



Moshav Bnei Zion P.O.Box 151, 60910 Israel Tel. 972-9-7907000 Fax. 972-97442444



סיכום מפגש שולחן-עגול

טכנולוגיות ומתודולוגיות בתחום השליטה
והבקרה הארגונית ו- CMDB

מנחה
פיני כהן

לקוחות נכבדים שלום,

תודה על השתתפותכם במפגש שולחן עגול Round Table בנושא טכנולוגיות ומתודולוגיות בתחום השליטה והבקרה הארגונית ו-CMDB. מצ"ב סיכום עקרי הדברים שעלו במהלך המפגש. במפגש עלו נושאים מהותיים שתומצתו בסיכום כפי שעלו. אין בסיכום זה המלצה גורפת ללקוחות אלא מתן פרספרטיבה והצגה של ההתלבטויות שעלו במפגש כלומר "מהשטח".

בדיון עלה שרובם המכריע של הלקוחות משתמשים ביותר מפתרון שליטה ובקרה אחד. גם לקוחות שבחרו ב- suite ארגוני סופו של דבר מכניסים מוצרים נוספים. אבל בכל מקרה ההמלצה הכללית היא לבצע פרויקטי שליטה ובקרה מ"למעלה למטה" ולא "לרוחב" כלומר לנטר מערכת אחת מהתחלה עד הסוף ולא (כפי שפעם היה מומלץ) לנטר את כל התקשורת, ולאחר מכן את כל השרתים, ולאחר מכן את כל מסדי הנתונים וכד'.

נושא שתפס משקל כבד בדיון היה ה- CMDB. על פניו עולה שיישום CMDB אינו משימה טריוויאלית וישנם מספר לא מבוטל של ארגונים אשר השקיעו מאמץ בנושא ולא קיבלו את התוצר הרצוי. בין הסיבות לכך ניתן למנות הגדרת דרישות מוגזמת לפרויקט ובמקביל כוח האדם הרב הנדרש לתחזוקת ה- CMDB. זאת מן הטעם שארגונים צריכים להשקיע מאמץ לא מבוטל בעדכון יכולות הגילוי והזיהוי של ה- CMDB. בדיון הועלתה הערכה שתחזוקת CMDB מצריכה אדם במשרה מלאה לסדר גודל של 1000 עד 2000 שרתים (לא כולל תחזוקת מערכת השליטה והבקרה הכללית).

עם זאת התועלות הפוטנציאליות ביישום CMDB הן גדולות ולכן ארגונים צריכים להשקיע יותר בהגדרת התוצרים המצופים ממהלך של יישום CMDB ובמקביל להיערך בכ"א מספיק וראלי. כמו כן שימוש בכלי עזר לגילוי וזיהוי תהליכים עסקיים כמו neebula (servicenow) או VNT מסייעים במשימה זו.

בברכה,

פיני כהן

תוכן

3.....	השכבות השונות המשתתפות ביצירת תמונת השליטה והבקרה
5.....	הקשר בין מערכות השליטה והבקרה למערכות אבטחת המידע SIEM
6.....	תועלות מצופות מיישום CMDB
8.....	לקחים מיישום CMDB
10.....	שימוש בכלי עזר לגילוי וזיהוי תהליכים עסקיים
11.....	תגובות של ספקים
11.....	Mobisec
12.....	חברת מתודה

השכבות השונות המשתתפות ביצירת תמונת השליטה והבקרה

בדיון ניסינו לסווג את השכבות הטכנולוגיות השונות הבונות את תמונת השליטה והבקרה.

