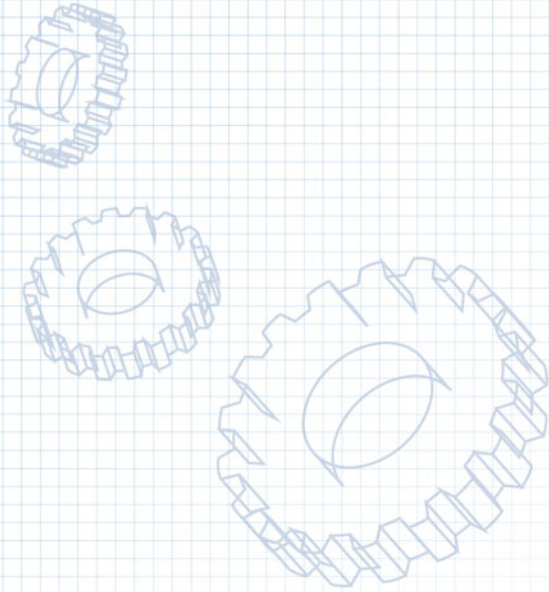# Sophisticated Indicators for the Modern Threat Landscape:
An Introduction to OpenIOC

Table of Contents

Introduction

In the current threat environment, rapid communication of pertinent threat information is the key to quickly detecting, responding and containing targeted attacks. Traditionally, threat information is harvested from hosts and networks, and then encoded in technology-specific configurations (e.g. Snort rules) or compiled into written reports that are passed on to humans for sharing. Even inside the same organization, the ability to share threat information may depend on overburdened staff reading paper reports and passing them on to others in the organization, with each transition increasing the time from when an attacker first strikes to when the organization reacts. By the time that many organizations begin to react, the information is outdated and the attackers have had plenty of time to infiltrate broadly across the network.

The key to increasing your ability to detect, respond and contain targeted attacks is a workflow and set of tools that allows threat information to be communicated across your enterprise at machine speed. OpenIOC is a format for recording, defining, and sharing information that allows your organization to accomplish this by sharing many different types of threat information both internally and externally in a machine-digestible format. OpenIOC is an open and flexible standard that can be modified on the fly as additional intelligence is gathered so that you can capture input from human subject matter experts and translate it into a format that can be used by various technologies to sweep your enterprise for signs that it has been compromised or is currently under attack to combat advanced targeted attacks in a manner that makes real remediation a realizable goal and enhances your security posture to combat future intrusions.

IOCs & OpenIOC

Indicators of Compromise (IOCs) are forensic artifacts of an intrusion that can be identified on a host or network. OpenIOC is a threat information sharing standard that allows you to logically group forensic artifacts, and communicate this information in a machine readable format. The terms are sometimes used interchangeably, but an IOC (also sometimes just called an *Indicator*) is a logically grouped set of descriptive terms (each called an "Indicator Term") about a specific threat while OpenIOC is the language used to describe those specific sets (e.g. an incident response team would use the OpenIOC format to write multiple IOCs during the course of responding to an incident).

OpenIOC is written in XML (Extensible Markup Language). XML provides a well-recognized standard format of encoding data into a machine readable format that is used in many different standardized methods of communicating data. The use of XML provides several benefits for consumers of OpenIOC. First, while the base schema used for OpenIOC is fairly small and lightweight, it can also be extended with indicator sets (also written in XML) that are supplied with the base schema. Additionally, custom indicators that suit a particular environment or threat that are not already described can be created and added if an organization needs them. Since OpenIOC is written in XML, it is also easy to create utilities to convert or parse OpenIOC to other formats that might contain information that could feed into or benefit from the threat information contained in an IOC.

Indicator Terms are the name of the specific types of data elements that are included in IOCs. Indicator Terms are usually organized in *Indicator Term Documents*, which are groupings of indicators inside an XML document. When creating an IOC, an investigator can use as many or as few terms from as many or as few sources as they like. An organization desiring to extend OpenIOC to include new types of elements that are unique to their enterprise or circumstances would create and host an Indicator Term Document that contained the new Indicator Terms they wished to make available for others to use.

