Threat Intel Processing at Scale

Don Franke

Threat Intel Processing at Scale

GIAC (GCTI) Gold Certification

Author: Don Franke, franke.don@gmail.com
Advisor: Rajat Ravinder Varuni
Accepted: 3/27/2019

Abstract

This paper examines the common but flawed practice of implicitly assigning trust to threat indicators (or "intel") that are shared by external providers. These indicators are often deployed automatically to security controls without adequate vetting, resulting in false positives and a false sense of security. This paper proposes a solution for how to implement an intel analysis process that separates noise from useful indicators, can handle a large volume of information received regularly and is scalable despite limited analyst resources.

1. Introduction

There is no shortage of threat intel feeds that provide IP addresses, domain names, and other indicators of compromise (IOCs) associated with a security event or threat actor. While the exchange of information is encouraged, the volume of data consumed can be overwhelming to process effectively, especially for smaller security teams. Inadequately vetting of IOCs in advance of their use in security controls like firewalls, proxies and event monitoring can result in false positives that end up requiring additional resources to resolve.

The intel analysis process can be time-consuming and resource intensive. Most organizations cannot afford a dedicated analyst function focused on processing the information received from threat indicator feeds. This paper introduces a process that reduces the amount of time required to adequately assess indicators consumed by a provider. Established non-cyber intelligence analysis concepts form the foundation of the proposed solution. The goal of this approach is to remove subjectivity and apathy and offer a method that scales to accommodate a large volume of indicators with only limited analyst resources to do the work.

The SANS Institute class FOR578 taught by Robert M. Lee (SANS, 2018) inspired this paper. This class is highly recommended for anyone interested in learning how to utilize threat intel for information security purposes effectively. The structure of this paper follows the intel lifecycle (Pokorny, 2018) and has the following sections:

- Direction - The objectives of the analysis process are defined.
- Collection – Consumers receive indicators from providers.
- Analysis - Intel is separated from indicators using a repeatable process.
- Production - Intel is utilized and deployed to security controls.
- Dissemination - Organizations share their intel with other organizations.

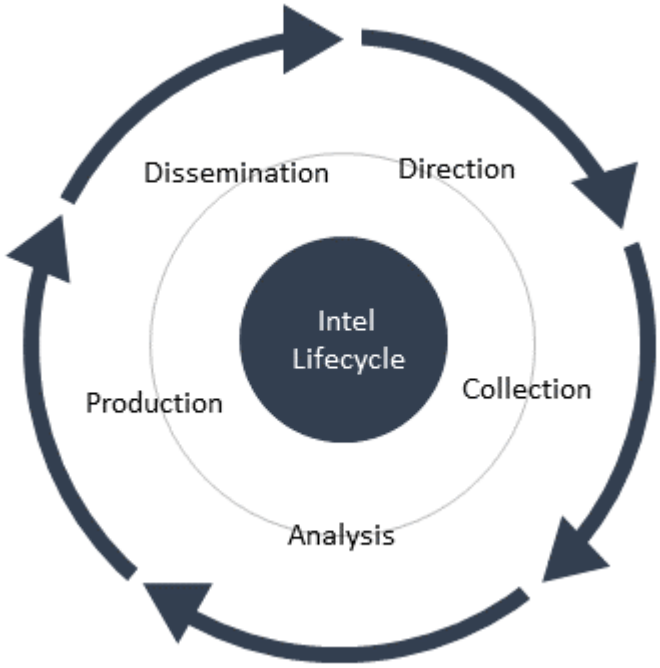

Figure 1: Overview of an organization consuming indicators from a provider, then utilizing and sharing them.

1.1. What is Intel?

In the context of information security, the term "intel" has become overloaded almost as much as the term "cyber." This paper and the proposed solution defer to the original meaning of intel: it is both a process and a product. While the terms intel, indicators, and threat information are often interchangeable, this paper uses a definition of intel that is modified from the Central Intelligence Agency (CIA) paper titled "A Definition of Intelligence" (Bimfort, 1995):

> *"Intel [is] threat indicators that have been determined to have value by being relevant and useful to the consuming organization."*

The goal of the analysis process offered in this paper is to separate intel (useful indicators) from the noise, which supports "the aim of intelligence [which] is to give insight into a specific, concrete event" (University of Groningen, 2005, p.51).