1. השכבה הראשונה הנה השוב התשתיתי או ה"קלאסי". שם מדובר על טכנולוגיה שבמקרים רבים מבוססת על AGENT ולעיתים גם ללא AGENT המספקת נתוני תשתית על הרכיב התשתיתי. האם הרכיב חי (האם יש ping והאם ה- service ב- running), בכמה זיכרון משתמש, כמה דיסק פנוי, קצב התקשורת שצורך וכד'. תוצר בסיסי משכבה זו הנה מפה פיזית המתארת את התרכיבים בארגון ו"בריאותם". לעיתים גם בונים מפה של "מערכות" ואז לדוגמה עם במערכת SAP יש 3 שרתים בשכבת ה- APP SERVER האו שאם שרת אחד ייפול אזי בתמונה הפיזית אכן נראה "אדום" אבל בתמונה של המערכות נראה רק "צהוב" כי עדיין המערכת מתפקדת. לרובם המכריע של הארגונים ישנו יישום של טכנולוגיה מסוג זה.
2. השכבה השנייה היא שכבת חויית המשתמש. כאן מדובר על פתרונות שעוקבים אחרי חויות המשתמש או באופן סינטטי (מריצים סנריו מוגדר מראש של פעולות משתמש ומצפים לתגובה ידועה מראש) או באופן "אמיתי" - Real אשר עוקבים אחרי זמני התגובה של הלקוחות בפועל.

הסנריו הסינטטי מופעל גם בתוך הארגון וגם עבור מערכות WEB ללקוחות מחוץ לארגון על ידי ספקים שלהם יש יכולות לבצע פעולה זו ממקומות רבים בעולם וכך מקבלים מידע על זמני תגובה וזמינות ממקומות שונים בעולם. בכדי לבצע את ניטור חויית המשתמש הסינטטית ישנן טכנולוגיות המתקינות AGENT על מחשב ה-PC וישנן טכנולוגיות אשר מזריקות לדף ה-HTML קוד javascript אשר בודק מתי הגיע הדף ומתי התקבלה תשובה וכך מצליחים לחשב את חויית המשתמש למרות שאין התקנה של AGENT ב-PC. טכנולוגיה זו מוגבלת למערכות WEB ולא תעבוד במערכות client server. גם שכבה זו מצויה בשימוש בקרב רובם המכריע של הלקוחות.

3. השכבה שלישית היא שכבה שעוקבת אחרי המעבר של טרנזאקציה בין הרכיבים הטכנולוגיים בארגון. יש כאן בחינת הנתונים העוברים ברמת התקשורת וזיהוי של סוג הטרנזאקציה ופרטי הטרנזאקציה (קוד משתמש או קוד לקוח). בכל שכבה בודקים מתי הטרנזאקציה נכנסה ומתי יצאה וכך יכולים לחלק את הזמן התגובה בין הרכיבים השונים. ביצוע משימה זו מתרחש או באמצעות port mirroring של רשת התקשורת או באמצעות התקנה של AGENT. מדובר על טכנולוגיה חשובה שעדיין לא מוטמעת בקרב כל הארגונים. בזמנו נקראה שכבה זו BTM – business transaction management.

4. השכבה הרביעית היא שכבת ניטור ספציפית לטכנולוגיה. אם מדובר על שרת JAVA הרי הכלי כאן צריך להכיר JAVA ולהבין באמצעות ה-bytecode מה מתרחש בכדי לראות מדוע יש בעיה של איטיות. אם מדובר על .NET או DBMS יש להשתמש בטכנולוגיית ניטור שונה. מדובר כאן על כלים המשותפים בין צוות השו"ב לבין הצוותים המקצועיים (dbms, system, app development וכד') ולעיתים באחריות הצוותים המקצועיים בלבד. מדובר כאן על חקירה לעומק.

5. השכבה הבאה היא אולי השכבה המורכבת ביותר, המאתגרת ביותר ובעלת השונות הגדולה ביותר בין הארגונים. מדובר על השכבה שאוספת את החיוויים מכל הרכיבים השונים (לעיתים נקרא event manager) ומבצעת קונסולידציה ואינטגרציה תוך ניסיון לבצע root cause analysis – זיהוי גורם התקלה, impact analysis (על איזה תהליכים עסקיים תשפיע התקלה ולפיכך גם הבנה עד כמה חמורה התקלה), איזה פעולות יש לבצע באופן אוטומטי או ידני (התראות) ולעיתים גם משימות נוספות כמו capacity planning וכד'. מדובר על שכבת ה"חוכמה" שנקראת לעיתים גם BSM כאשר ההחלטה מתבצעת בהסתמך על האירועים שהתקבלו ב-event manager ועל המידע הנמצא כבר ב-CMDB (המכיל מידע יותר סטטי).