IOC Functionality

The indicator terms that MANDIANT currently provides for OpenIOC detail over 500 different types of evidence that can be gathered in an enterprise. These definitions are derived from years of practical experience in industry leading incident response conducted by MANDIANT. This combined with the flexible nature of OpenIOC, with nested logical structures, has led to much greater functionality than standard static signature based technologies.

Indicators start in complexity with simply looking for signature of specific artifacts. These can be the traditional forensic artifacts such as MD5 checksums, compile times, file size, name, path locations, registry keys, and so on. But they can also include items from more advanced forensic techniques, such as memory forensics, looking for artifacts that are much harder for attackers to change, or artifacts that attackers are more likely to recycle, such as running process components (including process handle names), the imports and exports used by an executable, and more. These can all be combined together in different logically grouped combinations, which are refined as the investigator learns more about the intrusion they are working on. Many different types of specific indicators can be combined together in one IOC, so that any of several sets of signatures of differing types of complexity could apply within one particular IOC.

Going beyond making logically complex indicators, there are also additional ways to use IOCs than just a straight query against a host. IOCs can also be used with logical operators to exclude entire classes of the hosts or network being examined when querying against harvested data sets. Instead of looking for a specific file using terms that have to precisely match, IOCs can also be used to match all of the files that *should* be on a particular part of a system. An investigator would collect unfiltered data from systems, and then run an IOC against the collection to look for the files that stand out.

Simple use cases allow querying for forensic artifacts such as:

- ❖ Looking for a specific file by MD5 sum (hash), file name, size, create date, or other file attributes
- ❖ Looking for a specific entity in Memory (process information, running service information)
- ❖ Looking for a specific entry or set of entries in the Windows Registry
- ❖ Combining these together in various combinations to provide better matching and less false positives than searches for individual artifacts.

More complex use cases and techniques combine these together and allow even more depth:

- ❖ Instead of just looking for specific file artifacts in one part of the operating system or network, groups of artifacts can be combined together using the logic of OpenIOC to create a match on

    artifact groups that are common across families of malware or other intrusion tools (such as from the same authors or threat actor groups).

❖   Instead of hunting down a specific known bad file, an incident responder could make a whitelist of the files that were known to be in a directory, and then catch all the files that were NOT on that list, assuming the investigator had a full collection of what was on the system.

Looking for Methodology

Potentially, the most powerful way of creating an Indicator is to describe the attacker's methodology. Indicators attempting to detect methodology do not focus on a specific piece or pieces of forensic evidence directly tied to malware or compromise. Instead, they focus on the commonality of the methods that an attacker (or set of attackers) may use. Methodology indicators don't necessarily show a specific instance of a compromise, but they will show the result of recurring tactics repeated by a certain group of adversaries. As such, they are the hardest to write, but when done properly, they capture evidence of a behavior that is only done by attackers or intruders as opposed to legitimate users or system use.

Looking for methodology allows you to:

❖   Look for specific locations in the file system, registry, or other parts of the operating system that hostile actors regularly use in the course of their intrusion, even if it has nothing to do with the initial exploit or compromise.

❖   Look for sets of artifacts left by tools or toolkits used by adversaries that would be expensive for them to change or modify.

❖   Look for signs of adversary activity on systems that were used for lateral movement, that were not directly compromised but show signs of activity that does not fit normal system user usage.

In real world cases, IOCs can combine any and all of the above types of functionality, or you can just use a single type of functionality by itself. The investigator tailors IOCs to the needs of their investigation, and the flexibility of OpenIOC allows them to change that as the case evolves, without having to write a new Indicator.

Writing Effective Indicators

Unlike some other data standards used to describe threat information, there is no one-to-one mapping of an instantiation of a threat (such as a piece of malware) and the entry in the data standard used to describe it. This flexibility of OpenIOC is a great strength, but also makes it possible for some IOCs to not be as useful as others. An IOC of "OR filename = *.exe" is definitely going to identify a lot of files – but this is a rather poor indicator, generating many false positives as it hits every executable file on a system. Better IOCs achieve the best true positive rate while having the lowest false positive rate (hitting on things which are normally found on a system or not related to an intrusion). Ultimately, the best IOCs have these properties:

1. The IOC identifies only attacker activity.
2. The IOC is *inexpensive* to evaluate – it is typically simple and evaluates information that is less expensive to collect or calculate.
3. The IOC is *expensive* for the attacker to evade. In other words, to evade the IOC the attacker has to drastically change tactics, tools, or approach.