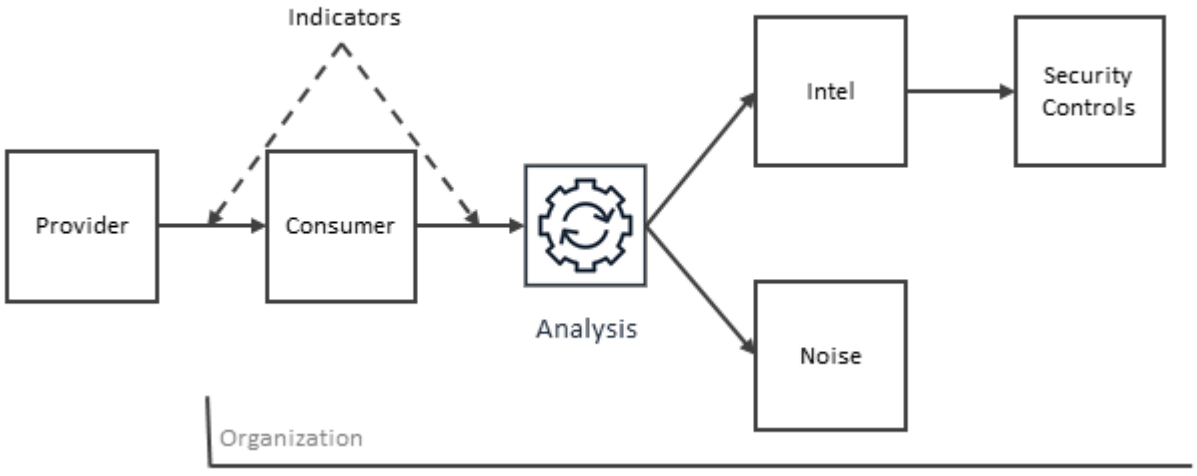

Figure 2: Threat indicators are made available to consumers by providers. The consumer then determines which indicators are intel.

1.2. Cognitive Bias

It is an ongoing challenge to keep analysis as objective as possible, as cognitive bias is not trivial to overcome. Even technology has been suspected to use biased algorithms (Cohen, 2018). It is human nature to defer to the path of least resistance, and so falling back on preconceptions can be "useful in helping us deal with complexity and ambiguity under many circumstances. However, they lead to predictably faulty judgments known as cognitive biases" (Heuer, 1999, p. 111). Bias can lead to distraction (a counterintelligence technique to prevent effective intelligence gathering.) Therefore, the amount of risk of cognitive bias affecting the analysis process is proportional to the volume of indicators evaluated.

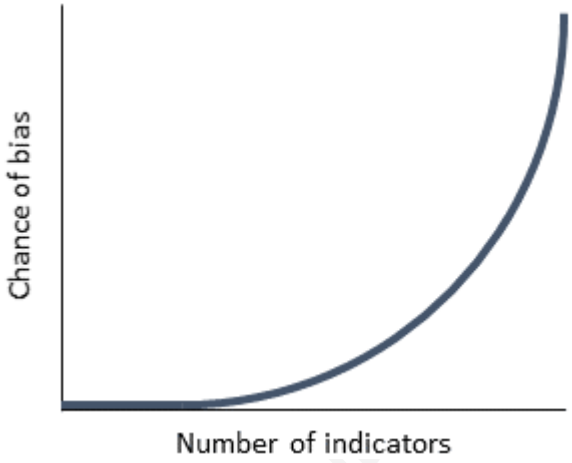

Figure 3: The volume of indicators correlates with the number of shortcuts (or bias) unintentionally used in the analysis process.

"We take shortcuts to make the cognitive load easier," says social advocate Yassmin Abdel-Magied. "But we can learn to manage those natural impulses" (TED Radio Hour, 2019). Confirmation bias--where evidence is sought and used selectively to support a pre-existing belief--may be the most prevalent form that an analyst needs to be aware of and keep in check.

Per the seminal book “The Psychology of Intelligence Analysis” by Richards J. Heuer, "cognitive biases are mental errors caused by our simplified information processing strategies" (Heuer, 1999, p. 2). Heuer discusses four types of cognitive bias that an analyst may exhibit: evaluation of evidence, percent of cause and effect, estimating probabilities, and hindsight. These have been selected for this paper as they may apply more specifically to information (cyber) security than others.

1.2.1. Evaluation of Evidence

"Information presented in vivid and concrete detail often has [an] unwarranted impact, and people tend to disregard abstract or statistical information that may have greater evidential value" (Heuer, 1999, p. 115). There is a tendency to prefer indicators that originate from

interesting sources or events, which may cause less interesting (but possibly more relevant) indicators to become overlooked. Indicators received from a source that has provided useful intel in the past may also get preferential treatment. The following are examples of non-objective thinking as it relates to the evaluation of evidence:

- "The indicators are probably good because they come from a trustworthy source, even though I don't see how they're relevant."
- "I don't think any of these indicators are intel, but I don't want to throw them all out."

While indicators may come from a trustworthy source, it is possible that they're still of low value. It is also possible that a provider's feed has been tampered with to inject incorrect or distracting information. An analyst should resist the desire to find intel among indicators. The evaluation of each indicator must be on its merits.

1.2.2. Perception of Cause and Effect

"Because of a need to impose order on our environment, we seek and often believe we find causes for what are … accidental or random phenomena" (Heuer, 1999, p. 127).

It is human nature to form a hypothesis that explains causality quickly, but sometimes there is no obvious connection between an indicator and the intent of its usage. The analyst may make faulty assumptions and form incorrect conclusions to satisfy their need to understand. In these cases, the analyst makes cognitive shortcuts and may ignore facts to form a satisfactory conclusion. Some examples of this kind of faulty thinking include:

- "The indicators were probably used in this way because that's how I've seen them used in the past."
- "The alternative theory is more unlikely and therefore should not be considered." The principle of Occam's razor (Wikipedia, 2019) should not influence the analysis

process. Making assumptions and taking shortcuts runs the risk of overlooking indicators that have actual value, and overvaluing indicators that may prove to be worthless. An open mind

must be maintained to ensure that each indicator is evaluated using an objective, fact-based approach.

