



האוניברסיטה הפתוחה

המחלקה למתמטיקה ולמדעי המחשב

החטיבה למדעי המחשב

פרויקט מתקדם במדעי המחשב:
יישום מעשי וסימולציה של מודל
Trust-based Dynamic RBAC

מנחה: פרופ' אהוד גודס

מס' גרסה: 1

סמסטר א' 2017

מגיש:

תמיר לביא, ת.ז. 032409542

תוכן עניינים

3.....	רקע	1
3.....	תיאור הפרויקט	2
3.....	מודל הרשאות TDRBAC	3
4.....	הגדרות המודל	3.1
5.....	תיאור מערכת הסימולציה	4
5.....	סביבת העבודה	4.1
5.....	מבני נתונים - תשתית	4.2
5.....	מבני נתונים - אובייקטים וישויות	4.3
7.....	מסכים	4.4
8.....	אתחול הנתונים	4.5
8.....	הרצת המודל הבסיסי בסימולציה	4.6
9.....	טיוב רמת האמון הנדרש בשיוך הרשאות לתפקידים (RPA)	5
11.....	מדד דרגת השימושיות של המערכת - Usability Degree	5.1
12.....	השפעת תהליך הטיוב על מדד ה- Usability Degree	5.2
13.....	תקריות בסיכון	5.3
14.....	השפעת טיוב רשומות ה- RPA על התנהגות המשתמש ביחס לשינוי רמת ההרשאה	5.4
15.....	מסקנות	6
15.....	סיכום	7

1 רקע

מערכת הרשאות הינה מרכיב חשוב ובסיסי בכל מערכת מחשב. הצורך לקבוע עבור כל ישות במערכת מה הן הפעולות המותרות ו/או האסורות לביצוע מהווה בדרך כלל תנאי הכרחי לתפקוד תקין של המערכת ולמניעת שימוש לרעה במערכת. במהלך השנים נבנו מודלים שונים לניהול מנגנון ההרשאות שתי גישות עיקריות ובולטות בתחום זה הינן:

- גישה מבוססת תפקידים (RBAC). על פי גישה זו, המשתמשים משויכים לתפקידים במערכת, ואילו ההרשאות ניתנות לתפקידים כך שהמשתמשים מקבלים את ההרשאות בהתאם לתפקידים המשויכים להם.^{1, 2}
- גישה מבוססת תכונות (ABAC). על פי גישה זו, הרשאות המשתמש נקבעות בזמן הריצה בהתאם לתכונות שונות (לרבות התפקיד, למשל).³

הגישה הראשונה נחשבת לחסינה יותר ומאפשרת ניהול ובקרה ברורים באשר למצב ההרשאות במערכת, אך יחד עם זאת שיטה זו אינה מותאמת למערכות בהן יש מספר רב של משתמשים או ישויות במערכת או אם נדרשת מידה רבה של גמישות בהענקה או הסרה של הרשאות בהתאם לתכונות משתנות.

מאידך – הגישה השנייה, בה ההרשאות נקבעות בהתאם לחוקים המבוססים על תכונות הישויות במערכת הינה מדויקת ודינאמית מצד אחד, אך מסורבלת מאוד לניהול ובקרה ועלולה ליצור מצבים בלתי רצויים של מתן הרשאות יתר מבלי שיש למנהל המערכת יכולת טובה לבדוק ולבקר את סטטוס ההרשאות במערכת.

במסגרת עבודת תזה בכותרת Trust-based Dynamic RBAC, פותח והוצג מנגנון בקרת גישה אשר מרחיב את המודל ההרשאות RBAC. המנגנון המוצע משלב התייחסות לרמת האמון של האובייקטים (המשתמשים) במערכת ובכך לאפשר מידה מסוימת של גמישות בחלוקת ההרשאות מחד, ומאידך לשמור על המבנה החסין של המודל המקורי ולהתאימו לסביבות של מערכות פתוחות בהן יש מספר רב של משתמשים. בעבודת התזה מוצעים כלים ורעיונות לטיוב השימוש במודל וכן הרחבות המאפשרות מענה לנושא פרטיות וכן לנושא האצלת הרשאות (Delegation) תוך התייחסות לנושא רמת האמון. המודל הוצג במסגרת כנס ICISSP, ופורסם ב- SCITEPRESS⁴

2 תיאור הפרויקט

פרויקט זה הינו עבודה יישומית אשר במסגרתה פותחה מערכת תוכנה שממחישה יישום ממשי של מודל ההרשאות TDRBAC. כחלק מהמערכת נעשה שימוש בבסיס נתונים ובו אובייקטים המדמים משתמשים והרשאות. המערכת מיישמת את המודל TDRBAC ומציעה ממשק משתמש לטובת "משחק" עם המודל, תוך בחינת המשמעויות הנלוות לשינוי פרמטרים במערכת ובמנגנונים של המודל וכן מדידה של פרמטרים שונים הקשורים במנגנון ההרשאות, ניהולו ואופן השימוש בו.

במסמך זה יוצגו תכונות מערכת הסימולציה, הדגמת השימוש במערכת וכן מסקנות הנובעות מיישום בפועל של המודל התיאורטי.

3 מודל ההרשאות TDRBAC – תיאור בסיסי

מודל ההרשאות TDRBAC, הינו מודל אשר משלב בין הצורך במערכת הרשאות קשיחה ומוגדרת היטב, לבין הגמישות הנדרשת במערכות פתוחות מרובות משתמשים. מודל ההרשאות הוותיק RBAC מציע את החוזק הנדרש והמוכח להגדרת מדיניות ההרשאות של מערכות רבות. יחד עם זאת, במקרים בהם ישנו מספר רב של משתמשים, אנו מזהים תופעה של ריבוי תפקידים⁵. כמענה לתופעה זו פותחו מודלים חדשים, כדוגמת ABAC (Attribute Base Access Control) – מודל הרשאות המבוסס על

מאפיינים של המשתמש, כגון תפקיד, ותק, שיוך ארגוני ודרגה. במודל זה ניתן להתייחס גם לרמת ההשאה של המשתמש בכל רגע נתון. מודלים המבוססים על ABAC, הינם מודלים אשר מאפשרים גמישות רבה בקביעת מדיניות ההרשאות של המערכת, אך מנגד ישנו ויתור משמעותי על היכולת לשלוט, לנהל ולבקר את הרשאות המערכת לאורך זמן.