Using IOCs in the Investigative Lifecycle

IOCs are a part of an effective and proven workflow that is at the core of MANDIANT's own incident response process. The flexibility and machine-readable nature of the OpenIOC format are what makes this possible. The following outline shows some of the steps involved in the lifecycle of an investigation, and how OpenIOC and IOCs make that possible.

❖ **Initial Evidence** – Evidence of a compromise is detected, either on a host or on the network. This may be in response to law enforcement (LE) notification or an anomaly noticed from a variety of sources. Regardless of what led to it, responders investigate and identify something which is a concrete forensic indicator of an intrusion.

❖ **Create IOCs for Host & Network** – Following the initial discovery of forensic evidence, the investigators create an IOC from the existing data. The specific type of IOC created will vary based on the evidence, the environment, and the skill and comfort level of the investigator. The flexibility of OpenIOC allows a limitless number of permutations on how an Indicator can be crafted, so the investigator using OpenIOC has a lot of options as to how they want to proceed.

❖ **Deploy IOCs in the Enterprise** – Once an IOC or set of IOCs have been created, the investigator will deploy these to technology that can look for the existence of the IOC(s) on other systems or other parts of the network. In the MANDIANT workflow, these are fed into MANDIANT Intelligent Response™ (MIR) appliances, which then communicate with MIR Agents on hosts, or

monitor network traffic. IOCs could be easily transformed and fed into IDS, IPS, HIDS/HIPS, SIEMS, or other investigative tools to look into the enterprise.

❖ **Identify Additional Suspect Systems** – After deploying the IOCs into suitable technologies, additional systems will be identified, unless the first host was the only endpoint compromised.

❖ **Collect Evidence** – Additional evidence is acquired from the additional systems that have been identified.

❖ **Analyze Evidence** – The additional data collected is analyzed. This can identify further intrusion, false positives, or additional intelligence for the investigators. This feedback then allows for the investigator to refine their searches and to return to the start of the workflow.