1.2.3. Estimating Probabilities

"Using the ‘availability' rule, people judge the probability of an event by the ease with which they can imagine relevant instances of similar events or the number of such events that they can easily remember" (Heuer, 1999, p. 147).

The process used to determine an indicator’s value may be flawed by “anchoring,” an attempt to find a personal connection to information to make it more understandable. However, deferring too much on personal experience may prevent an objective assessment of an indicator. Overlooked intel and operationalized indicators that generate false positives are the results. Some examples of this kind of faulty thinking include:

- "I worked a security incident before that had indicators like this. They probably have value in this context too."
- "I've never seen an attack scenario before that used indicators like this, so they're probably worthless."

Relying too much on personal experience runs the risk of an analysis that is flawed by cognitive bias. It is crucial that indicator evaluation "consciously avoid[s] any prior judgment as a starting point" (Heuer, 1999, p. 152) to ensure objective conclusions.

1.2.4. Hindsight

"Analysts overestimate the quality of their analytical performance, and others underestimate the value and quality of their efforts. These biases are not simply the product of self-interest and lack of objectivity. They stem from the nature of human mental processes and are difficult and perhaps impossible to overcome" (Heuer, 1999, p. 161).

Cognitive bias exists in situations where experience overly influences the analyst. Over-confidence may cloud the analyst's judgment. Humility is important for maintaining objectivity

and ensuring assessments are based more on facts and less on hunches. Each analysis session should begin with a mental reset which can reduce (if not altogether remove) the influence of past successes and failures. Some examples of faulty thinking include:

- "I operationalized this indicator at my previous job, and it generated a lot of false positives."
- "I helped stop a security attack last week; I know what I'm talking about."

The analyst must always accept that they can make mistakes or may be right even though there are inclinations to believe otherwise. "After action reviews" can also be conducted to review operationalized intel, as they can yield useful information that can improve future analysis.

2. Collection

In terms of origination, there are two types of indicators: Discovered by Us (DBU) and Received by Us (RBU). Discovered by Us (DBU) indicators are internally-generated, culled from incidents, events, and security controls within an organization. Internally-derived indicators may have a higher likelihood of yielding contextually-relevant intel. Received by Us (RBU) indicators, on the other hand, are information provided by an outside organization. Indicators received from this type of source may be of lesser quality (and therefore require more scrutiny) as there is a reduced likelihood of them being contextually relevant. Regardless of origin, the analysis process used for every indicator deserves the same level of rigor and the use of a consistent evaluation process.

2.1. Quality and Context

Sharing organizations have analysis processes for reviewing indicators and generating intel. Some are more mature than others. The quality of various threat feeds, therefore, can vary greatly. For example, one organization's indicator collection process may sweep up all IP addresses involved in an incident, including localhost and common DNS queries. These indicators, while technically relevant to a security event, in most cases should not be shared as they have low value to consumers. Also, some sharing organizations may just "pass-through" intel that they receive without vetting, resulting in a "garbage in garbage out" scenario. A lack of analysis introduces a supply chain risk, where organizations exchange worthless, distracting or even malicious intel.

Intel also has a shelf life. Useful indicators should be imported into security controls as quickly (and safely) as possible, as often vulnerabilities are exploited soon after information about them is disclosed and made public. There is a critical period before potential targets respond by taking defense measures such as applying patches or making configuration changes. Attackers move on once defenders have caught up, and the related intel quickly depreciates. At some point, these indicators are no longer useful and need to be removed from service as they only waste resources by keeping them in operation. Indicators removed from service should be tagged as deprecated in an intel repository, but not deleted in case they need to be reactivated or used for future analysis.

2.2. Attributes

Store indicators with specific attributes, which can then be used by the indicator analysis process. An indicator by itself has no value. The minimal attributes an indicator should have are:

- Value - The value of an indicator, such as an IP address, hostname, or checksum.

- Type - The type of object the indicator is (e.g., an IP address, a hostname, or a checksum.)
- Source - The entity or event that provided the indicator.
- Date first observed - The date when the indicator became intel, used to calculate indicator age.
- Date received - The date when the consuming organization received the indicator.

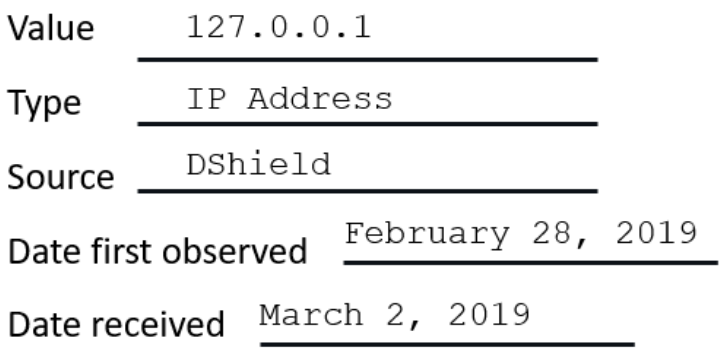

Figure 4: Example form that can be used to document the attributes of each indicator.