מודל ההרשאות TDRBAC מבוסס על המודל RBAC ומרחיב אותו באופן שמאפשר את הגמישות הנדרשת בניהול של מספר רב של משתמשים, תוך התייחסות לרמת ההרשאה של המשתמש מחד תוך שמירה על השלמות ויכולת הבקרה של מודל ההרשאות המקורי.

3.1 הגדרות המודל

המודל מתבסס על ההגדרות הקיימות במודל RBAC ומוסיף את ההגדרות הבאות

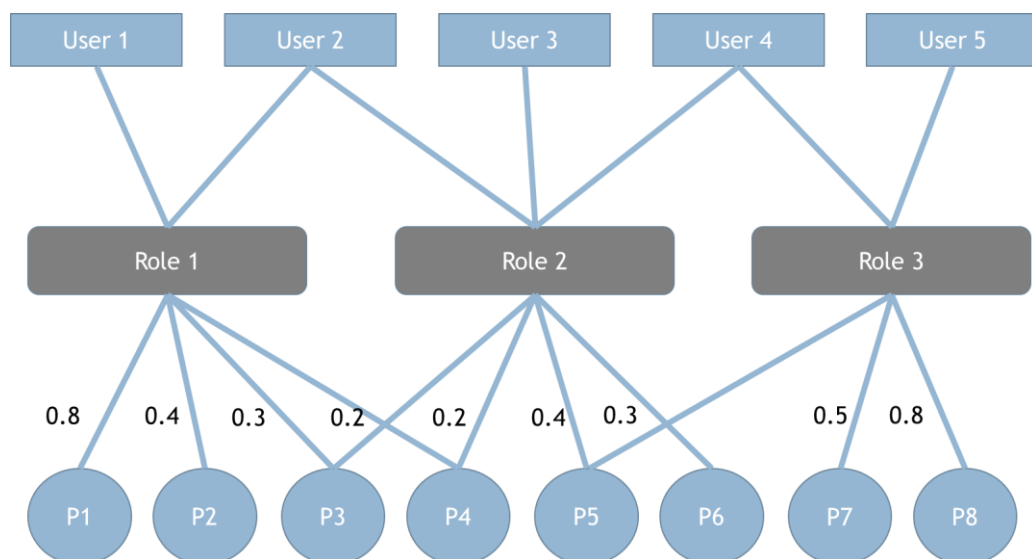
- **userTrust** – מספר שנע בין 0 ל-1, המתאר את רמת האמון של משתמש במערכת כאשר 1 משמעותו אמון מלא, ו-0 הינו חוסר אמון מוחלט במשתמש
- **permissionTrust** – רמת האמון המינימלית הנדרשת למשתמש כך שיורשה להשתמש בהרשאה ספציפית
- **URA** – User to Role Assignment, שיוך של משתמש לתפקיד
- **RPA** – Role to Permission Assignment, שיוך של תפקיד להרשאה ספציפית. שיוך זה יכול את רמת האמון הנדרשת. לדוגמא:

ROLE	PERMISSION	Trust
Manager	Can assign roles to users	0.9
Guest	Read public posts	0
Admin	Change system configurations	1

3.2 תיאור גרפי

ניתן לתאר את המודל בגרף באופן הבא:

כל משתמש משויך לתפקיד. כל תפקיד משויך להרשאה ספציפית, כאשר המספר על הקו המקשר מצוינת רמת ההרשאה המינימלית הנדרשת מהמשתמש כדי שיוכל לעשות שימוש בהרשאה



3.3 בקשה לקבלת הרשאה

כאשר המשתמש מבקש לעשות שימוש בהרשאה מסוימת, המערכת פועלת בהתאם לתהליך המתואר להלן, על מנת להחליט אם המשתמש יכול או לא יכול לקבל את ההרשאה

Algorithm 1 Decision Function

Require: *user, permission*

```
for each ura ∈ URA where ura ∈ user × ROLES do
  for each rpa ∈ RPA where rpa ∈ ura.ROLES × permission × [0, 1] do
    if rpa.Trust == 0 then
      return ACCEPT {Avoiding the calculation of userTrust when it's not necessary}
    else if rpa.Trust ≤ user.Trust then
      return ACCEPT
    end if
  end for
end for
return REJECT
```

4 תיאור מערכת הסימולציה

4.1 סביבת העבודה

מערכת הסימולציה פותחה בסביבת "קוד מנוהל", על בסיס פלטפורמת הפיתוח Net. מבית מיקרוסופט ובשפת התוכנה C# כתוכנה מסוג Windows Forms. הפיתוח בסביבה זו הינו קל ומהיר באופן יחסי. בדומה לשפות תוכנה הפועלות תחת "קוד מנוהל" (כגון JAVA), סביבה זו אינה נחשבת ליעילה ביותר בהיבט של ביצועים אך לטובת יישום מנגנון הסימולציה תכונה זו בעייתית.

4.2 מבני נתונים - תשתית

המערכת עושה שימוש במבנה נתונים המבוסס על הספרייה DataSet. מבנה זה מאפשר להחזיק טבלאות נתונים ולקשר אותן לבסיסי נתונים מסוגים שונים לטובת אחסון המידע (כגון MySQL, MSSQL וכד'). על מנת לשמור על רציפות נתונים מצד אחד ועל יכולת לשמור מצבים סטטיים (Snap-Shots) של בסיס הנתונים, בחרתי להשתמש באחסון הנתונים בפורמט XML, אשר נתמך ע"י הספרייה DataSet. מבנה זה הינו קל לתחזוקה וניהול, חסכוני במקום ואינו מצריך החזקת שרת בסיס נתונים ייעודי.