6. השכבה הבאה היא שכבת התצוגה המציגה לכל משתמש במערכת השו"ב את התמונה שהוא צריך לקבל. גם כאן ישנה שונות בארגונים – עד כמה לבצע קונסולידציה בין שכבות התצוגה השונות לתוך manager of manager ובאיזה מקרים להשאיר מספר קונסולים שונים המוסרים חיווי במקביל.

7. השכבה הבאה היא שכבת ה-CMDB. מסד נתונים האוסף מידע על הרכיבים (CI) השונים והקשרים ביניהם. לאחר קבלת המידע על הרכיבים יש ניסיון לאתר שירותים עסקיים ומערכות המאוגדות ל-service catalog. מעל שכבת ה-CMDB ניתן להפעיל מספר תהליכים – תהליכים המעדכנים את מערכות השליטה והבקרה, תהליכי ניהול שינויים וכד'. ה-CMDB גם מסייע בתהליך ה-BSM לדוגמה מתקבל חיווי על נפילה של FW וה-CMDB ימסור על איזה מערכות הדבר ישפיע. ה-CMDB מקבל את המידע על הרכיבים והקשרים ביניהם באמצעות כלים discovery מובנים בתוך החבילה או באמצעות כלי צד שלישי כגון VNT או Neebula (כיום servicenow).

בזמנו נהוג היה להפריד בהגדרות בין השכבות השונות אולם כעת נהוג לאגד את שכבות 2 עד 4 ולעיתים גם 5 ו-6 תחת השם הכוללני APM – application performance monitoring.

הקשר בין מערכות השליטה והבקרה למערכות אבטחת המידע

SIEM

נקודה מעניינת שעלתה בדיוק הנו הקשר בין התחום השליטה והבקרה לתחום אבטחת המידע. בעקרון כל אירועי השליטה והבקרה אמורים להיות חלק מאירועי אבטחת המידע. זאת מן הטעם שלא ניתן לדעת האם נפילה של שרת או התנהגות אפליקטיבית שאינה נורמאלית אינם חלק מאירוע אבטחת מידע.

אולם למרות שרבים מהאירועים משותפים לקבוצת התפעול ולקבוצת אבטחת המידע ולעיתים גם חלק מהתהליכים משותפים (מה מבצעים כאשר מתבצע חיווי מסוים) הרי שנדרשת הפרדה בין הצוותים השונים. זאת מן הטעם הבסיסי שארגון אבטחת המידע אמון על גילוי וזיהוי אירועים היכולים להיגרם גם מתוך הארגון ואפילו מתוך צוות התפעול. עם זאת מערכות השו"ב מנטרות את מערכות אבטחת המידע – בכדי לוודא שמערכות אלו עובדות באופן תקין.

תועלות מצופות מיישום CMDB

מצ"ב רשימה של תועלות אפשריות מיישום CMDB. מדובר על רשימה שהוגדרה בשולחן עגול קודם ולא ראינו צורך לעדכן רשימה זו באופן מהותי. בפועל מעבר לתועלת הבסיסית של עדכון מפת השליטה והבקרה (לדוגמה – אם מוסיפים שרת נוסף ל- SAP אזי שמערכת השליטה והבקרה תזהה אותו בתור שרת SAP ובמידה ותהיה בו תקלה תתריע בהתאם) הרי שרוב הלקוחות דיברו על תועלות ברמה התכנונית – "מה יקרה כאשר...".