An indicator can have other attributes, such as the operationalization date, usage, and its success rate (e.g., number of false positives versus accurate detections). These can be used to monitor the effectiveness of deployed indicators, as well as how successful the analysis process was that yielded the intel.

2.3. Sources

Not all sources are created equal. Due to the resources required for a consumer to process indicators, the receiving organization should be very selective when deciding which threat feeds to ingest. In addition to poor quality, false information in intel feeds misdirects or negatively impacts the trust of sharing organizations. The motto "you get what you pay for" also does not always apply since it is possible that some paid-for threat information providers (or vendors) incorporate low-quality feeds into their product. Therefore, a thorough evaluation should be performed on potential sources before beginning to ingest data from them. Indicator sources

should also be re-evaluated periodically to ensure they are continuing to provide useful information.

2.4. Storage

Intel that results from processing threat indicators requires protection against unauthorized changes or deletions. Integrity controls are needed to enforce the principle of least privilege. False positives or false negatives that could result from operationalizing compromised intel (due to a lack of sufficient integrity and access controls) could pose serious security, legal or compliance consequences for an organization. To ensure sensitive data protection, apply the label of "critical" to intel stored in a repository.

Indicators that were not deemed to be intel, or were previously utilized but later deprecated, should also be stored. Review these indicators if they are received again in the future, or if a related event occurs but is not detected or prevented due to non-operationalized intel. Reviews can drive improvements in the analysis process.

3. Analysis

Indicator analysis needs to be as objective as possible, which is easier said than done, as all analysts have some form of cognitive bias to overcome based on their perspectives, experiences, beliefs, and interests. Use of a repeatable process that has minimal subjectivity is the best way to ensure finding only useful and relevant indicators. The following are the components of a proposed threat indicator analysis process.

3.1. Cognitive Bias Checklist

Every indicator analysis session begins with a clear mind to keep the cognitive biases listed in section one "front of mind." This exercise uses a quick self-evaluation checklist. An

analyst runs through this checklist at the start of each session, which contributes to a mental reset (similar to checking your "mood elevator" (Senn Delaney, 2016) that reduces the chance of cognitive bias negatively influencing the evaluation process.

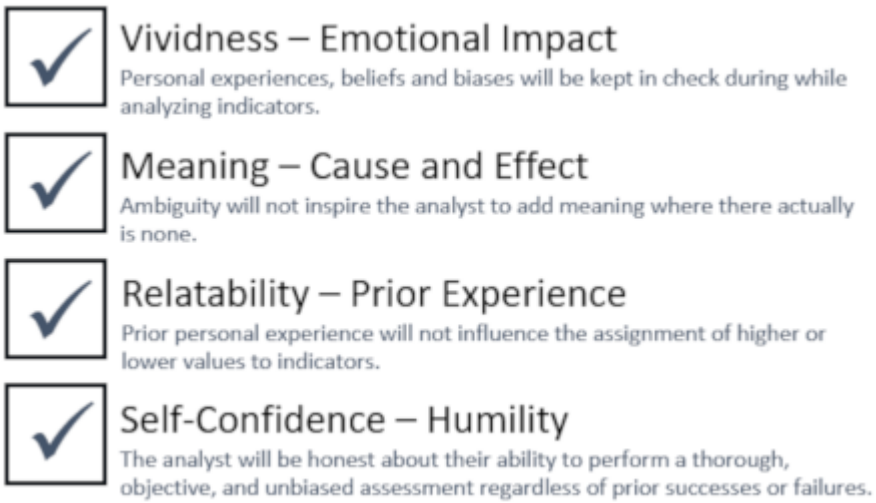

Figure 5: A checklist based on Heuer's four cognitive biases, used at the start of each indicator analysis session.

3.2. Evaluation Criteria

The attributes of each indicator are scored. These attributes include, at a minimum:

- Source - The trustworthiness of the producer, based mostly on whether it has provided useful intel in the past.
- Type - The type of indicator could influence its relevance. For example, an analyst should assign a lower score to ephemeral IP addresses.
- Age - The longer intel is exposed and known, the more likely defenders have caught up, and attackers have moved on.
- History - Indicators consuming organization had prior experience with may be considered more contextually relevant and therefore of a higher (or lower) value.

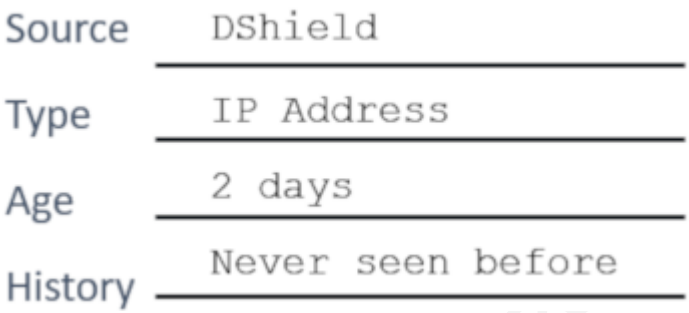

Figure 6: An example worksheet used for each indicator.

