תפיסות שגויות נפוצות לגבי Windows Event Logs

מאת דניאל קויפמן

הקדמה

Windows EventLog היא תת-מערכת מורכבת עם מספר רב של אירועי קצה. Windows Event Logs יכולים לתפוס מגוון רחב של פעילויות ולספק לנו תובנות יקרות בנושא המערכות והסביבות שלנו. החלטתי לכתוב את המאמר הזה כי באופן אישי לא מצאתי יותר מדי מקורות מידע שצוללים לעומק של הנושא הזה, ואין ספק שאנשים שמתעסקים ב-IR צריכים לדעת בדיוק למה הם נכנסים.

תפיסות שגויות נפוצות שאני נתקל בהן:

- EventID הוא ייחודי באופן גלובלי
- EventID ממופה באופן ישיר להודעות אירוע
- כל מחרוזות האירוע משומשות בהודעת האירוע
- כל המידע של הודעת האירוע נמצא במחרוזת התבנית
- מחרוזות תבנית נשארות זהות
- XML של אירוע הוא XML1.0 תקין
- אלמנטים של מידע ב-XML של אירוע הם יחודיים
- EventLogs ו-ETW הם שווי ערך

הכנה - אלמנטים של אירוע Windows

כאשר אנחנו רוצים לבדוק כל אירוע שקשור למחשבים, זה חשוב מאוד שנבין את המידע שזמין לנו - וכדי להבין את המידע הזה, אנחנו קודם כל צריכים להבין את הפורמט שבו הוא מוצג. ייתרון אחד של עבודה עם Windows Event Logs הוא שכל האירועים (שנאספים על המערכת עצמה, על תוכנה, או למטרות ביקורת) מסודרים בצורה סטנדרטית כדי לגרום להם להיות קלים להבנה ככל שניתן.

בואו ניקח מבט על האלמנטים המרכזיים שנמצאים באירוע:

- **שם/מפתח הרשומה Log Name/Key** - המפתח מתייחס לסיווג של כל רכיב רישום, תוך ציון שם הלוג שאליו ייכתבו אירועים מרכיבים אלו.
- **רמה / Level** - האם האירוע נרשם רק עבור מטרת יידוע, או שאולי הוא מציין שגיאה קריטית? הרמה של האירוע תגיד לנו מה היא החומרה של האירוע הנרשם. רמות כוללות: error,critical, warning, information, verbose.
- **תאריך ושעה / Date/Time** - מתייחס לתאריך ולשעה של האירוע שנרשם. אם היינו מתחברים למכונה של Windows בשעה 10:03 ביום שני, סביר מאוד שיש אירוע שנרשם שקשור לתאריך ולשעה האלו.
- **מקור / Source** - זהו השם של הרכיב שייצר את האירוע הנרשם. במקרים רבים, זה יהיה השם של התוכנה או של התהליך שרשם את האירוע. לדוגמה, אם האירוע קשור לכשל של בסיס נתונים על המכונה, אז מקור האירוע עשוי להיות השם של האפליקציה שמפעילה את בסיס הנתונים הזה.
- **Event ID** - פריט מידע זה יכול להיות שימושי ביותר עבור מישהו שצריך לבצע פתרון תקלה. ה- Event ID משמש בתור מזהה לאירוע מסויים. המזהה הזה קושר הודעה שמצביעה על סיבת הבעיה, מה שיתן לנו את האפשרות להבין איך לפתור אותה.
- **קטגוריה / Task category** - הקטגוריה משמשת בתור מידע נוסף כדי לעזור לנו עם תקלה של מערכת או תוכנה. המפתחים של תוכנה מסוימת יכולים להגדיר את הקטגוריות כדי לעזור לנו עם קונטקסט על אירוע מסוים.
- **משתמש / User** - שדה זה יכול לציין את המשתמש אשר התחבר למכונה מסוימת בזמן שבו האירוע נרשם. לדוגמה, כאשר מתקינים תוכנה, שם המשתמש יציין את המשתמש אשר ביצע את ההתקנה.
- **מחשב / Computer** - השם של המכונה שרשמה את האירוע.

