Malware Attribute Enumeration and Characterization — MAEC™

A Structured Language for Attribute-Based Malware Characterization

MAEC is a structured language for encoding and communicating high-fidelity information about malware based upon attributes such as behaviors, artifacts, and attack patterns.

By eliminating the ambiguity and inaccuracy that currently exists in malware descriptions and by reducing reliance on signatures, MAEC aims to:

- **Improve human-to-human, human-to-tool, tool-to-tool, and tool-to-human communication about malware**
- **Allow for the faster development of countermeasures by enabling the ability to leverage responses to previously observed malware instances**
- **Reduce potential duplication of malware analysis efforts by researchers**

Challenge

Modern methods for detecting and combating malware often rely on the characterization of malware attributes and behaviors. The use of static and dynamic analysis techniques allows for an encompassing profile of malware to be constructed based upon its disassembled binary and observed run-time behavior.

Yet, the lack of an accepted standard for unambiguously characterizing malware means that there is no clear method for communicating the specific malware attributes detected in malware by the analyses, nor for enumerating its fundamental makeup. The results are non-interoperable and disparate malware reporting between organizations, disjointed or inaccurate malware attribution, the duplication of malware analysis efforts, increased difficulty in determining the severity of a malware threat, and a greater period of time between malware infection and detection/response, among others.

Solution

MAEC solves these problems. The characterization of malware using abstract patterns offers a wide range of benefits over the usage of physical signatures, and allows for the accurate encoding of how malware operates and the specific actions that it performs. Such information can not only be used for malware detection, but also for assessing the end-goal the malware is pursuing and the corresponding threat that it represents.

Focusing on the attributes and behaviors of malware facilitates detection and analysis of emerging, sophisticated malware threats that circumvent the traditional signature-based and heuristic approaches. Characterizing malware in a standard way supports collaboration across organizations and the identification of common behavior, functionality, and code bases across instances of malware.

MAEC achieves this end result by utilizing three community-developed components to define the standardized MAEC Language:

- Element dictionaries
- Schemas for defining vocabulary syntax
- Standard output formats based on schemas

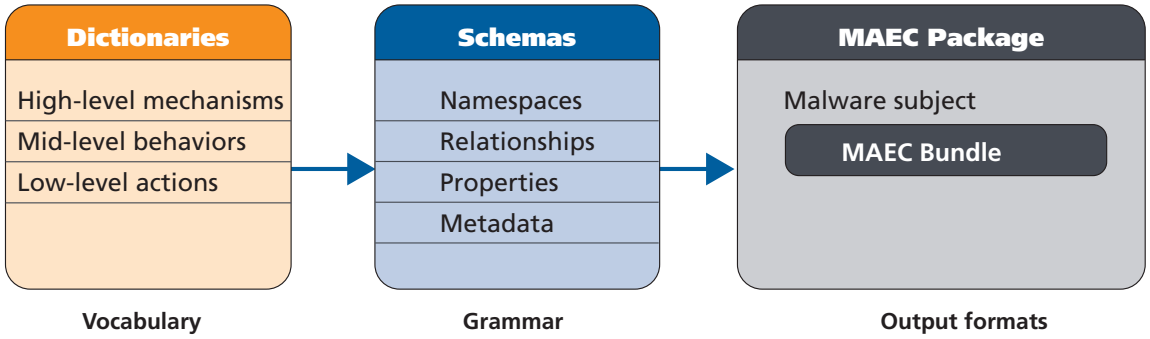
MAEC Language

MAEC is being developed as a formal language for characterizing attributes and behaviors of all types of malware. Initially MAEC will focus on characterizing the most common malware types, including Trojans, worms, and rootkits, but will ultimately be applicable to more esoteric malware types. As a language, MAEC will have a grammar and vocabulary that provide a standard means of communicating information about malware attributes.

MAEC™
maec.mitre.org

The MITRE Corporation maintains MAEC and its public Web site presence and provides impartial technical guidance to the MAEC Community throughout the process to ensure MAEC serves the public interest.

MITRE
202 Burlington Road, Bedford, MA 01730-1420
www.mitre.org



MAEC's core components include a vocabulary, grammar, and forms of standardized output.

MAEC Dictionaries – a series of dictionaries for defining three distinct levels of malware elements—low-level actions, mid-level behaviors, and high-level mechanisms.

MAEC Schemas – a syntax for the vocabulary of actions, behaviors, and taxonomies, and an interchange format for structured information about these elements.

MAEC Output Formats – standard output formats that can be used for particular use cases, including the description of a malware instance, malware intrusion set, or malware families in terms of MAEC's dictionaries and schemas.

MAEC Use Cases

As a domain-specific language for the characterization of malware, MAEC has a broad range of uses, especially with regards to malware analysis and anti-malware operations. The following are just a few of the use cases that MAEC will support:

Analysis-Oriented Use Cases

- Common Vocabulary for Malware Analysis
- Enhanced Data Sharing Between Malware Repositories
- Common Format for Malware Analysis Tool Output
- Objective Criteria for Anti-malware Tool Assessment

Operations-Oriented Use Cases

- Uniform Malware Reporting Format
- Malware Detection
- Malware Threat Assessment
- Malware Response
- Malware/Attacker Correlation

MAEC Compatibility

MAEC Compatibility provides for a product or service to be reviewed and registered as officially "MAEC-Compatible," thereby assisting organizations in understanding and leveraging the three different types of capabilities that can leverage the MAEC Language:

- Content Creation Product or Service
- Content Repository
- Content Consumer

Clearly defining and articulating these three capabilities allows enterprises and end users to easily understand how a given product, service, or repository is using the MAEC Language, and thus how their requirements for discussing, analyzing, detecting, and/or preventing malware could be further enhanced through the use of MAEC-Compatible Products and Services.

If your organization uses or is planning to use MAEC, review the MAEC Compatibility Program section on the MAEC Web site for instructions on how to participate and/or contact maec@mitre.org to learn more.

Feedback Requested

MAEC Community members can make contributions to MAEC development and manage issue tracking for the MAEC schemas, utilities, specifications, and supporting information by joining the MAEC Community at https://maec.mitre.org/community/. Members of the cyber security community are invited to participate in this growing community effort.