתמש, משתמש מחשבון אחר, שירות (כגון EC2), ישות שבצעה הזדהות
  + ה-role מקבל policy, ישות שנגשה ל-role מקבלת את ה-policy האלה באופן זמני.
  + ה-role פעיל לפרק זמן שהוגדר, בין שעה ל12 שעות66עהצ
* למשתמש ROOT יש מספר הרשאות שאין לאף משתמש IAM
* ניהול התחברות: קיימת אפשרות לנהל את מדיניות הסיסמאות, ולהגדיר MFA (multi factor authentication)

EC2

* EC2 – elastic compute cloude – שרת כללי.
  + מוגדר במשאבים: CPU, Memory, Storage, Networking
  + AMI – Amazon Machine Images – image עבור שרתי EC2, עם מערכות הפעלה משתנות
    - יש סוגי AMI שונים: רשמיים, community, ופרטיים. ניתן להגביל מי יכול לגשת לכל AMI פרטי.
    - השרתים נוצרים עם משתמש ברירת מחדל ec2-user
    - אם ה-AMI מ-storage – הוא template. אם הוא מ-EBS – הוא image
  + ה-storage הינו זמני, וכל המידע שבו נמחק ברגע שה-EC2 נכבה.
  + EBS – elastic block store – אכסון המוצמד למכונה אחת, אך לא נמחק כשהיא נכבית
  + Security group - ניתן להגדיר לכל EC2 FW (whitelist) דרך AWS, הגדרות ה-security group יכולות להיות משותפות למספר שרתים, ולכל שרת יכול להיות עד 5 security group.
  + ניתן ליצור template עבור הגדרות שרת, וכך לא נצטרך להגדיר הכל מחדש, בכל פעם שניצור שרת
  + Elastic IP - לכל מכונה יש IP חיצוני (ופנימי) דינמי שנוצר לה. מכיוון שאנו זקוקים ל-IP חיצוני קבוע על מנת להנגיש שירות, נוצר Elastic IP שהוא דרך לשמור על אותה כתובת IP. ה-elastic IP מוגבל ל-EC2 מאותו region
  + ELB – LB שיכול לשמש עבור EC2.
  + ASG – auto scaling groups
  + Types – שינם מספר סוגים של שרתים, לפי קטגוריות (ימוש כללי, יעילות זיכרון, יעילות CPU וכד'), כל אחד מציע מגוון סוגי וגדלי שרתים, בהתאם לצורך. ניתן להחליף את סוג השרת בכל רגע נתון.
* EC2 – שרת כללי
* EBS – מהווה סוג של אכסון חיצוני עבור EC2, שניתן להעביר אותו משרת לשרת, מהיר יותר מ-S3
* EFS – elastic file system – עבור לינוקס, מהווה מעין NAS, ניתן להתחבר אליו מכמה שרתים במקביל.
* S3 – אכסון לטווח ארוך
* cloudFront – CACHE עבור S3
* AURORA – עוטף את ה-RDS כדי לבצע אוטומציות של הקמות DB
* RDS – DB רלציוני
* RDS Proxy – פרוקסי ל-RDS
* Elasticcache – CACHE עבור RDS
* dinamoDB – DB לא רלציוני
* DAX – cache עבור dinamoDB
* VPC – virtual private cloud – סביבות שונות על מנת להגביר אבטחה, ומכיל מתחתיו שירותים ואינטרנט מנותק (כמו להיות מאחורי FW עם VPN).
* Road53 - DNS

3 tier web application

Road53