3.3. Scoring

Assign a score to each indicator based on its attributes. Score each attribute on a 1 to 5 scale, where one is the least important or relevant and five is the most. The following is an example calculation based on figure 5:

Source	(S)	=	4	Good intel has been received from the provider before.
Type	(T)	=	3	The indicator is an IP address, which is somewhat ephemeral.
Age	(A)	=	4	This is a recently-discovered indicator.
History	(H)	=	2	The consuming organization has never seen this indicator before.
Total		=	14	
Threshold		=	12	
Conclusion			Useful, should be considered intel	

Figure 7: Example of indicator scoring.

Calculate the indicator's overall score by evaluating its attributes. Indicators with a sufficiently high score (represented numerically or as a letter grade) become intel. Preserve the calculus used for periodic reevaluation. Each organization can customize the scoring algorithm, such as assigning weights to the attributes, adding attributes to the equation, and determining the threshold that must be reached for an indicator to be considered intel. Ensuring that all analysts in an organization use the same algorithm promotes consistency as well as a more reliable and valid assessment process.

4. Production

New intel is identified and operationalized by deploying to security controls. Controls can detect and prevent security events and incidents. The indicator type informs the deployment strategy. A checksum is not usually utilized by a network proxy, just as file integrity monitoring does not find DNS hostnames useful.

4.1. Deployment

The following are some of the IT security controls that organizations use, followed by their function. “Detect” means it generates an alert when a matching event is observed, whereas “prevent” means the control fires an alert but also takes steps to prevent the attack or intrusion attempt from being successful, such as blocking a connection attempt to a malicious IP address. Observing connection attempts to malicious IP addresses or domains can also be traced back to malware installed on a host.

- Security Information and Event Management (SIEM) solution - Detect.
- Firewall - Detect and Prevent.
- Web application firewall (WAF) - Detect and Prevent.
- Proxy - Detect and Prevent.
- File integrity monitor (FIM) - Detect.
- Intrusion detection system (IDS) - Detect.
- Intrusion prevention system (IPS) - Detect and Prevent.

4.2. Monitoring

Once intel is deployed, monitor their usage to ensure that the number of false positives that result is limited. There will always be false positives, but their volume should be contained such that they don't significantly distract or debilitate security functions. If there are too many false positives, then it is possible the indicator has a lower value or relevance than initially thought. It is also possible that the indicator is deliberately distracting. In either case, un-deploy

the indicator and review it and any related security controls. Test events can also help identify false negatives, where an incident occurred, but security controls and personnel failed to detect or prevent it. For example, if a desktop makes a DNS lookup request for a malicious hostname and the resolvers have blocking enabled (with that indicator in the blocklist), then a matching event should be observed. If not, then the indicator has not been implemented correctly, or the control using it is having issues.

Establish a rollback plan before making any changes to production systems. Deploying new indicators can cause adverse impact, such as blocking legitimate traffic or generating too many false positives. Be prepared to quickly remove indicators from service as needed.

5. Dissemination

Contributing to the security community supports the philosophy that "a rising tide lifts all boats." Any organization that receives threat information via sharing should reciprocate by making that same intel available to peer organizations. The process used to share intel includes the use of controls and procedures that support the security tenets integrity and availability, such that consumers can trust the information they receive, and that it is available when requested. It is essential to maintain a high level of quality for threat feeds as they can quickly become an integral part of a security operations center (SOC). However, any organization that consumes a feed is placing trust in the producer, which exposes sharing organizations to cyber supply chain risks. Figure 7 shows an example of the cyber supply chain involved in the sharing of a single indicator.

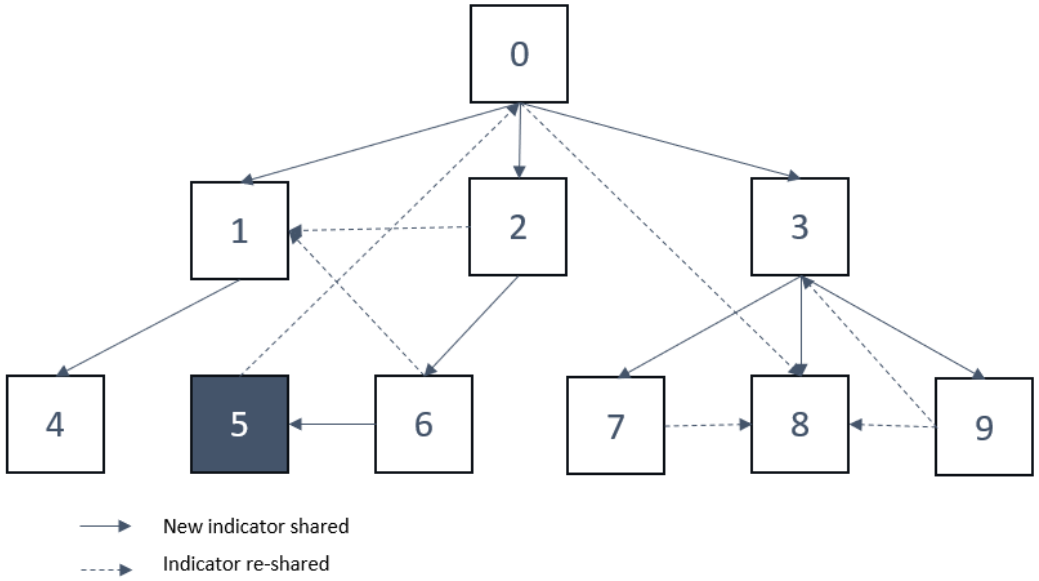

Figure 8: An example of how indicators are shared (and reshared) among peer organizations.