4.3 מבני נתונים - אובייקטים וישויות

במערכת מנוהלים האובייקטים הבאים (כל אובייקט הינו טבלה)

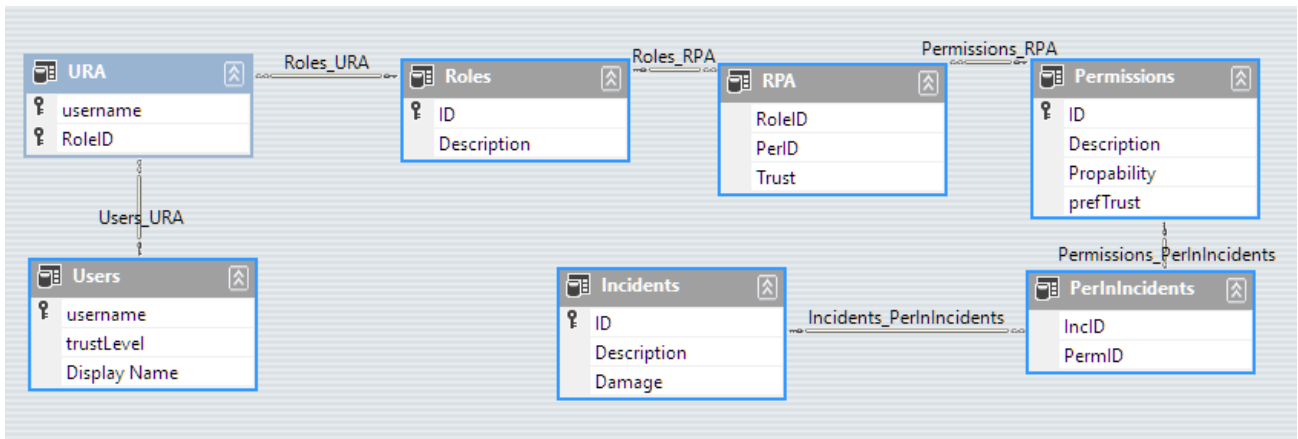
- משתמשים
- הרשאות
- תפקידים
- תקריות - Incidents (בהמשך יוסבר בהרחבה מהן תקריות)

טבלאות קשר בין ישויות:

- טבלת קשר של משתמשים לתפקידים (URA)
- טבלת קשר של הרשאות לתפקידים (RPA)
- טבלת קשר של תקריות להרשאות (PerInIncidents)

כל הקשרים בין אובייקטים ראשיים (כגון: משתמשים, הרשאות) הינם מסוג Many to many, כלומר ניתן לקשר כל אובייקט למספר אובייקטים אחרים מאותו סוג. הדרך ליישם את הקישורים במבנה הנתונים הינו באמצעות טבלת קשר נפרדת.

להלן תרשים המתאר את סכמת מבנה הנתונים במערכת:



מסך ראשי

המסך הראשי מציע כפתורי פעולה כגון פעולות לניהול הנתונים, וכן תצוגה של נתונים שונים המשקפים מצב נוכחי במערכת, כגון תוכן טבלאות הנתונים ואת מספר הרשומות בכל טבלה. המסך מאפשר עריכה ידנית של כל הטבלאות במערכת.

כפתורי הפעלה

מדדים

username	trustLevel	Display Name
user1	0	User 1
user2	0.34	User 2
user3	0.48	User 3
user4	0.2	User 4
user5	0.07	User 5
user6	0.43	User 6
user7	0.87	User 7
user8	0.36	User 8
user9	0.28	User 9
user10	0	User 10
user11	0.69	User 11
user12	0.07	User 12

מסך תצוגת משתמש

מסך זה מאפשר קבלת תמונת מצב על משתמש ספציפי, וכן לשנות באופן ידני את רמת האמון של המשתמש. המערכת מציגה את ההרשאות של המשתמש באופן דינאמי – באופן שמציג את ההרשאות הזמינות לו בהתאם לרמת האמון, ואת אלו אשר נלקחו ממנו בשל רמת האמון.

User View

User: user1

User's Trust Level: 0.56

Roles: Role 23

Allowed Permissions: Permission 149, Permission 263, Permission 478, Permission 357, Permission 191, Permission 791, Permission 393

Prevented Permissions: Permission 716, Permission 17, Permission 346

4.5 נתוני אתחול

עם עלייתה, המערכת טוענת לזיכרון את קובץ הנתונים המכיל את האובייקטים של המודל. האובייקטים עליהם הופעלו הסימולציות כוללים:

- 10000 משתמשים
- 100 תפקידים
- 1000 הרשאות
- 100 תקריות אבטחת מידע
- 10987 שיוכים של משתמשים לתפקידים (URA)
- 1000 שיוכים של תפקידים להרשאות (RPA)
- 200 שיוכים של הרשאות לתקריות (Inc/Per)

הנתונים נבנו באופן אקראי לחלוטין באמצעות מתודות ייעודיות שנבנו לצורך הסימולציה. המערכת מאפשרת להוסיף ישויות ושיוכים שונים לפי דרישה באמצעות כפתורי ההפעלה. כמוכן, המערכת מאפשרת עריכה של כל הנתונים וניתן לשמור מצב קיים או לטעון את המידע לאחר עריכה או החלפה ידנית של קובץ מקור הנתונים (קובץ XML).

