Implementation of Behavioral Indicators in Threat Detection and User Behavior Analysis

Yegor Anashkin [1] and Marina Zhukova [1]

[1] *Reshetnev Siberian State University of Science and Technology, 31 Krasnoyarskii rabochii prospekt, Krasnoyarsk, 660037, Russia*

Abstract

This paper considers the evolutionary path of indicator development in the tasks of monitoring and threat detection. The work aims to form a unified descriptive structure for behavioral indicators. The resulting description standard is designed to create an open database of behavior indicators. The base of behavior indicators shall be the basis for the user action profiling system that's developing by the authors. Prospects of application of the obtained results are also seen by the authors in the field of Threat Hunting, Threat Intelligence, and automation of correlation rules for SIEM systems.

In addition, the possibilities, benefits, and methods of implementation of behavior indicators in the process of user actions profiling are considered.

Keywords

Indicators, IoC, IoA, IoB, Threat Hunting, UBA, user behavior analytics

1. Introduction

The opposition to security threats is a permanent task of information security. It is a continuously and difficult process, that's consists of the next subprocesses:

- identification and analysis threats
- development protection system against cyber threats
- monitoring attempts to implement threats
- response against attempts to implement threats
- analysis and conclusions based on results of response
- implementation of corrective/improvement measures

We focused on the monitoring process because it is the key process required to detect successful attacks or attempts to implement cyber threats.

A classic and widespread approach to monitoring is a triggered approach (alert-driven). With this approach, detection and response against attempts to implement threats occur after the information protection means are triggered [1]. This approach cannot be called sufficient against modern cyber threats. Attackers are actively modernizing their techniques and tools to bypass existing information security systems.

Another more mature approach to monitoring cyber threats is Threat Hunting. This term should be understood as the process of cyclical analysis of telemetry collected from the infrastructure in order to identify successfully implemented threats that were not noticed by preventive information security systems deployed in the infrastructure [1]. The use of Threat Hunting can reduce the time from the moment an attacker penetrates the victim's infrastructure to the moment he is detected [2].

Today, in both approaches to monitoring cyber threats, you can find the use of indicators. Let us define the concept of an indicator in information security as a sign that signals the presence of realized

AISMA-2021: International Workshop on Advanced in Information Security Management and Applications, October 1, 2021, Stavropol, Krasnoyarsk, Russia

EMAIL: a.yegoriy@gmail.com (Yegor Anashkin); zhukova@sibsau.ru (Marina Zhukova)
ORCID: 0000-0003-1696-0965 (Yegor Anashkin); 0000-0003-3441-3041 (Marina Zhukova)

threats or attempts to implement threats. When classifying indicators in information security, three types of indicators can be distinguished:

- indicator of compromise (IoC)
- indicator of attack (IoA)
- indicator of behavior (IoB)

2. Indicators of Compromise

Today, indicators of compromise (IoCs) are the most widely used. An Indicator of Compromise (IoC) is an object observed on a network or an endpoint, that is highly likely to indicate unauthorized access to the system (that is, its compromise) [3]. These indicators are used to detect malicious activity at an early stage, as well as to prevent known threats. Popular types of IoC are IP addresses, DNS names, and file hashes.

However, IoCs have not become a complete and sufficient solution for detecting all attempts to implement threats. The major shortcomings of compromise indicators are highlighted [4]:

- Professional attackers who conduct targeted attacks either develop new tools or modify known hacker tool signatures, such as mimikatz. Due to their uniqueness, such tools are not detected by indicators.
- Possibility of flooding databases with indicator noise. Attackers send a lot of false indicators, due to which professionals need to filter indicators. It also leads to a decrease in the informativeness of the indicators. It also leads to a decrease in confidence in the indicators.
- Professional attackers use the «fileless» malware technique [12]. In this technique, the malicious file is not delivered to the victim's device but is built on the end-device by downloading the malicious code through standard OS features, such as PowerShell.
- Generally, IoCs are used in reactive mode. It means a successful attack is discovered when IoCs are found out in forensic artifacts. Thus, IoCs are instruments to identify compromising, but not to provide proactive protection.