If providers are trusted implicitly, then so too are the controls they use. Tampering and altering of data in transit are always possible. Therefore, producers of shared intel need to earn the trust of consumers by demonstrating the use of security controls to enforce integrity and availability.

5.1. Sharing Process

Providers should use a formalized, repeatable, and ideally automated method for information sharing. Intel tagged shareable is made available via either a pull or push model. The use of standards such as STIX and TAXII (MITRE, 2018) promote shareability. The process used to make intel available to other organizations should include not just the intel but information about it, such as how it originated, when it was received, and if there were any matches (hits.) Additional information helps the consumer determine whether the data obtained

is intel-worthy. Ideally, the sharing organization can also get feedback from consumers about any events they observed as a result of using the intel they received.

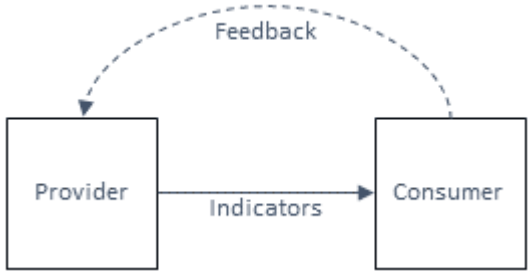

Figure 9: A feedback loop helps intel providers improve their product.

6. Proof of Concept

A proof of concept (POC) was developed to demonstrate this approach using open source tools and Amazon Web Services (AWS). The open source intel repository solution Threat_Note (Defense Point Security, 2016) and AWS GuardDuty (Amazon Web Services, 2019) comprise the POC. GuardDuty is a service that can be configured to alert on and prevent a wide array of suspicious and malicious security events, including blocking network traffic to specific IP

addresses.

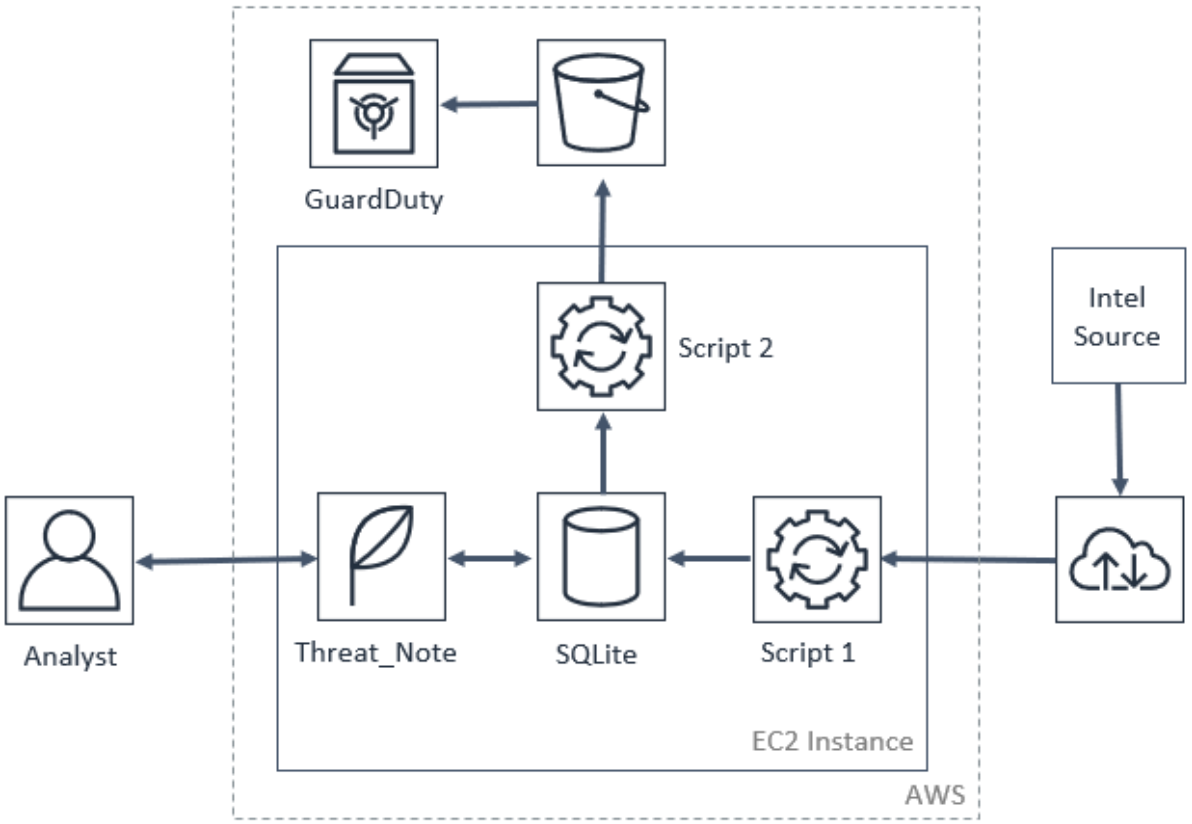

Figure 10: High-level architectural diagram of the analysis process.