4.6 קוד המקור

קוד המקור מצורף לעבודה זו והוא מהווה חלק בלתי נפרד ממנה. כאמור בסעיף 4.1, הקוד נכתב בשפת התוכנה C#. הקוד כולל מספר קבצים עיקריים ובהם:

- קובץ המתאר את התצוגה הגרפית של המסכים השונים
- קובץ המתאר את מבנה הטבלאות (קובץ xsd), את תכונות הטבלאות, השדות ואת הקשרים ביניהם כמתואר בסעיף 4.3
- מספר מתודות המטפלות ביצירת אובייקטים באופן אוטומטי, כגון משתמשים, תפקידים הרשאות ועוד
- מתודות המטפלות בשמירה וטעינה של נתונים קודמים שנשמרו
- לב המערכת מצוי בקבצי הקוד של המתודות המיישמות את המודל וניתוח הנתונים

4.7 הרצת המודל הבסיסי בסימולציה

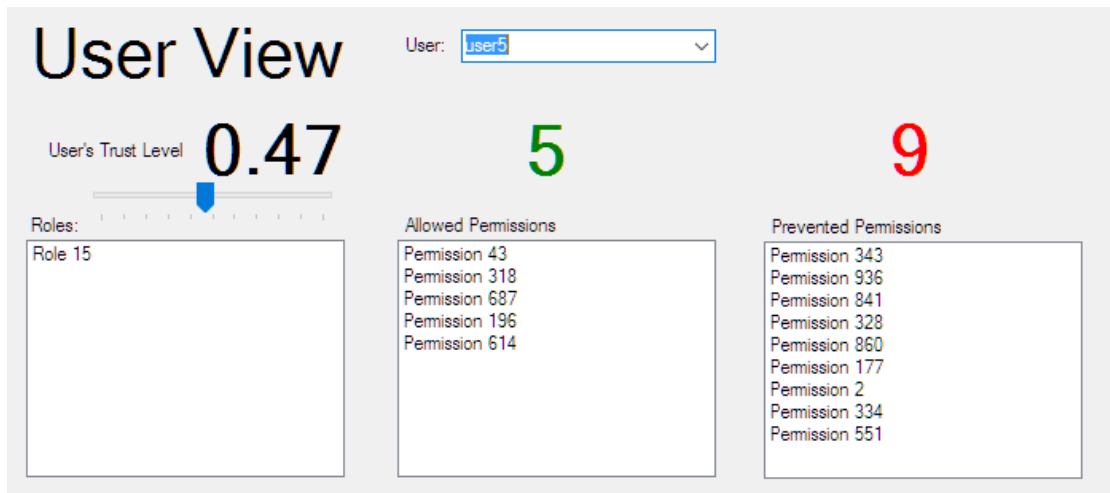
המודל הבסיסי של TDRBAC מצפה שלכל משתמש תהיה רמת אמן אשר על בסיסה תקבע המערכת מה ההרשאות המותרות למשתמש ואילו הרשאות לא יהיו זמינות למשתמש חרף העובדה שהן משויכות לתפקידים המוגדרים למשתמש.

בעת הרצה בסיסית של המודל, ניתן לצפות במסך המציג תמונת מצב של משתמש בודד ולראות את השינויים במערכת בהתאם לשינויים ברמת האמן של המשתמש הבודד.

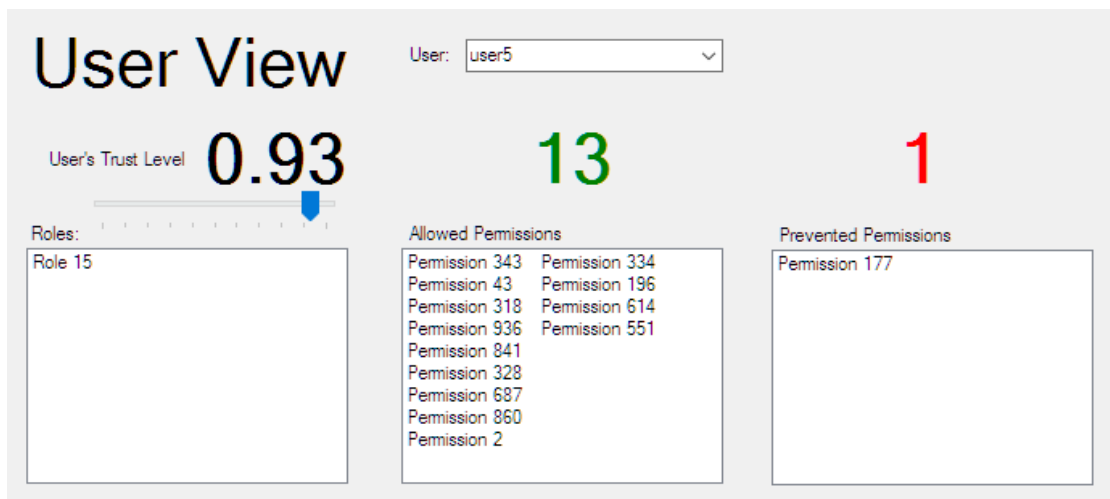
העלאה של רמת האמן תגדיל באופן טבעי את ההרשאות המותרות למשתמש ואילו הורדה של רמת האמן תקטין את מספר ההרשאות הזמינות למשתמש. ניתן לראות כי כאשר רמת האמן הנדרשת לכל הרשאה הינה אקראית, בהתאם למידע באתחול המערכת – שינוי רמת האמן של המשתמש מעלה ומטה, מוסיפה ומסירה הרשאות באופן מדורג ורציף כך ששינוי קטן ברמת ההרשאה של המשתמש מוסיף או מוריד לו מספר קטן של הרשאות בכל תנועה.

ניתן לראות את השפעת השינוי ברמת האמן של המשתמש על ההרשאות הזמינות עבורו במסך תצוגת המשתמש. לדוגמא, משתמש השייך לתפקיד בו הוגדרו 14 הרשאות:

ברמת אמן של 0.47 ניתן לראות כי המשתמש רשאי ל- 5 הרשאות אך אינו נגיש ל- 9 האחרות



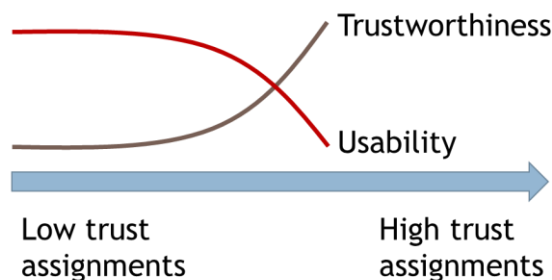
ברמת אמון של 0.93 ניתן לראות כי המשתמש רשאי ל- 13 הרשאות אך אינו נגיש ל- 1 בלבד