התועלת	התייחסות STKI
עדכון מפות השו"ב	תועלת שכיחה.
לפני שמנתקים רכיב בודקים ב- CMDB האם הוא מחובר לרכיבים אחרים. זאת בכדי לוודא שלא מנתקים רכיב שנמצא בשימוש באופן שלא ידענו עליו	תועלת קלה ליישום
כאשר יש תקלה בודקים ב- CMDB באיזור הרלוונטי "מה השתנה מאתמול"	תועלת קלה ליישום. לא מתרשמים שיש שימוש רווח בתועלת זו.
דוחות Asset שונים. לדוגמה: איפה מותקן אורקל 8, איפה נמצאים מחשבי DELL מסוג מסויים וכד'. דוחות אלו משמשים לצרכים מגוונים – ביטוח המצאי בארגון, דוחות complianvce (האם יש חריגה ברישוי) וכד'. כאמור בהמשך לא משתמש ש- CMDB מחליף מערכות ASSET באופן מלא	תועלת בשימוש בינוני
הגדרת קונפיגורציה מועדפת golden image והשווה למה שמותקן בפועל	תועלת שנמצאת בשימוש נמוך. לעיתים משתמשים בכלי צד שלישי יעודים כגון Evolven.
קישור ה- CMDB ל- service desk. לכך תועלות רבות כגון קישור בין קריאה שפותח לקוח לטיפול בה במסגרת ניהול שינויים	חלק בסיסי בחלק מהפתרונות. לא נראה שביישום גבוה בקרב ארגונים.
ביצוע פעולות בסיסיות באופן אוטומטי. לדוגמא כאשר CMDB מגלה רכיב חדש אז הוא מתקין אליו AGENTS ומוסיף אותם לרשימת הניטור	יישום בינוני.

התייחסות STKI	התועלת
<p>תועלת חשובה ביותר. נמצאת בשימוש ביישומים מועטים בלבד.</p>	<p>ניהול change management בארגון. ביסוס ישיבות ה- CAB על בסיס מערכת ה-CMDB. תזמון שינויים. התייחסות לקריטיות של שינויים. גילוי "על מה משפיע" וכד'.</p>
<p>תועלת חשובה ביותר. נמצאת בשימוש ביישומים מועטים בלבד.</p>	<p>תועלת ספציפית בניהול שינויים – עדכון אוטומטי ממערכת השו"ב על כך שמבצעים טיפול במערכת בייצור כך שלא יתייחסו להתראות שמתקבלות בזמן ביצוע הטיפול</p>
<p>תועלת חשובה. נמצאת ביישומים מועטים בלבד.</p>	<p>עדכון דו סיטרי של ה- CMDB וסביבת ה-Business Process Analysis BPA, הן בהקשרים כלליים (רגולציות שמתבססות על BPA) והן בהקשר של פיתוח מערכות חדשות ככלי עזרה למנתחי המערכות</p>
<p>תועלת חשובה. ישנם יישומים ראשוניים. לדוגמה אם יש תקלה גם מ- DBMS וגם משרת (בו נמצא ה- DBMS) יש להגדיר את התקלה ב- DBMS בתור סימפטום. כנ"ל לגבי תקלה באחסון וברכיב ה- file system של שרת שנמצא על האחסון – יש להגדיר את חיווי ב- file system כסימפטום.</p>	<p>Topology Based Event Correlation – ביצוע קורלציה של תקלות באמצעות המבנה הטכנולוגי שמתקבל מה- CMDB.</p>
<p>לא מיושם בהקשר של CMDB</p>	<p>ביצוע מעבר מ- incident management ל- problem management</p>
<p>לא מיושם בהקשר של CMDB</p>	<p>Capacity Planning</p>
<p>תועלת חשובה ביותר. מתקשרת ל- Closed Loop Change Management. עדיין לא מיושמת בפרוייקטים בישראל.</p>	<p>בהקשר של ניהול שינויים – לוודא שכל השינויים שהתבצעו במערכת אכן עברו דרך מערכת ניהול השינויים.</p>
<p>עדיין לא בשימוש בישראל</p>	<p>מעבר למתודולוגיה של ענן פנימי שבו יהיה שיוך דינאמי של משאבים לפרוייקטים לפי דרישות הפרוייקטים, חשיבות עסקית, טיפול בתקלות לפי עדיפויות וכד'. חלק גדול מהכלים בתחום הם כלים שמקורן במערכות השו"ב וה- CMDB מהווה נדבך מרכזי בתפעול תפיסה</p>