6.1. Design

The code base for Threat_Note was modified and forked. For the POC, Threat_Note is the central spoke of the hub where new IPv4 addresses are received by an open source threat indicator provider (Dshield (Internet Storm Center, 2019)) via Script 1. These IP addresses are added to the Threat_Note SQLite database as new indicators with the tag NEEDS_ANALYSIS. An analyst first uses the cognitive bias checklist, then reviews each indicator by evaluating its attributes. The result of the analysis is an overall score. Indicators whose score meets or exceeds the intel threshold receive the tag READY_TO_DEPLOY; all others get DO_NOT_DEPLOY. Include notes that support the decision.

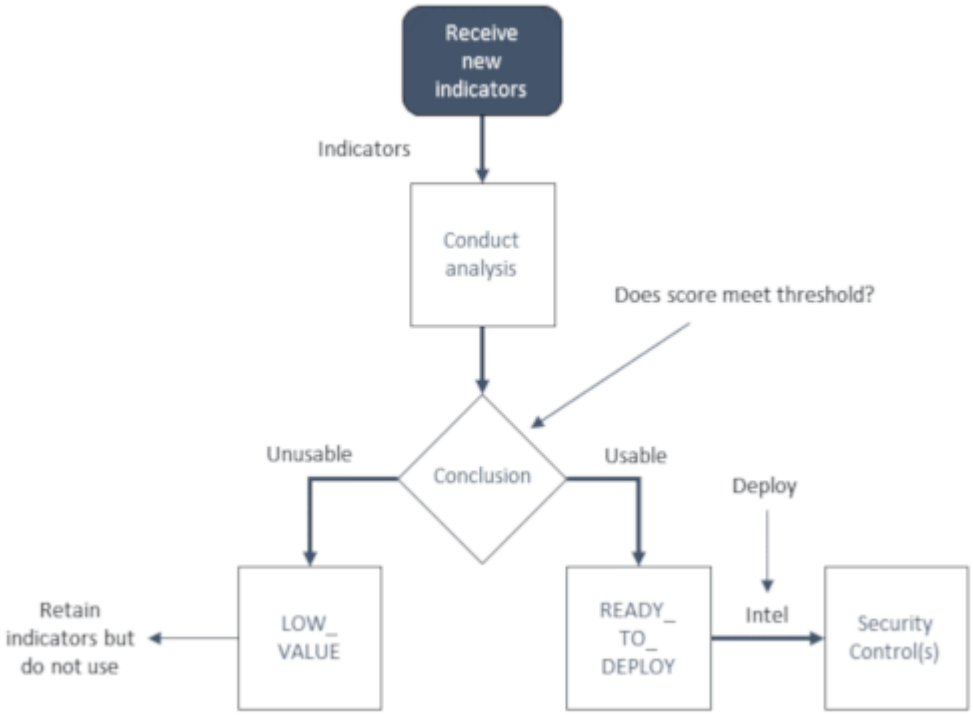

Figure 11: Overview of how indicator analysis is used to push only intel (actionable indicators) to a security control.

Per figure 10, Script 2 automatically pushes indicators tagged with READY_TO_DEPLOY to a security control. In this case, the tagged indicators are written to a file hosted by an S3 bucket, from which GuardDuty retrieves an IP blocklist. Just as this approach applies to various security controls and endpoints, other Malware Information Sharing Programs (MISP) such as Hive (TheHive Project, 2019) can be used instead of Threat_Note to facilitate indicator analysis. Per the proof of concept, the following are the steps that an analyst would take to review newly-received indicators:

1. Run through the cognitive bias checklist (see 3.2.)
2. Use a web browser to visit the Threat_Note URL and log in using analyst credentials.
3. View indicators in the dashboard that have the tag NEEDS_ANALYSIS.

4. Review each of these indicators using the analysis process.
    a. Run through evaluation criteria (see 3.2.)
    b. Calculate the score.
    c. Based on the score, assign an appropriate label to the indicator.
        i. Use READY_TO_DEPLOY to promote to intel.
        ii. Use DO_NOT_DEPLOY for unusable indicators.