In summary, the use of IoCs can help detect attacks in which attackers use already known objects (files, DNS, IP, etc.). However, IoCs remain powerless against modern targeted attacks. This led to the emergence and application of a new type of indicators – Indicator of Attack (IoA) and Indicator of Behavior (IoB).

3. Indicators of Attack

An Indicator of Attack is a rule (chain of actions) containing a description of suspicious behavior in the system, which may be a sign of a targeted attack [5]. To understand the IoA, refer to the Lockheed Martin Kill Chain Model [6] and the ATT&CK [7]. The Kill Chain model clearly shows a clear breakdown of an attacker's actions into a sequence of stages to achieve a set goal. The MITRE Knowledge Base is the structured and most comprehensive knowledge base of the tactics, techniques, and procedures of professional attackers.

Thus, an indicator of an attack can be a separate technique/procedure (for example T1562.002 Impair Defenses: Disable Windows Event Logging), or a sequence of techniques used within the framework of related tactics. As an example, consider running the command line (T1059 Windows Command and Scripting Interpreter) followed by modification of the registry keys responsible for autostart to persistence into the system (T1547.001 Boot or Logon Autostart Execution: Registry Run Keys / Startup Folder).

The use of attack indicators (IoAs) to detect attempts to implement cyberthreats is more effective than IoCs because changing TTPs (Tactics, Techniques, and Procedures) is the most difficult thing for an attacker to do [8].

4. Indicators of Behavior

An Indicator of Behavior is a digital behavior monitored to understand risks within an organization [9]. A set of behavioral indicators (IoBs) includes a subset of actions from the attack indicators. The main difference between IoAs and IoBs is:

- IoAs are more related to TTPs (Tactics, Techniques, and Procedures) of professional attackers (APTs). In turn, IoBs are signs of potentially dangerous behavior.
- IoBs can be used to detect an internal intruder, an insider or a user who disregards established security policies.

The following are examples of behavioral indicators:

- use of external media
- work on multiple hosts
- remote login
- work with system utilities
- use of RATs (Remote Admin Tools)

Indicators of behavior, therefore, have a broader scope of coverage. Behavioral indicators are applicable in the detection of internal intruders, insiders, breaches or non-compliance with established information security policies, leaks of confidential information, and others.

5. Integration of indicators

Specialized solutions called the Threat Intelligence Platform are used to integrate IoCs into the threat detection process [18]. Threat Intelligence Platform is able to collect the information about possible threats from different sources (commercial and free, closed and open, public and private) in real-time, classify it, and perform various operations with it, including uploading it to the information security tools. A typical diagram of such a solution is shown in Figure 1.

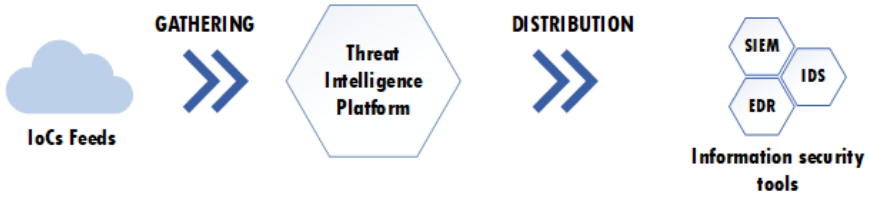

Figure 1: IoCs integration scheme

In turn, the attack indicators and behavior indicators currently come as paid rule sets when you purchase the product [5][10]. Open databases, as in the case of IoCs, are not developed. In addition, full implementation of IoBs requires specific tools for profiling user actions. A user profile should be built, including the user's IoBs, and each user action should be recorded and compared with the database of IoBs. Therefore, if IoBs and IoAs are integrated, the diagram shown in Figure 1 will change to look like Figure 2.