התועלת	התייחסות STKI
זו. ניתן גם להסתכל על תועלת זו באופן יותר מצומצם- "מי משתמש ב- X web service?"	
שימוש ב- CMDB ובתשתית השו"ב לצורך קבלת חיווים עסקיים בזמן אמת ¹ – BAM Business Activity Monitoring	נמצא ביישום אצל חלק מהארגונים- לא קשור ישירות ל- CMDB
תעדוף עיסקי נכון יותר של תקלות הניטור המתגלות כתוצאה מהבנה של השירות העיסקי שמושפע מאותה תקלה הודות לימפת השירות העיסקי שמופיע ב CMDB ² .	להערכתנו מספר מועט של לקוחות הגיע לתועלת
בניית הקונסול המרכזי כרכיב שמתבסס על המידול שמתבצע ב- CMDB (ולא על פי הרכיבים הפיזיים שמנוטרים בשו"ב)	יש ארגונים שמתחילים להטמיע קונסול מרכזי לפי תפיסה זו.

לקחים מיישום CMDB

לקוחות שיתפו את החוויות שלהם מיישום CMDB. אחד הלקוחות תאר מצב שבו בסופו של דבר לא הפיקו תוצרים משמעותיים מהפרויקט. לדעת הלקוח הסיבה לכך היא בשלות נמוכה של הפתרונות בשוק המחייבת השקעה גדולה מבחינת כ"א ותשומת לב ארגונית. התוצר שמתקבל מיישום CMDB הוא ניהול מצאי תוך קבלת דוחות על מה מותקן. תוצר זה הנו ברמה גבוהה. אולם מבחינת גילוי הקשרים בין הרכיבים האפליקטיביים למרות המאמץ לא הגיעו לתוצאה טובה. במקרים רבים הכלי לא יודע לבצע מיפוי אפליקטיבי וגם כאשר יש מיפוי כזה הוא לוקה בחסר. תהליך המיפוי האפליקטיבי מתבצע באופן כזה. בתחילה מקבלים מידע ממנהלי המערכת על הרכיבים השונים במערכת, אופן פעולתם והקשרים בתוך המערכת ולמערכות משיקות. לאחר מכן מבצעים סריקה באמצעות רכיב ה- cmdb discovery ואז מעבדים ומטייבים את תוצאת הסריקה. לדוגמה, ה- CMDB גילה ששרת X יוצר קשר עם שרת Y אולם מסתבר שהדבר אינו חלק מהתהליך העסקי כמו לדוגמה שרת Y הוא שרת אנטייורוס שמעדכן הגדרות. דוגמה

¹ בניגוד למערכת DW שמסוגלות לספק חוויים עמוקים אך לרוב לא בזמן אמת
² הקרדיט לנועם צוברי מ- HP

נוספת היא שרת X יוצר קשר עם שרת Y כחלק מהתהליך העסקי אולם בטעות שרת Y נמצא בסביבת ה- DR ולא בשכבת הייצור. כלומר יש כאן טעות בקונפיגורציה האפליקטיבית של המערכת. מדובר על תהליך שחלקו אוטומטי אבל כמעט תמיד יש רכיב ידני (כולל בדיקה עם מנהל המערכת) ולכן יישום ה- CMDB באופן כללי הנו משימה הצורכת משאבים. גם כאשר לוקחים בחשבון שהמערכות משתנות כל הזמן ובפועל יש לבצע תהליך זה באופן תמידי (ברור שמפיו של שינוי בתהליך עסקי לתוך ה- CMDB יערך פחות ממיפוי בפעם הראשונה אך עדיין מדובר על תהליך זהה במהותו).

נקודה נוספת שמקשה על קבלת התועלות המקסימליות מיישום CMDB היא העובדה שהסריקה מתבצעת רק פעם או פעמיים ביום ולכן כבר קרו מקרים שבהם התבצעה אצל לקוח פעולה לא רצויה (שינוי בהגדרות FW) שה- CMDB אמור היה לזהות ולהתריע. אולם מכיוון שהסריקה באותו ארגון מתבצעת פעמיים ביום, התרחשה כבר תקלה לפני שהתבצעה הסריקה שאמורה הייתה להתריע על השינוי הלא רצוי.