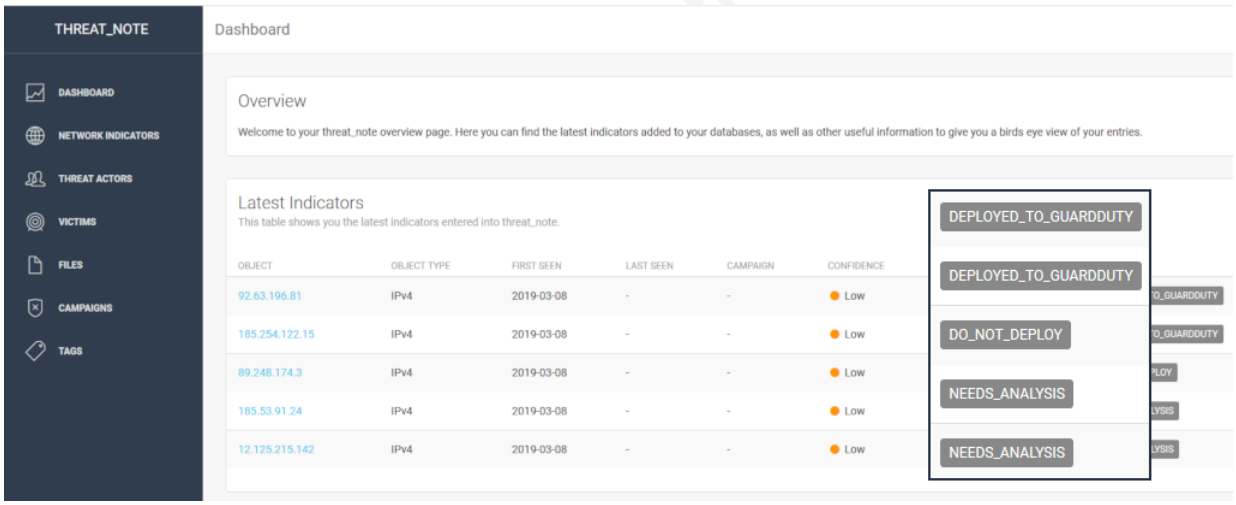

Figure 12: Example screenshot of using Threat_Note for a proof of concept.

A separate process (Script 2) runs independently at intervals which does the following:

1. Indicators in the Threat_Note repository that are tagged READY_TO_DEPLOY are selected.
    a. Tagged indicators are pushed to security controls (in this case writes to a CSV file in an S3 bucket that GuardDuty reads from.)
    b. The tag of these indicators changes to DEPLOYED (or can be altered to DEPLOYED_TO_[CONTROL]) in the Threat_Note database.

Security controls detect and prevent security events related to deployed indicators (e.g., network traffic to IP addresses in the blocklist.) Retain indicators tagged DO_NOT_DEPLOY for reevaluation.

7. Conclusion

Starting each indicator analysis session with a run through of a cognitive bias checklist reduces the subjectivity concerns raised by Richards J. Heuer Jr. and other intel thought leaders. Use of a repeatable and consistent process that scores each indicator based on its attributes increases the quality of the review process. Internal security controls limit access to and protect the integrity of intel that has been operationalized, as well as indicators that are deemed low value (including analyst notes.) Lastly, all organizations with a security function are encouraged to participate in the threat intel community by sharing with peer organizations, including providing feedback about intel utilization. Intel sharing is more significant than any single organization; it can also improve security for a community.

References

University of Groningen (2005). Dutch intelligence - towards a qualitative framework for analysis. Retrieved January 17, 2019, from https://www.rug.nl/research/portal/files/33123471/c3.pdf

Heuer, Richards J., Jr. (1999). Retrieved January 19, 2019, from https://www.cia.gov/library/center-for-the-study-of-intelligence/csi-publications/books-and-monographs/psychology-of-intelligence-analysis/PsychofIntelNew.pdf

Defense Point Security LLC. (2016). Retrieved January 2, 2019, from https://github.com/defpoint/threat_note

Stevens, Didier (2019). Retrieved February 1, 2019, from https://www.dshield.org/

TheHive Project (2019). The Hive Incident Response Platform. Retrieved February 13, 2019, from https://thehive-project.org/

Pokorny, Zane (October 16, 2018). What Are the Phases of the Threat Intelligence Lifecycle? Retrieved December 23, 2018, from https://www.recordedfuture.com/threat-intelligence-lifecycle/

SANS Institute (2018). Retrieved October 20, 2018, from https://www.sans.org/course/cyber-threat-intelligence

Bimfort, Martin T. (September 18, 1995). A Definition of Intelligence. Retrieved January 19, 2018, from https://www.cia.gov/library/center-for-the-study-of-intelligence/kent-csi/vol2no4/html/v02i4a08p_0001.htm

The MITRE Corporation (2018). Structured Threat Information eXpression (STIX™) 1.x Archive Website. A structured language for cyber threat intelligence. Retrieved January 20, 2019, from https://stixproject.github.io/

TED Radio Hour (February 15, 2019). Yassmin Abdel-Magied: Is It Possible To Unravel Unconscious Bias? Retrieved February 20, 2019, from https://www.npr.org/2019/02/15/694279494/yassmin-abdel-magied-is-it-possible-to-unravel-unconscious-bias

Cohen, Noam (December 14, 2018). Google's Algorithm Isn't Biased, It's Just Not Human. Retrieved February 2, 2019, from https://www.wired.com/story/google-algorithm-conservatives-biased-its-just-not-human/

Senn Delaney (2016). The mood elevator: Harnessing your emotional intelligence. Retrieved March 2, 2019, from http://sdtv.senndelaney.com/senn-delaney-chairman-larry-senn-discusses-the-mood-elevator-how-to-live-life-at-your-best

Wikipedia (2019). Occam's razor. Retrieved February 23, 2019, from https://en.wikipedia.org/wiki/Occam%27s_razor

Amazon Web Services (2019). Amazon GuardDuty: Protect your AWS accounts and workloads with intelligent threat detection and continuous monitoring. Retrieved February 28, 2019, from https://aws.amazon.com/guardduty/