❖ **Refine & Create New IOCs** – The investigative team can create new IOCs based of their findings in the enterprise and additional intelligence, and continue to refine their cycle until they feel they either have exhausted the need to find new information, or other factors force them to stop investigating and move to remediation.
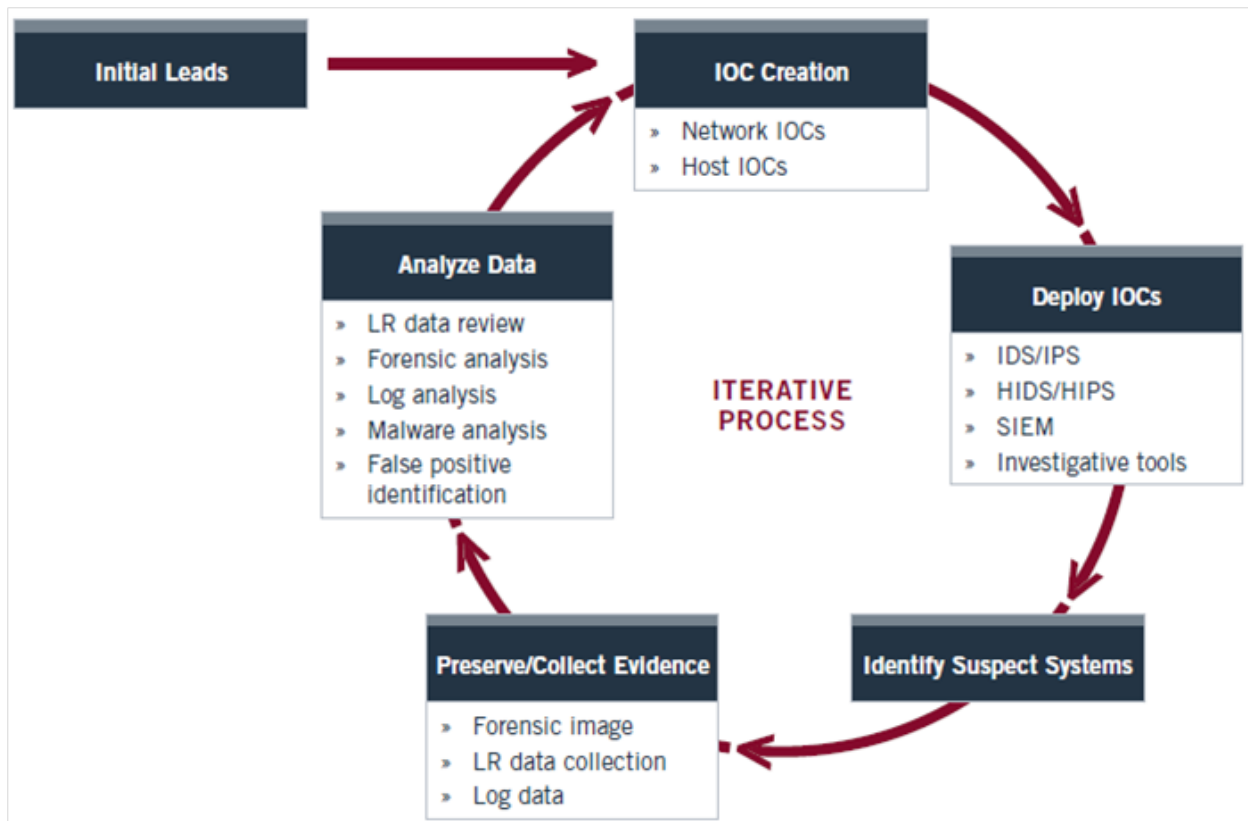
Available Tools to Create, Edit & Use OpenIOC

MANDIANT released OpenIOC under an open source license so that the greater incident response community can benefit from MANDIANTs process and lessons learned and have a standardized format to communicate threat information. To further this goal, MANDIANT has released tools that allow interested parties to utilize IOCs written in OpenIOC without having to have a business relationship with MANDIANT. These tools currently are:

MANDIANT IOC Editor: This tool allows for the easy creation of IOCs using a graphical interface rather than having to edit raw XML. IOCs created with IOC Editor can then be shared with other responders inside and outside that organization.

MANDIANT IOC Finder: Once an IOC has been created, the creator can use IOC Finder tool harvest data from a host. Once data is harvested, IOC Finder can be used to check the IOC against the collection of data and see if the host matches the conditions laid out in the IOC. Based on the results they can refine the IOC, or use it to search other endpoints once the IOC has been verified.

Using these tools incident responders can:

❖ Take evidence and create an IOC using IOC Editor
❖ Use IOC Finder to verify the newly created IOC
❖ Share intelligence with other entities who understand OpenIOC
❖ Use IOC Finder to search other individual endpoints in their enterprise

In addition, you can use commercial tools such as MANDIANT Intelligent Response to search for IOCs across tens of thousands of endpoints. It is MANDIANT's hope and expectation that over time additional free utilities and commercial tools will be able to consume IOCs.

Conclusion

The threat landscape that confronts both the government and commercial sector is more challenging than it has ever been. While defenders must succeed 100% of the time, the attackers only need to get through once to be successful. In almost all environments, some type of compromise is inevitable. Rapid dissemination of pertinent information among defenders is one of the few effective ways to combat target attacks by sophisticated threat actors, and a necessary component of a successful incident response & containment workflow. Indicators of Compromise, written in OpenIOC, allow organizations to define pieces of threat intelligence in a standardized, logically organized manner, encode the experience and knowledge of human subject matter experts in a machine readable format, and use the speed of responding in machine time to communicate that intelligence across their enterprise or to other entities they wish to share intelligence with. With OpenIOC, your organization can harness the power of the many years of incident response experience that went into creating and refining OpenIOC, and empower your personnel to respond to incursions with the speed and intelligence they need to change the current imbalance of power that is so greatly favors the attackers.