לקוח נוסף תאר את יישום ה- CMDB בארגונו. בארגון עושים discovery לרוב הרשתות והתוצאה מתממשת עם ARIS (כלי חיצוני) המשמש כמאגר מרכזי למיפוי התהליכים העסקיים בארגון³. הממשק בין מערכת ה- CMDB ל- ARIS הנו ממשק צמוד. כלומר כאשר מתארים תהליך עסקי ורוצים לציין איזה מערכת מידע מבצעת תהליך זה בשלביו השונים – ניתן לבחור רכיבים רק מתוך ה- CMDB. כמו כן, בעקרון, כל פעם שמוסיפים רכיב למערכת מידע (או רכיבים) יש לציין מהו התהליך העסקי שאליו הוא משתייך מתוך ה- ARIS. כאשר מתרחש מצב שבו ישנה הוספה של רכיב (CI) שמתגלה על ידי הסריקה של ה- CMDB ישנו תהליך לשיוך ה- CI לתהליך העסקי, זאת תוך שימוש בקונבנציית שמות בשילוב של אינדיקטורים נוספים כמו כתובת סגמנט רשת וכד'. במידה ולא מצליחים אתר באופן אוטומטי מהו התהליך העסקי אליו שייך ה- CI מתבצע תהליך ידני ולפיו מחליטים לאיזה תהליך עסקי ב- ARIS ה- CI שייך.

המידע שנמצא במערכות משמש באופן יומי בין היתר לניהול שינויים. כאשר מזינים שינוי במערכת ניהול השינויים, מתוך ה- CMDB מקבלים מידע על התהליכים העסקיים שעלולים להיות מושפעים מהשינוי. זאת מכיוון שבחרים במערכת ניהול השינויים את הרכיב ורואים דרך ה- ARIS באיזה תהליכים עסקיים הוא משתתף.

³ ישנם תועלות רבות בשימוש בכלי למיפוי תהליכים עסקיים בארגון – כמו בסיס לניהול סיכונים תפעוליים, עמידה ברגולציה, הגדרת תפקידים ועוד ועוד. בדרך כלל פרויקט זה מטופל על ידי גורמי האו"ש.

כמו כן משמש המידע ב- CMDB לעדכון מפות השו"ב – כאשר לדוגמה מתווסף CI נוסף, לפי ה- CMDB מחליטים לאיזה מפת שו"ב להוסיף את ה- CI וכיצד לנטר אותו. כמו כן ה- CMDB משמש למשימות עד הוק לדוגמה – איפה מותקנות גרסאות X JAVA, עם איזה גורמים מדבר שרת DBMS מסוים וכד'.

שימוש בכלי עזר לגילוי וזיהוי תהליכים עסקיים

בדיון עלה הצורך לשימוש בכלי עזר למיפוי תהליכים עסקיים מעבר ליכולות המיפוי שנמצאות בתוך חבילות ה- CMDB כאשר שני הכלים שהוזכרו היו neebula (כיום servicenow) ו- VNT. neebula ו- VNT עובדות בצורה שונה.

כאשר neebula עוקבת אחרי טרנזאקצית WEB לדוגמה אז היא מסתכלת על שרת ה- IIS ורואה עם איזה DBMS (או רכיבים אחרים) ה- IIS מחובר ולפי זה "מבינה" לאן להמשיך. כלומר לפי הקונפיגורציה של האפליקציה והתשתית.

כאשר VNT עוקבת אחרי טרנזאקציית WEB אז היא מסתכלת על התקשורת שעוברת במערכת. ואז אם היא רואה ש- pattern מסוים של תקשורת עובר מה- IIS לשרת אחר, והדבר הזה קורה הרבה, אז היא מסיקה שבטרנזאקציה הזו ה- IIS מדבר עם אותו שרת. אחד הלקוחות תאר מצב שבו תוך יומיים VNT זיהה בצורה דיי טובה תהליך CRM.

על פי הנאמר בדיון מדובר על כלי אוטומציה לזיהוי תהליכים עסקיים מתקדמים אך בשני המקרים צריך לעיתים לבצע התערבות ידנית. לדוגמה, ב- neebula, אם יש שרת ה- IIS מדבר עם DBMS לצרכי שליחת לוגים עסקיים – אז ב- neebula זה יראה כאילו שרת הלוגים העסקיים הוא חלק מהתהליך העסקי – ואז צריך לתקן זאת ידנית.