ניתן לשנות באופן ידני את רמת ההרשאה ולראות את השינוי בזמן אמת

5 טיוב רמת האמון הנדרש בשיוך הרשאות לתפקידים (RPA)

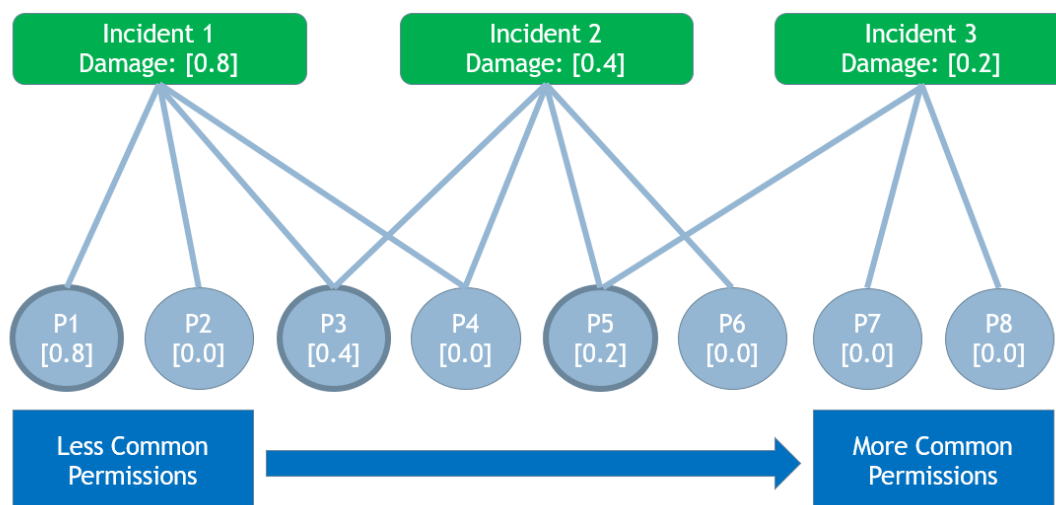
רמת אבטחת מידע של מערכת נמצאת על פי רוב ביחס הפוך למידת השימושיות של המערכת. בדומה לעולם הפיזי, ככל שמיישמים מנגנוני אבטחה הדוקים יותר ישנה השפעה על אופן הפעולה של המשתמשים במערכת ועל מידת החופש שבה הם יכולים לנוע ולפעול בתוכה. בהקבלה למודל ההרשאות TDRBAC, נמצא כי ככל שנקבע רמת הרשאה גבוהה יותר בשיוך של תפקידים להרשאות, כך תעצור המערכת יותר בקשות של משתמשים לביצוע פעולות ולקבלת הרשאות.



מודל TDRBAC מציע אלגוריתם לטיוב רמת האמון הנדרשת כאשר משייכים תפקיד להרשאה. מטרת הטיוב הינה למצוא את האיזון הנכון בין אבטחת מידע לבין "שימושיות". מערכת אשר בה נקבעו רמות הרשאה גבוהות מדי לצורך ביצוע פעולות נפוצות, תהיה מערכת בעלת שימושיות נמוכה אשר תבוא לידי ביטוי בקושי של המשתמשים לפעול בה. מאידך לא נרצה לקבוע רמות הרשאה נמוכות מדי כיוון שאז תהיה המערכת פגיעה ביותר.

מודל TDRBAC מתאר תהליך חישוב אשר בסיומו מציע המודל את הערך המיטבי לשיוך רמת אמון לכל הרשאה. תהליך זה מבוסס על "תקריות" (Incidents). הרעיון מוסבר בהרחבה בעבודת התזה, אולם ניתן לתאר בקצרה באופן הבא:

- תחילה יש למפות אלו תקריות אבטחה יתכנו במערכת ולקבוע לכל אחד מתקריות אלו את מידת הנזק של התקרית והשפעתה על המערכת כאשר 1 מציין תקרית בעלת משמעות "חמורה", ואילו רמת נזק נמוכה תבוא לידי ביטוי ברמה הקרובה או שווה ל-0.
- בשלב שני, יש לקבוע מהן ההרשאות הנדרשות כדי להשלים ולבצע את התקרית. כלומר – מהן אותן הרשאות להן זקוק המשתמש כדי לממש את תרחיש התקרית. על בסיס נתונים אלו, ניתן לצמצם ולהוריד את רמות האמון הנדרשות עבור מרבית רשומות השיוך של תפקיד להרשאה ולהשאיר רמות אמון גבוהות להרשאות בודדות באופן שימנע הפעלה של תקריות אבטחת מידע.
- רמת האמון שתיקבע לכל הרשאה תהיה רמת האמון שתיקבע בשיוך של התפקיד להרשאה בהתאמה.



להלן האלגוריתם הפורמלי של תהליך הטיוב:

1. Sort all Incidents by the value of the damage in descending order, such that Incident with high value of damage is ordered before an Incident with a lower level of damage.
2. Sort all permissions by the value of common in ascending order, such that permission that is less usable is ordered before a more commonly used permission.
3. Initialize all permissions to the default trust value (as determined by the security administrator)
4. Set $i = 0$
5. For each subset of permissions PER(Incident i) if there is no permission with a trust level of Damage(Incident i) or higher, then assign trust level of Damage(Incident i) to the first permission in the subset of permission.
6. Adjust i in 1 and repeat step 5.

את ההשפעה של טיוב רמת האמון ברשומות RPA על המערכת ניתן למדוד בשני פרמטרים:

- "רמת השימושיות" או *Usability Degree*
- תקריות בסיכון

5.1 מדד דרגת השימושיות של המערכת - *Usability Degree*

כאמור, אחד המתחים המוכרים בעולם אבטחת המידע הינו המתח שבין רמת האבטחה לבין רמת השימושיות במערכת – כלומר, כאשר אנו מכילים ומיישמים תכונות של אבטחת מידע במערכת כגון הרשאות, הזדהות, הצפנה וכד', ישנה לרב השפעה על "חופש התנועה" של המשתמשים במערכת ובאופן שבו הם חופשיים לפעול ולבצע את משימותיהם. גישה נכונה לתכנון של פתרונות אבטחת תביא יביא לידי ביטוי את הצורך "להפריע" כמה שפחות להתנהגות הטבעית של המשתמשים במערכת.