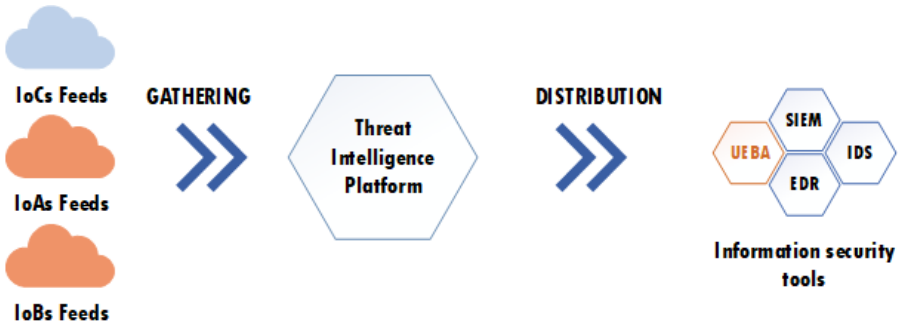

Figure 2: IoBs and IoAs integration scheme

Due to the lack of open IoBs databases, a general structure for describing behavioral indicators is being developed as part of the work in order to create and populate the IoBs database. In the future, it is planned to use this database of indicators in our own system for profiling user actions, which is being developed [11].

An example XML description of a behavior indicator is given in Listing 1.

```
<IOB>
    <Id>000001</Id>
    <Name> Using the Windows command line </Name>
    <Description>Command line usage may indicate an attempt to execute a system command to
run scripts, change system configuration, retrieve system information, etc. Not all users need
to interact with the command line when performing their work tasks.</Description>
    <Priority>Medium</Priority>
    <Category>Policy Violation, Improper Use</Category>
    <MITRE_TACT>Execution</MITRE_TACT>
    <MITRE_TECH>T1059</MITRE_TECH>
    <Standalone_IOA>true</Standalone_IOA>
    <BehaviorOn>Windows</BehaviorOn>
    <Detection>
        <Detector id="1">
            <LogSourceName>Windows Security Log</LogSourceName>
            <EventID>4688</EventID>
            <Parameter>NewProcessName</Parameter>
            <Condition>Contains</Condition>
            <Value type="string">
                Windows\System32\cmd.exe
            </Value>
        </Detector>
    </Detection>
</IOB>
```

Listing 1: Example description of a behavior indicator

Semantically, the structure of the IoB description can be divided into two components: a block with the necessary descriptive characteristics of the indicator and a block with information to detect the indicator. The description and purpose of the fields are shown in the Table 1.

Table 1
Description of the IoB fields

Field name	Description
Id	Unique identifier of the behaviour indicator
Name	Name of behaviour indicator
Description	Brief description of behaviour indicator
Priority	Priority of behaviour indicator
Category	Category of behaviour indicator
MITRE_TACT	Display of the behaviour indicator in MITRE base tactics
MITRE_TECH	Display of the behaviour indicator in the MITRE base technique
Standalone_IOA	Field shows if the behaviour indicator can be considered as a separate attack indicator
BehaviorOn	This field shows where the behavior indicator can be observed: on the Windows host, on a network or on a Linux host.
Detection	The field includes detectors that can be used to detect an IoB
Detector	The field includes the necessary data to detect IoB: in which source to watch, which field and which value.
LogSourceName	Event source name
EventID	Identifier of the event in the event source system
Parameter	Parameter of the event to analyze
Condition	Condition that must be met by the parameter
Value	Value for condition

The resulting behavior indicator description structure includes not only descriptive fields but also typical event sources, fields, and their values required for indicator detection. This feature allows the use of the IoBs database to automate the writing of correlation rules of SIEM systems.

6. Implementing behavioral indicators in user behavior analysis (User Behavior Analytics)

User Behavior and Entity Analytics is a class of information security tools for detecting threats to information systems, based on the analysis of user, device, application, and other behavior [13].

In today's User Behavior Analytics/User Behavior and Entity Analytics solutions, the Scoring method or Scoring models (counting the value, in information security, this is counting the value of the risk) are mostly found [14, 15].