לדוגמה ב- VNT אם ברוב המקרים שרת ה- IIS מדבר עם שרת X אבל במקרי קצה מדבר גם עם שרת Y אז יכול להיות שבזמן שה- VNT לומד את הארגון לא יהיו הרבה הפניות לשרת Y ולכן VNT לא יכלול אותו בתהליך העסקי – וגם כאן צריך לתקן ידנית. כנ"ל אם בשרת יש כמה רכיבים – ה- VNT לא יוכל לזהות ולדווח על הרכיבים כי הוא מסתכל על התקשורת. neebula נרכשה על ידי servicenow אבל על פי המידע שנמצא ברשותנו כעת אפשר לרכוש פתרון זה גם בנפרד.

תגובות של ספקים

Mobisec

מערכת CMDB הינה אבן יסוד מרכזית ביישום IT as a business. מערכות CMDB הינן מורכבות ליישום ובסופו של דבר בגלל עלויות, סדרי עדיפויות ותחזוקה שוטפת אנו מוצאים יותר ויותר ארגונים שיישמו את המערכת באופן חלקי ולכן גם התוצר המתקבל ממנה הינו רדוד ולא עונה על הצורך הארגוני. ההתייחסות בין השאר לניהול משאבי ה IT ויחידת המחשוב הארגוני עוברת מהפך דרמטי בשנים האחרונות. מנהלי ה IT נדרשים לשקול שיקולים עסקיים ומקצועיים בתהליך קבלת ההחלטות הכרוכות בפיתוח ובניהול התשתיות (IT operations and Infrastructure) לצד הצורך לתת שקיפות ביחס לניהול הסיכונים (Risk Management) ואכיפת המדיניות הארגונית (Compliance & Governance).

הסוגייה המרכזית העומדת בפני מנהלי ה IT הינה ביכולתם לנהל בין השאר את תשתיות ה IT בצורה אפקטיבית ופרואקטיבית. הכישלון באי יישום מלא של פרויקטי CMDB בארגונים מותיר חלל ניהולי. המידע אותו מקבלים המנמ"ר ומנהל תשתיות המחשוב הארגוני אינו אחוד ומתקבל מכל מערכת תשתית בנפרד ובמגבלות היצרן, שלרוב אינו מביא בחשבון את ההשפעות של מערכות אחרות על אותה סביבה או את ההשפעה של המערכת על תפקוד מערכות אחרות. סוגיה זו מקצינה לאור המציאות בה רכיבי תשתית בתחום מסויים (Servers, Storage, Backup, HBA ועוד) הינם מתוצרת של מספר Vendors באופן שאפילו המידע הזה הסגמנטלי, אינו אחוד ונגיש. אם היינו צריכים לספק תשובה ברורה לסיבת כישלון היישום של מערכות ה CMDB בארגונים, כנראה שרשיה נעוצים בריבוי הדרישות מהמערכת ומהסתירה המובנית שבשימוש במערכות הן לשני צרכים שונים, תפעוליים וניהוליים. עד היום, הדרישה התפעולית לא גובתה במידע אובייקטיבי לצורך קבלת החלטות ניהוליות. הבקרה הנוצרת משנה את אופן ניהול התשתיות ומשפיעה על אופן קבלת ההחלטות התפעוליות.

הסוגייה של ניהול תשתיות ה IT אינה עוסקת רק בהקצאת משאבים, אלא נדרשת גם ליכולת לתת למנהלי ה IT מענה לסוגיות של ניהול פרואקטיבי של התשתיות הארגוניות. המאמץ לשמור על רמת שירות של 99.999 מחייבת בין השאר אכיפה של המדיניות הארגונית בתחומי התשתיות והגיבוי שתאפשר לאתר חריגה ממדיניות/פוטנציאל תקלה עוד בטרם התרחשותה.