Event ID הוא ייחודי באופן גלובלי

כנראה התפיסה השגויה הנפוצה ביותר לגבי אירועי Windows היא שהם ייחודיים באופן גלובלי. ממייקרוסופט עצמם:

> "Event identifiers uniquely identify a particular event. Each event source can define its own numbered events and the description strings to which they are mapped in its message file."

מרמז שמזהיי אירועים הם ייחודים רק "Each event source can define its own numbered events" כאשר מדובר במקור אירוע ספציפי (ייחודי באופן מקומי/יחסי). לצערנו, יש הרבה מאוד מקורות שרק מייחסים את ה-Event ID ללא מקור האירוע והגרסה שלו. אם נרצה לאמת זאת בעצמנו, אנחנו רק צריכים לחפש את Event ID 1 באתר MyEventLog. אולי שמתם לב שחוץ ממספר מגוון של מקורות אירוע שונים, יש גם מספר הודעות עבור המקור "WinVNC4". זה עשוי לקרות עקב שוני בין גרסאות של אותו מקור אירוע.

Event ID ממופה באופן ישיר להודעות אירוע

מהמקור "Windows XML Event Log format", סעיף 5.4, Externally stored values אנחנו יודעים שהודעות אירוע נשמרות בקבצי משאב הודעות (message resource files, PE/COFF). הודעות האירוע נשמרות עם מזהה מחרוזת הודעה (message string identifier). ישנן מספר דרכים כדי לקבוע את המזהה הזה מ-Event ID:

- מיפוי אחד לאחד של ה-Event ID למזהה עצמו.
- מיפוי שמשתמש ב-Event ID ובתכונת Qualifiers.
- מיפוי שמשתמש במשאב WEVT_TEMPLATE PE/COFF.

מיפוי שמשתמש ב-Event ID ובתכונות Qualifiers

במקרים מסוימים אנחנו צריכים את תכונת ה-Qualifiers כדי לקבוע את מזהה מחרוזת ההודעה. בואו ניקח לדוגמה את המקרה הבא:

```
<EventID Qualifiers="16384">7036</EventID>
```

הערך של EventID עם ה-Qualifiers מקביל למזהה מחרוזת ההודעה הבא:

```
message (template) string identifier = ( qualifiers << 16 ) | event identifier

16384 in hexadecimal is 0x4000
 7036 in hexadecimal is 0x1b7c
message (template) string identifier = ( 0x4000 << 16 ) | 0x1b7c = 0x40001b7c
```

נשים לב שזה נצפה רק עם מקורות אירוע שמוגדרים תחת "Services\EventLog" ב-registry Windows.

מיפוי שמשתמש במשאב WEVT_TEMPLATE PE/COFF

דרך נוספת שבה ה-EventID יכול להיות ממופה למזהה מחרוזת הודעה היא דרך מידע שנמצא במשאב ה-WEVT_TEMPLATE. לדוגמה, ניקח את המקורות הבאים:

```
Log source      : Microsoft-Windows-OfflineFiles
Identifier      : {95353826-4fbe-41d4-9c42-f521c6e86360}
Log type        : System
Event message files : %systemroot%\system32\cscsvc.dll

Log source              : Microsoft-Windows-BranchCacheSMB
Identifier              : {4a933674-fb3d-4e8d-b01d-17ee14e91a3e}
Event message files     : %systemroot%\system32\cscsvc.dll
```