This approach is combined with the time decay method. This means that when a user stops taking actions that add negative points to their risk score, the risk score will gradually decrease, e.g. every 5 minutes by 10 points. Thus, this approach is not sensitive to time-distributed attacks. This approach also lacks retrospective analysis.

Behavioral indicators can be used to build both retrospective graphs of potential actions that preceded the current behavior and predict graphs of future actions. Therefore, their use allows for decoupling from the time frame. Detection should not depend on the frequency of potentially dangerous actions but on the sequence of such actions.

Also, the quality of UEBA class solutions is highly dependent on the number of data sources used to enrich actions with context [16]. Data enrichment allows finding deeper connections. For example, if integrated correctly with the helpdesk, UEBA can eliminate false positives related to the execution of applications by administrators on users' hosts. Thus, the number and quality of sources connected and processed directly affect the accuracy figure (false positive rate).

Hence, a direct way of improving UEBA class solutions is to work on parsing all sorts of existing data sources, natural language text processing, etc.

The authors have chosen a different direction – increasing the number of models used.

In order to reduce false positives and increase the number of scenarios for the use of user action profiling, a multi-model approach was previously proposed. The multi-model approach, as originally conceived, consisted of the following models

- a user behavior model
- a working behavior model
- a security behavior model
- a model of a potential attacker

Previously, the multi-model approach was based on analyzing the sequence of all user actions. However, the main purpose of this class of solutions is to detect malicious intent in the user's actions. To detect malicious intent, behavior indicators and attack indicators are sufficient. Therefore, let us now consider the transformation of each model with the implementation of the behavior indicators.

6.1. The user behavior model

The user behavior model consists of a set of characteristics of the infrastructure with which the user interacts (e.g. IP address and work hostname) and a set of behavior indicators. The set of behavior indicators generated by user action profiling is primarily designed to avoid false positives.

Let's look at a specific example. Let's take an internal attacker as the subject. It is assumed that the internal attacker already has initial access to the system as opposed to the external attacker. However, an internal attacker may use his/her colleague's account to elevate his/her rights or hide his/her actions. To detect such attempts, let's introduce an appropriate behavior indicator - logging in under someone else's account. The entered indicator will work based on the work host specified in the user's profile. When a user logs in to a host they have never logged in to before - the system considers this behavior a possible indicator of logging in under someone else's account.

Reflecting on this behavioral indicator, it is possible to conclude that there are scenarios with false positives. For example, a system administrator or helpdesk employee may log on to users' hosts to resolve technical problems. Therefore, to avoid false positives, these user roles need to have another behavioral indicator in their profile - operating on multiple hosts.

6.2. The potential attacker model

The potential attacker model is the most significant in terms of the threat posed. This model is therefore subject to particularly stringent false positives. A solution to this requirement could potentially be to set it to trigger only when a specific sequential chain of behavior indicators is detected. The fixation of behavior indicators relating to different stages of an attack is a tell-tale sign of malicious behavior on the part of the user. This concept can be represented as an IoB matrix:

$$IoB = \begin{bmatrix} b_{11} & \cdots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{n1} & \cdots & b_{nn} \end{bmatrix}, \quad (1)$$

where b_{ij}=1 if it is possible to move from stage IoB(i) to stage IoB(j).

Then, for example, putting L = 3 as the chain length required to trigger an alert, consider a case study. Introduce a behavior indicator, "use of external media", which corresponds to the Initial Access tactic of the MITRE matrix. The current chain length is 1. Next, we notice the "launching a program from removable media" indicator, which correlates with the Execution tactic. From the 'use of external media' indicator, it is possible to move to the "launching a program from removable media" indicator, so the chain length becomes 2. The next indicator observed is 'change in registry values associated with autorun'. This indicator is related to the Persistence tactic. The observed indicator can be associated with the previous one, the chain length becomes 3. Chain length reaches a threshold value - an alert is generated.

In addition, with this matrix, it is possible not only to detect current events but also to predict expected indicators of behavior in the future. An example of such a predictive chain is shown in Figure 3.