דרגת השימושיות הינה נתון המחושב מתוך נתוני המערכת ומציגה את מידת השימושיות של המערכת תחת הגדרות המערכת במצב נתון. מערכת בעלת דרגת שימושיות גבוהה הינה מערכת אשר ככלל, המשתמשים בה חופשיים להשתמש בהרשאות שנתנו להם וקיימת סבירות גבוהה שהמערכת תאשר את בקשות המשתמשים לשימוש בהרשאות, ואילו במערכת בעלת דרגת שימושיות נמוכה קיימת סבירות גבוהה יותר שהמשתמשים יתקלו בסירוב בעת בקשה לעשות שימוש בהרשאה ספציפית.

דרגת השימושיות הינה מספר בטווח של 0 ל-1, כאשר 1 מתאר מערכת אשר בה יש חופש של המשתמשים לעשות שימוש בהרשאות הנדרשות להן, ואילו 0 מתאר מערכת אשר תקשה על המשתמשים לפעול בה באמצעות סירוב לבקשות הרשאה. חישוב דרגת החופש לוקח בחשבון את רמת ההרשאה הנדרשת לכל הרשאה ואת הסתברות השימוש בהרשאה.

לצורך החישוב נצטרך קודם להגדיר מהי ההסתברות לשימוש בהרשאה:

הגדרה:

א. PER_{prop} (Permission Probability) – ההסתברות לשימוש בהרשאה, כלומר עד כמה ההרשאה נמצאת בשימוש במהלך העבודה השוטפת של המערכת. בהמשך אפרט על אופן החישוב של נתון זה, והשפעתו על מדד השימושיות.

דרגת השימושיות מחושבת באופן הבא:

תיאור מילולי לאופן חישוב המדד:

המשלים ל-1 של הממוצע המשוקלל של רמת האמון הנדרשת לכל הרשאה ברשומות RPA, לפי הסתברות השימוש בהרשאה (PER_{prop})

נוסחה פורמלית:

$$Useability Degree = 1 - \frac{\sum_{RPA}(Per_{trust} \cdot Per_{prop})}{\sum_{RPA}(Per_{prop})}$$

הגדרות המשתנים בנוסחה:

- RPA – רשומות השיוך של תפקידים להרשאות
- Per_{trust} – רמת האמון הנדרשת לקבלת ההרשאה, לאחר הרצת תהליך הטיוב
- Per_{prop} – ההסתברות לשימוש בהרשאה

5.1.1 השפעת תהליך הטיוב על מדד ה- $Useability Degree$

בנתוני אתחול מערכת הסימולציה – מדד השימושיות של המערכת הינו **0.502**

Usability Degree
0.502

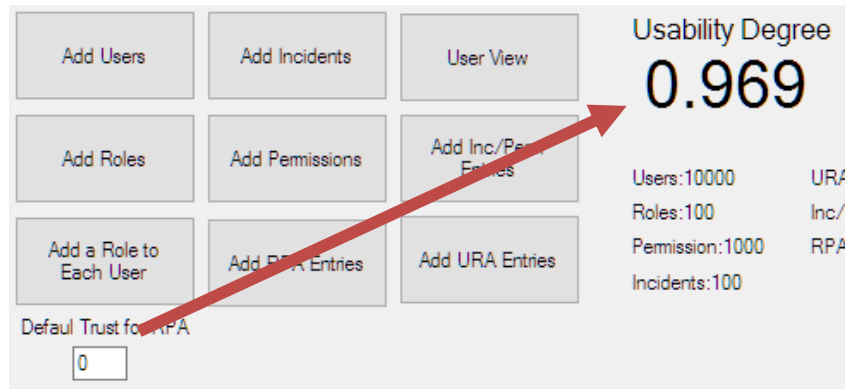
לאחר הרצת תהליך הטיוב של רשומות ה-RPA, וקביעת רמות אמון באמצעות התהליך המוצע במודל, משתנה מדד השימושיות באופן הבא:

כאשר ברירת המחדל נקבעה על 0.2, עלה מדד השימושיות לרמה של - **0.780**.

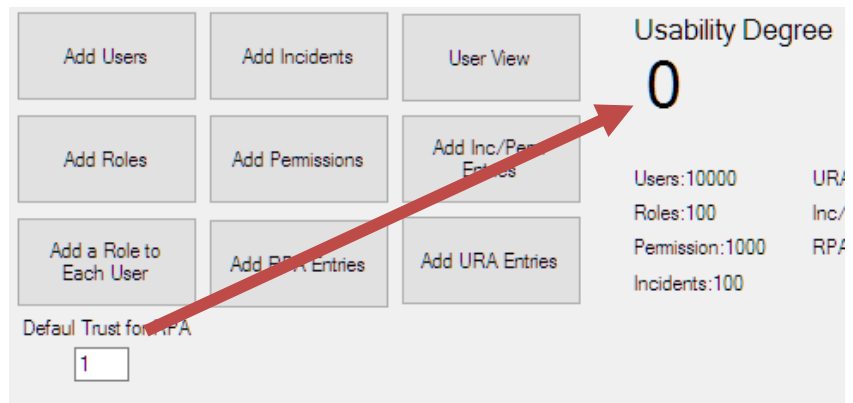
כאשר ברירת המחדל נקבעה על 0, עלה מדד השימושיות לרמה של - **0.969**.

ברירת המחדל נקבעת באופן ידני על ידי מנהל המערכת ומציינת את ערך המינימום של רמת האמון ברשומות RPA, כלומר בשיוך של הרשאה לתפקיד. ראינו כי רמת המינימום משפיעה באופן הפוך למדד השימושיות של המערכת בסיום תהליך הטיוב.