כפי שניתן לראות, שני המקורות משתמשים באותו קובץ משאבים. אם נחלץ את מזהיי האירוע ומחרוזת ההודעה בעזרת פקודות ה-Powershell הבאות:

```
(Get-WinEvent -ListProvider Microsoft-Windows-OfflineFiles).Events | Format-Table Id, Description


  Id Description
  -- ----------
   1 The Offline Files service started successfully.
   2 The Offline Files service is terminating.
...
2011 Rename of file %1 to file %2 was blocked. The source and/or target file name is an excluded
file type.
(Get-WinEvent -ListProvider Microsoft-Windows-BranchCacheSMB).Events | Format-Table Id,
Description

  Id Description
  -- ----------
3000 SMB BranchCache was enabled with min hash version %1 and max hash version %2.
3001 SMB BranchCache was disabled.
```

ונשווה נגד החלק של משאב ה-PE/COFF בנתיב %systemroot%\system32\en-US\cscsvc.dll.mui:

```
message         : 55
Identifier      : 0xb00007db
data            : Rename of file %1 to file %2 was blocked. The source and/or target file name is
an excluded file type.

message         : 56
identifier      : 0xb1000bb8
data            : SMB BranchCache was enabled with min hash version %1 and max hash version
%2.

message         : 57
identifier      : 0xb1000bb9
data            : SMB BranchCache was disabled.
```

הקובץ הזה מכיל מחרוזות הודעה עבור גם המקור של Microsoft-Windows-OfflineFiles וגם עבור המקור Microsoft-Windows-BranchCacheSMB.

כל מחרוזות האירוע משומשות בהודעת האירוע

בדרך כלל, רשומה של EventLog מציינת מחרוזות באירוע. לדוגמה:

```
<EventData>
    <Data Name="param1">Volume Shadow Copy</Data>
    <Data Name="param2">stopped</Data>
</EventData>
```

כאן, מחרוזות האירוע "Volume Shadow Copy" ו-"stopped" מצוינות בתור EventData ב-XML עצמו. בדוגמה, מזהה הודעת האירוע המקביל הוא "The %1 service entered the %2 state", כאשר מחרוזת האירוע הראשונה צריכה להחליף את מחזיק המקום הראשון (%1) בהודעה הסופית. ההודעה הסופית צריכה להיראות ככה:

"The **Volume Shadow Copy** service entered the **stopped** state."

למרות זאת, יש מקרים שבהם מזהה מחרוזת האירוע לא מכיל את מחזיק המקום עבור מחרוזת אירוע ספציפית. לדוגמה באירוע מספר 2003 במקור Microsoft-Windows-OfflineFiles:

Sync info for %1%nServer copy exists, client copy deleted. %n%13%nSee details for more information.

כפי שאפשר לראות, מזהה מחרוזת האירוע הזה מכיל רק שני מחזיקי מקום עבור המחרוזת הראשונה והמחרוזת השלוש עשרה. הצורה שבה מחרוזות אירוע אחרות צריכות להיות מפורשות במקרים כאלו בהתבסס על ה-EventLog, ללא עובדות נוספות, היא ספקולציה טהורה מנקודת מבט של פורנזיקה דיגיטלית.

כל המידע של הודעת האירוע נמצא במחרוזת התבנית

הודעות אירוע יכולות להשתמש בקונספט שנקרא "parameter expansion". לדוגמה, ההודעה הבאה:

"The %1 call failed for %2 with the following error: %3."

ומחרוזות האירוע הבאות:

```
<EventData>
    <Data Name="param1">ScRegSetValueExW</Data>
    <Data Name="param2">FailureActions</Data>
    <Data Name="param3">%%5</Data>
</EventData>
```

ה-EventLog viewer יתרגם אותם למחרוזת הבאה:

The ScRegSetValueExW call failed for FailureActions with the following error: Access is denied.

כפי שאפשר לראות, "Access is denied" לא מוגדר במזהה מחרוזת ההודעה או במחרוזות האירוע. כאן, %%5 מקביל למחרוזת ההודעה עם המזהה 5 שמאוכסן בקובץ הפרמטרים של ההודעה של מקור האירוע (מקור).