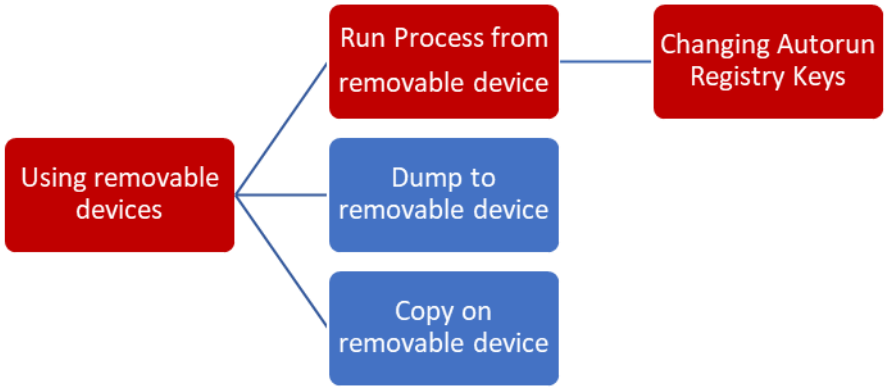

Figure 3: An example of a predictive chain of behavior indicators

6.3. The working behavior model

The working behavior model aims to reduce false positives associated with specific infrastructure and corporate policies.

For example, in some organizations the use of remote administration tools is legitimate, in others, it is not. Therefore, if the activity is legitimate, appropriate behavioral indicators should be added to the working model.

Examples of corporate behavioral indicators are:

- use of remote administration tools (RAT)
- use of telnet
- use of public file repositories

What is the underlying assumption for the effectiveness of this behavioral indicator approach? Three popular models related to attacker behavior are considered:

- The Kill Chain model by Lockheed Martin [6]
- MITRE ATT&CK matrix [7]
- DIAMOND model [17]

The Kill Chain model does a good job of showing the sequence of actions in an attacker's actions to achieve their goals.

The MITRE database is rich in techniques that are indeed capable of being indicators of malicious intent, as they are highlighted by analyzing the actions of multiple professional groupings (APTs).

The Diamond model shows that infrastructure features and capabilities (analogous to techniques) can identify a specific attacker (attacker attribution).

Thus, the multi-model approach combines the best practices of the three models for analyzing user behavior. The potential attacker model is based on the consistency principle of the Kill Chain model. To cover the behavior of professional attackers, the behavior indicators incorporate MITRE matrix techniques. The user behavior model adopts the Diamond model's experience of identifying a subject by infrastructure attributes and user capabilities (behavioral indicators).

Despite the perceived benefits of using best practices, the disadvantages cannot be overlooked:

1. The listed models (Kill Chain, MITRE, DIAMOND) target external attackers. To fully cover the sources of cyber threats, models need to be expanded and adapted to also target the internal attacker. As an example, Initial Access tactics from the MITRE base may be completely redundant for an internal attacker because the internal user has a priori certain access rights.
2. More relevant and precise points of contact between the behavioral indicators are needed. Building attack chains (transitions between indicators) on the basis of the attack tactics stage alone will potentially have errors of the first kind. It means detecting an attack attempt based on potentially consistent behavioral indicators, which actually come from different sources and are not related to each other. The presence of false-positive verdicts creates the need for additional manual analysis.

7. Conclusions

The next stage in the development of cyber threat monitoring and detection is the integration of behavioral indicators and attack indicators into this process. This requires not only the availability of specific tools but also the emergence of open and accessible indicator databases. To this end, attempts have been made to develop a descriptive framework of behavioral indicators to further build and populate the primary indicator base. This database will be used in its own system for profiling user actions. In addition, it is planned to place the database of indicators in the public domain, which will allow the community to use this database, for example, for the automated creation of correlation rules for SIEM systems.

In order to improve the quality of UEBA class solutions, the idea of implementing behavioral indicators in the behavior analysis is proposed. The advantage of using behavioral indicators is their focus on malicious intent. Behavioral indicator sequence analysis is able to detect time-distributed attacks, unlike the popular Scoring method.