ההסבר לכך מצוי בעובדה כי מרבית ההרשאות מקבלות בפועל את רמת ברירת המחדל אשר נקבעה באופן ידני. בנתוני אתחול מערכת הסימולציה, כאשר קבענו רמת ברירת 0, רק 83 הרשאות מתוך 1000 ההרשאות במערכת, קיבלו ערך הגבוה מ-0. במילים אחרות, כ-92% מההרשאות קיבלו את ערך ברירת המחדל. על פי נוסחת החישוב למדד השימושיות, ההשפעה של הצבת מספרים נמוכים ברמת האמון הנדרשת לקבלת הרשאה (Per_{Trust}) – תניב תוצאה שקרובה יותר ל-1 במדד השימושיות. מנגד, כאשר נציב את רמת ברירת מחדל על הערך המקסימלי – 1, כל ההרשאות יקבלו בפועל רמת הרשאה 1. כתוצאה, מדד השימושיות של המערכת תהיה 0.



כאשר רמת ברירת המחדל נקבע על 0, דרגת השימושיות עמדה על 0.969



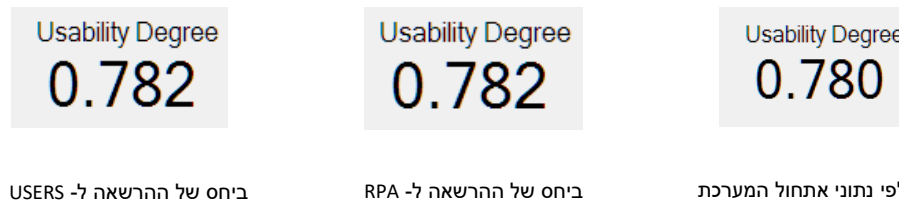
כאשר רמת ברירת המחדל נקבע על 1, דרגת השימושיות עמדה על 0.

5.1.2 השפעת ההסתברות לשימוש בהרשאה (PERprop) על מדד השימושיות

כאמור, ההסתברות לשימוש בהרשאה מתייחס להיקף השימוש בהרשאה. רמת האמון הנדרשת להרשאה אשר השימוש בה הוא גבוה, תהיה רבה יותר מאשר הרשאה אשר השימוש בה נמוך. ישנן מספר דרכים לחשב את ההסתברות לשימוש בהרשאה:

- א. על בסיס היסטוריית השימוש: הדרך המדוייקת ביותר לחישוב ההסתברות היא על בסיס ההיסטוריה של המערכת לאורך זמן מוגדר (למשל שבוע, או חודש אחרונים). במודל שלנו אין אפשרות לחשב את ההיסטוריה ולכן קביעת הנתון מתבצע באופן אקראי באתחול המערכת.
- ב. על פי היחס בין מספר המופעים של ההרשאה בשיוך לתפקידים, למספר השיוכים הכללי (RPA)
- ג. על פי היחס בין מספר המשתמשים המחוברים להרשאה לסך המשתמשים הכללי

מערכת הסימולציה מאפשרת לחשב את היחסים לפי סעיפים ב' ו- ג'. תוצאת השינוי בפועל על רמת השימושיות בהתאם לאופן החישוב של PERprop:



5.2 תקריות בסיכון

תקרית (Incident) נמצאת בסיכון כאשר בקבוצת ההרשאות הנדרשות להפעלתה, רמת האמון הגבוהה ביותר שנדרשת מבין כל רשומות ה- RPA הקשורות היא נמוכה יותר מרמת הנזק של התקרית.

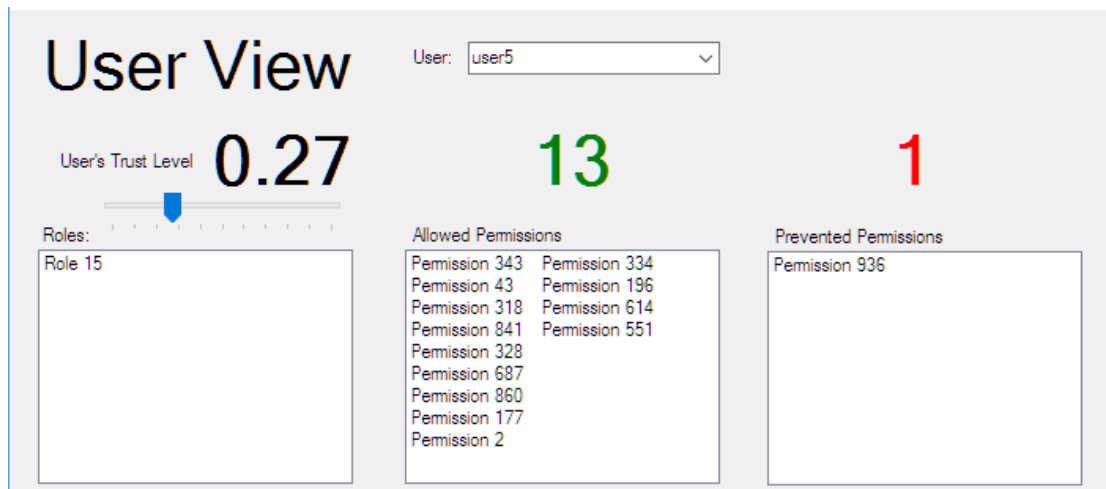
עם הפעלת המערכת ולפני הרצת תהליך טיוב המערכת, נמצאו **12 תקריות בסיכון**. לאחר הרצת תהליך הטיוב, לא נמצאו כלל תרחישים בסיכון. זאת כיוון שהאלגוריתם לטיוב רמות האמון מונע אפשרות שמשמש יוכל להפעיל אירוע אלא אם רמת האמון שלו תואמת את מידת הנזק של התקרית.