מחרוזות תבנית נשארות זהות

מחרוזת אירוע יכולה להיות שונה עבור גרסאות שונות של קבצי משאבים (PE/COFF). לדוגמה, אם נשווה את מחרוזת ההודעה של אירוע מספר 3000 במקור Microsoft-Windows-BranchCacheSMB בקובץ משאב של cscsvc.dll.mui, בגרסה של Windows 7:

Windows 7, product and file version 6.1.7600.16385: SMB BranchCache was enabled.

ובגרסה של Windows 10 20H2:

Windows 10 20H2, product and file version 10.0.19041.1: SMB BranchCache was enabled with min hash version %1 and max hash version %2.

לא רק שמחרוזת האירוע שונה ב-Windows 10 20H2, יש לה גם מחזיקי מקום. זה לא ברור אם רשומה של אירוע ב-Windows 7 תכיל את מחרוזות האירוע שיהיו מצופות במערכת של Windows 10 20H2.

XML של אירוע הוא XML1.0 תקין

למרות שמייקרוסופט בעצמה מרמזת ש-XML של אירוע הוא בפורמט XML1.0 תקין, בפועל לא כך המקרה. דוגמה למקרה שבו רשומת אירוע חורגת מפורמט XML1.0 תקין:

```
<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
  <System>
    <Provider Name="Microsoft-Windows-Store" Guid="{9C2A37F3-E5FD-5CAE-BCD1-43DAFEEE1FF0}"/>
    <EventID>8001</EventID>
    <Version>0</Version>
    <Level>4</Level>
    <Task>8001</Task>
    <Opcode>14</Opcode>
    <Keywords>0x8000100000000000</Keywords>
    <TimeCreated SystemTime="2018-09-03T01:54:01.996300600Z"/>
    <EventRecordID>218351</EventRecordID>
    <Correlation/>
    <Execution ProcessID="784" ThreadID="4980"/>
    <Channel>Microsoft-Windows-Store/Operational</Channel>
    <Computer>base-wkstn-01.shieldbase.lan</Computer>
    <Security UserID="S-1-5-19"/>
  </System>
  <EventData>
    <Data Name="Message">[1374] [000001B9730DC1F0:558F5D32-0827-EB7B-6AD6-D5DB4138B3AA]
StopProcessing : Key:NegotiatingUnleased</Data>
    <Data Name="Function">StateMachine<class KeyMachine,class KeyState>::PumpEvents</Data>
    <Data Name="Source">onecoreuap\enduser\winstore\licensemanager\lib\statemachine.h</Data>
    <Data Name="Line Number">173</Data>
  </EventData>
</Event>
```

כאן אלמנט ה-Data המודגש איננו בפורמט XML תקין, כי XML תקין לא אמור להכיל את התו < באלמנט מידע.

אלמנטים של מידע ב-XML של אירוע הם ייחודיים.