התופעה של שימוש בפתרונות יעודיים במקום במערכות ה CMDB הרחבות מצביע על התפכחות מסויימת של מנהלי ה IT מה"חלום" והבנה, שלצד CMDB "רזה" יש לעבוד עם מערכות נוספות, ייעודיות, חלקן ברמת התהליכים העסקיים שהוזכרו בסיכום השולחן העגול. פתרון שלא דנו בו הוא העוסק בניהול התשתיות עצמן והוא הפתרון של חברת iTAnalyzer, חברת סטארטאפ ישראלית מנס ציונה. מנהלי ה IT מקבלים באמצעות הפתרון תמונת תשתיות כוללת תוך ניהול אפקטיבי ופרואקטיבי של תשתיות המחשוב והפחתת העלויות והסיכונים הכרוכים בתפעולם, באמצעות מודולים אנליטיים ייחודיים. המערכת אוספת מידע מכלל מערכות תשתיות המחשוב (שרתים, איחסון, גיבוי, בקרים ועוד...) באופן עצמאי ללא תלות במערכות המידע או במערכות הניהול המסופקות על ידי היצרנים השונים, מנתחת את המידע באמצעות מנוע אנליטי מתקדם וייחודי והצגתו בממשק אחד ברור, מובן ובשפה עסקית. המערכת מציגה את המגמות המתפתחות לאורך זמן בתחומים הרלוונטיים, מתריעה על הפרות במדיניות הארגונית בתחומי התשתיות והגיבוי, מאפשרת תכנון תקציבי ומסייעת בניהול נכסי המחשוב וביישום גישת IT as a business. הקמת המערכת בארגון היא מהירה מאוד והתוצאות המוכחות שלה מצביעות על הפחתת עלויות ושיפור במדדים התפעוליים תוך הפחתת הסיכונים מייד עם עלייתה לאוויר. לחברה לקוחות ממגוון ענפי המשק והיא מקיימת שיתופי פעולה עם החברות HP, IBM, EMC, Informatica.

לפרטים: גדי גילאון, מוביסק טכנולוגיות, 054-9209999

חברת מתודה

כרמל פירוטין, CTO מתודה, ITIL V3 Certified EXPERT

מניסיון מצטבר עולה שמרבית פרויקטי CMDB לא נחלו הצלחה והסתיימו באי עמידה במטרות שהוגדרו.

להלן תובנות והמלצות חברת מתודה, שמטרתם למקד את הפרויקטים ולהביא להצלחתם. האתגר העומד בפנינו הוא שילוב שירותים אפליקטיביים עם CMDB תשתיתי - הרחבה ל-ACMDB

(Application CMDB).

עקרונות

- ✓ **התמקד בעיקר** (תפסת מרובה לא תפסת) - התמקד במספר תהליכים קריטיים במקום למפות את כל ה-IT
- ✓ **תכנן רחב ורוחבי אך ממש בשלבים**, התחל בקטן, אמצ' גישה אג'ילית למחזור החיים, חפש את התועלת עכשיו
- ✓ **הגדר תהליכי עבודה** ב-CMDB **ותעדף אותם** לכניסה לייצור - הגדר ואפיין במדויק את שימושיות תהליכי ITIL ב-CMDB, עם התאמה לתכנים ומבנה המידע, ואף שקול התאמת והכוונת התהליך לאילוצים
- ✓ **הגדר את המאגרים** השותפים לפדרציה
- ✓ **הגדר את תהליכי האכלוס והעדכון** של פרטי התצורה

טיפים להקמת מערכת

- ✓ שילוב גילוי ומיפוי Bottom-Up לתשתיות רוחביות ו-Top-Down ליישומים
- ✓ לפני אכלוס מידע ב CMDB יש להחליט איך ובאחריות מי לעדכן אותו בשוטף
- ✓ בניית ותחזוקת עצי שירות היא פעילות מתמשכת, שחייבת להתבצע כחלק מתחזוקה שוטפת של המערכת
- ✓ אל תכפו את ממשק התצוגה על הארגון
- ✓ לא הכול חייב להיות אוטומטי
- ✓ להגדיר תפקידים ואחריות: מנהלן (מומחה מערכת), מידען (מומחה תוכן) ומנהל תהליך בקרת תצורה