Taking into account the implementation of behavior indicators and the basic ideas of the Kill Chain, MITRE, and DIAMOND models, an early multi-model approach to user action profiling has been redesigned. Further work will focus on developing algorithms for modeling and detecting behavioral indicators from different log events.

8. References

[1] Cyber Polygon, Threat Hunting. Why might you need it, 2020. URL: https://cyberpolygon.com/materials/threat-hunting-why-might-you-need-it/.

[2] SANS, Threat Hunting Survey: The Differing Needs of New and Experienced Hunters, 2019. URL: https://www.sans.org/media/analyst-program/2019-threat-hunting-survey-differing-experienced-hunters-39220.pdf.

[3] Encyclopedia by Kaspersky, Indicator of Compromise (IoC). URL: https://encyclopedia.kaspersky.com/glossary/indicator-of-compromise-ioc/.

[4] S. Curry, Indicators of Behavior: The New Telemetry To Find Advanced Cyber Attackers, 2019. URL: https://www.forbes.com/sites/samcurry/2019/06/27/indicators-of-behavior-the-new-telemetry-to-find-advanced-cyber-attackers/?sh=2d7920e4193e.

[5] Kaspersky Anti Targeted Attack Platform, Using indicators of compromise (IOC) and attack (IOA) for Threat Hunting. URL: https://support.kaspersky.com/KATA/3.7/en-US/194907.htm.

[6] E. Hutchins, M.J. Cloppert, R. M. Amin. "Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains", Lockheed Martin Corporation, 2010. URL: https://www.lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/LM-White-Paper-Intel-Driven-Defense.pdf.

[7] MITRE ATT&CK®. URL: https://attack.mitre.org/.

[8] David J. Bianco. The Pyramid of Pain, 2013. URL: https://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html.

[9] A. Ross, Indicators of Behavior (IOBs) – With 2020 Vision, Forcepoint, 2020. URL: https://www.forcepoint.com/blog/x-labs/indicators-of-behavior-iob.

[10] Forcepoint. Dynamic User Protection. URL: https://www.forcepoint.com/product/dynamic-user-protection.

[11] Yegor V. Anashkin, Marina N. Zhukova. "About the System of Profiling User Actions Based on the Behavior Model" IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), St. Petersburg, Russia 2021. doi: 10.1109/ElConRus51938.2021.9396158.

[12] Microsoft Docs, Fileless threats, 2021. URL: https://docs.microsoft.com/en-us/windows/security/threat-protection/intelligence/fileless-threats.

[13] Encyclopedia by Kaspersky, UEBA (User and Entity Behavior Analytics). URL: https://encyclopedia.kaspersky.com/glossary/ueba/.

[14] J. Wang. Deep Learning in Security – An Empirical Example in User and Entity Behavior Analytics (UEBA), Spark Summit, Video, 2017. URL: https://databricks.com/session/deep-learning-in-security-an-empirical-example-in-user-and-entity-behavior-analytics-ueba.

[15] Derek Lin, Leonid Kladko, User Behavior Analytics for Cyber Security and Its Implementation In Scala, ScalaUA Conference, Video, 2018. URL: https://www.scalaua.com/2018/03/22/user-behavior-analytics-for-cyber-security-and-its-implementation-in-scala-derek-lin-leonid-kladko/.

[16] G. Sadowski, A. Litan, T. Bussa, T. Phillips, Market Guide for User and Entity Behavior Analytics: analytical report, Gartner, 2018. URL: https://www.cbronline.com/wp-content/uploads/dlm_uploads/2018/07/gartner-market-guide-for-ueba-2018-analyst-report.pdf.

[17] S. Caltagirone, A. Pendergast. The Diamond Model of Intrusion Analysis, Center for Cyber Threat Intelligence and Threat Research, Hanover, MD, 2013, 61 P. URL: https://www.activeresponse.org/wp-content/uploads/2013/07/diamond.pdf.

[18] Cyberpedia, What is a Threat Intelligence Platform, Palo Alto Networks. URL: https://www.paloaltonetworks.com/cyberpedia/what-is-a-threat-intelligence-platform.