כפי שאפשר לראות ב-XML הבא, שמות של אלמנטים של מידע ב-EventData אינם יחודיים. ישנם מספר אלמנטים של מידע שיש להם את השם "IoCount", "TotalLatencyUs", "TotalBytes" ו-"IoTypeIndex":

```
<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
  <System>
    <Provider Name="Microsoft-Windows-Ntfs" Guid="{3FF37A1C-A68D-4D6E-8C9B-F79E8B16C482}"/>
    <EventID>146</EventID>
    <Version>1</Version>
    <Level>4</Level>
    <Task>0</Task>
    <Opcode>0</Opcode>
    <Keywords>0x4000000000200000</Keywords>
    <TimeCreated SystemTime="2018-09-05T01:51:32.107675800Z"/>
    <EventRecordID>3291</EventRecordID>
    <Correlation/>
    <Execution ProcessID="4" ThreadID="15136"/>
    <Channel>Microsoft-Windows-Ntfs/Operational</Channel>
    <Computer>hostname</Computer>
    <Security UserID="S-1-5-18"/>
  </System>
  <EventData>
    <Data Name="VolumeCorrelationId">{4994EA25-0000-0000-0000-501F00000000}</Data>
    <Data Name="VolumeNameLength">2</Data>
    <Data Name="VolumeName">C:</Data>
    <Data Name="IsBootVolume">true</Data>
    <Data Name="HighIoLatencyCount">0</Data>
    <Data Name="IntervalDurationUs">3750962784</Data>
    <Data Name="NCReadIOCount">206</Data>
    <Data Name="NCReadTotalBytes">9112304</Data>
    <Data Name="NCReadAvgLatencyNs">11765267</Data>
    <Data Name="NCWriteIOCount">2858</Data>
    <Data Name="NCWriteTotalBytes">36175488</Data>
    <Data Name="NCWriteAvgLatencyNs">4904584</Data>
    <Data Name="FileFlushCount">560</Data>
    <Data Name="FileFlushAvgLatencyNs">14539870</Data>
    <Data Name="VolumeFlushCount">0</Data>
    <Data Name="VolumeFlushAvgLatencyNs">0</Data>
    <Data Name="FileLevelTrimCount">0</Data>
    <Data Name="FileLevelTrimTotalBytes">0</Data>
    <Data Name="FileLevelTrimExtentsCount">0</Data>
    <Data Name="FileLevelTrimAvgLatencyNs">0</Data>
    <Data Name="VolumeTrimCount">38</Data>
    <Data Name="VolumeTrimTotalBytes">8957952</Data>
    <Data Name="VolumeTrimExtentsCount">303</Data>
    <Data Name="VolumeTrimAvgLatencyNs">53391</Data>
    <Data Name="IoBucketsCount">44</Data>
    <Data Name="TotalBytesBucketsCount">40</Data>
    <Data Name="ExtentsBucketsCount">1</Data>
    <Data Name="IoCount">59</Data>
    <Data Name="IoCount">21</Data>
    ...
    <Data Name="IoCount">13</Data>
    <Data Name="TotalLatencyUs">151222351</Data>
    <Data Name="TotalLatencyUs">252569191</Data>
    ...
    <Data Name="TotalLatencyUs">2073327422</Data>
    <Data Name="TotalBytes">2813952</Data>
    <Data Name="TotalBytes">884736</Data>
    ...
    <Data Name="TotalBytes">8957952</Data>
    <Data Name="TrimExtentsCount">303</Data>
    <Data Name="IoTypeIndex">0</Data>
    <Data Name="IoTypeIndex">1</Data>
    ...
    <Data Name="IoTypeIndex">7</Data>
  </EventData>
</Event>
```

EventLogs ו-ETW הם שווי ערך

EventLog מהווה סטנדרט לתוכנות ולמערכת ההפעלה לרשום אירועי תוכנה וחומרה חשובים, עבור המטרה של אדמיניסטרציית מערכת.

ETW זה קיצור של Event Tracing for Windows. ETW הוא מנגנון מעקב שעובד ברמת הקרנל שנותן לנו אפשרות לרשום אירועי קרנל או תוכנה. נשים לב שקובץ אירועים בקונטקסט של ETW לרוב יהיה בפורמט etl. שהוא שונה מפורמט של קובץ EventLog, שהוא evt. או evtx.

סיכום

בתור אנשים שמבצעים חקירות פורנזיות, אנחנו צריכים להבין את האינפורמציה ואת מקרי הקצה של המידע שאנחנו מנתחים. Windows EventLog היא תת-מערכת מורכבת עם מגוון מקרי קצה, שעל רובם לא מדברים בקהילות פורנזיות, אז אני מקווה שמאמר זה שפך קצת אור על הנושא ופתח דלתות למחקר עתידי יותר מעמיק.

על הכותב

דניאל קויפמן, בן 24, מהנדס מערכות SIEM ו-Detection, ובעל זיקה חזקה לעולמות ה-DFIR. ניתן ליצור קשר בלינקדאין, טוויטר (@KoifSec) או במייל: koifmandani@gmail.com