5.3 השפעת טיוב רשומות ה- RPA על התנהגות המשתמש ביחס לשינוי רמת ההרשאה

כאמור, במסך תצוגת משתמש, ניתן לשנות באופן ידני את רמת האמון של המשתמש ולראות את הרשאות המשתמש משתנות בהתאם. במסך ניתן לראות אילו הרשאות משויכות למשתמש בהתאם לתפקידים שהוקצו לו, בחלוקה של הרשאות אשר נדחו בשל רמת אמון נמוכה. ראינו כי לפני הרצת תהליך הטיוב, שינוי רמת הרשאה הוסיף והוריד הרשאות באופן רציף ומדורג – כלומר שינוי קטן ברמת ההרשאה הוסיף בדרך כלל מספר בודד של הרשאות. לאחר הרצת תהליך הטיוב, נמצא כי מרבית הרשאות המשתמש יישארו זמינות למשתמש כל עוד רמת האמון שלו גבוהה יותר מרמת המינימום ("ברירת המחדל") שנקבעה בזמן הרצת תהליך הטיוב. בהנחה שנקבעה ברירת מחדל נמוכה (שזהו המצב הרצוי), המשמעות כלפי המשתמש תהיה שמרבית ההרשאות תהינה זמינות עבורו ורק הרשאות בודדות במערכת ידרשו רמת אמון גבוהה.

לדוגמא, על אותו משתמש שהוצג בסעיף 4.6, ניתן לראות כי לאחר הרצת תהליך הטיוב (עם רמת ברירת מחדל של 0.2) המשתמש רשאי להשתמש ב 13 מתוך 14 ההרשאות שנקבעו לו מעצם שיוכו לתפקיד כבר ברמת אמון של 0.27



6 מסקנות

יישום בפועל של המודל התאורטי מחייב התבוננות פנימה אל תוך פרטי המודל, והתמודדות עם התהליכים והאלגוריתמים המוצעים בו. קיומו של יישום כזה, מאפשר ללמוד על תכונות נוספות אשר לא נראו קודם לכן כאשר המודל היה על הנייר בלבד. עם בנייתו והפעלתו של המודל נמצאו תנאים למציאת כלים מדידים לבחינת המודל וצפייה בתוצרים שמוציא המודל בפועל.

תהליך בניית המודל הבסיסי היה פשוט ולא נמצאו אתגרים מהותיים. האובייקטים המרכיבים את המודל, הקשרים בין האובייקטים ומבנה הנתונים, כמו גם תהליכי העבודה, הותאמו למודל המוצע ללא קשיים מיוחדים וללא צורך בהתאמות כלשהן. האתגרים הגיעו בשלב היישום של תכונות מתקדמות במודל TDRBAC.

לצורך המחשה של אחת התכונות המתקדמות המוצעות במודל, בחרתי ליישם את תהליך הטיוב של רמת האמון בשייך של הרשאות לתפקידים (RPA) באמצעות הרעיון של "תקריות". על אף שהאלגוריתם מוסבר בצורה ברורה, נדרשה מחשבה נוספת כדי למצוא את הדרך הנכונה ביותר ליישום, אך לא נדרשו שינויים בבסיס התהליך. עם זאת, דווקא הצגת המשמעויות הייתה אתגר אשר ממנו הופקו שני כלים חדשים: "מדד השימושיות" של המערכת ו-"תקרית בסיכון".

כלים אלו, אשר מוסברים בעבודה זו, לא היו קיימים במודל TDRBAC במקור, והינם תוצר של עבודת היישום. מדדים אלו אפשרו להמחיש כיצד תהליך הטיוב של רמת ההרשאה ברשומות RPA משפיעים על המערכת בפועל והם ישולבו בהמשך בעבודת התזה והיו חלק ממודל ה-TDRBAC.

7 סיכום

בפרויקט זה נפגשו לראשונה התיאוריה והיישום המעשי של מודל ההרשאות TDRBAC. במסגרת הפרויקט פותח כלי אשר מדמה את פעולתה של מערכת מחשב בה יש משתמשים, תפקידים והרשאות כך שמודל ההרשאות מבוסס על המודל המוצע.

המעבר מהמילה הכתובה, ומהאלגוריתמים שעל הנייר אל תוכנת מחשב שעובדת ורצה באופן עצמאי חשף צדדים נוספים של המודל התאורטי והעשיר אותו. הצורך להתמודד עם סוגיות הקשורות בכתיבת קוד, תכנון מבנה נתונים ותהליכי ריצה נועד לבדוק את נכונות המודל, ואת המשמעויות ביישום בפועל. יחד עם זאת, היישום אפשר התבוננות פנימה וחשף פנים אשר קשה היה לראות לפניכן. כך למשל ניתן היה לראות כיצד מנהג המודל אל מול שינוי ברמת ההרשאה של המשתמש לפני הרצת תהליך הטיוב.

בנוסף היישום הוליד רעיונות חדשים אשר העשירו את המודל, כגון מדד דרגת השימושיות. חלקים מן העבודה כאן יילקחו חזרה אל המודל התיאורטי, יוסברו בעבודת התזה.

¹ Sandhu, R., Ferraiolo, D. and Kuhn R. 2004. "American National Standard for Information Technology – Role Based Access Control", ANSI INCITS 359-2004, February 3, 2004.

² Ferraiolo, D. and Kuhn, R. 1992. "Role-based Access Control", Proceedings of 15th NIST-NCSC National Computer Security Conference, Baltimore, MD, 13-16 October 1992, pp. 554-563.

³ Xin Jin, Ram Krishnan, and Ravi S. Sandhu. A unified attribute-based access control model covering dac, MAC and RBAC. In Data and Applications Security and Privacy XXVI - 26th Annual IFIP WG 11.3 Conference, DBSec 2012, Paris, France, July 11-13,2012. Proceedings, pages 41–55, 2012.

⁴ Lavi T. and Gudes E. "Trust-based Dynamic RBAC". In *Proceedings of the 2nd International Conference on Information Systems Security and Privacy - Volume 1: ICISSP*, pages 317-324. (2016)

⁵ Elliott, Aaron, and Scott Knight. "Role Explosion: Acknowledging the Problem." *Software Engineering Research and Practice*. 2010.