

Continuous Monitoring via Threat-related Software Assurance Automation Standards

Sean Barnum

Sep 2012



MITRE

Software Assurance and Continuous Monitoring

- **Current move occurring from discrete point-in-time security assessments to continuous monitoring**

- **Continuous monitoring is currently focused on vulnerability management and secure configuration management utilizing SCAP set of standards (CVE, CVSS, OVAL, XCCDF, CPE, etc.)**

- **Evolutionary goal is to incorporate more software assurance activities through ongoing weakness scanning and security testing**
 - Software Assurance automation requires a balance of weakness (inward) and threat (outward) related information standards
 - Weakness side is covered primarily by CWE and SCAP standards
 - Threat side is covered by CAPEC, MAEC, CybOX and STIX

What are Attack Patterns?

- **Blueprint for creating a specific type of attack**
- **Abstracted common attack approaches from the set of known exploits**
- **Capture the attacker's perspective to aid systems and software developers, acquirers and operators in improving the assurance profile of their systems and software**

Common Attack Pattern Enumeration and Classification (CAPEC)

- **Community effort targeted at:**
 - Standardizing the capture and description of attack patterns
 - Collecting known attack patterns into an integrated enumeration that can be consistently and effectively leveraged by the community
 - Gives you an attacker's perspective you may not have on your own
- **Where is CAPEC today?**
 - <http://capec.mitre.org>
 - Currently 386 patterns, stubs, named attacks



Leveraging Attack Patterns Throughout the Software Lifecycle

- Guide definition of appropriate policies
- Guide creation of appropriate security requirements (positive and negative)
- Provide context for architectural risk analysis
- Guide risk-driven secure code review
- Provide context for appropriate security testing
- Provide a bridge between secure development and secure operations

Attack Patterns Help Answer Foundational Questions Regarding Secure Operations

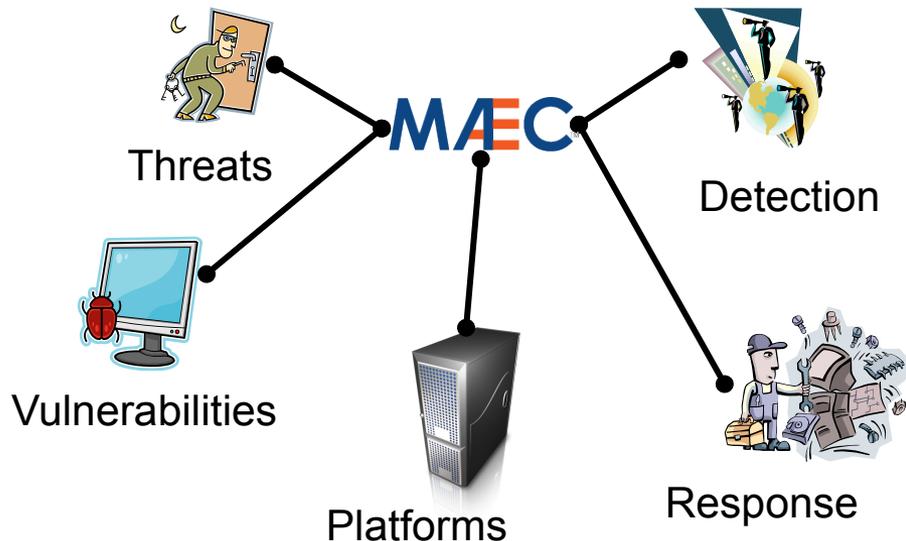
Question	Role of Attack Patterns
Are we being attacked? (Were we attacked?)	Attack patterns offer structured descriptions of common attacker behaviors to help interpret observed operational data and determine its innocent or malicious intent.
How are we being attacked?	Attack patterns offer detailed structured descriptions of common attacker behavior to help interpret observed operational data and determine exactly what sort of attack is occurring.
What is the objective of the attack?	Elements of attack patterns outlining attacker motivation and potential attack effects can be leveraged to help map observed attack behaviors to potential attacker intent.
What is our exposure?	The structure detail and weakness mapping of attack patterns can provide guidance in where to look and what to look for when certain attack pattern behaviors are observed.
Who is attacking us?	Attack pattern threat characterization and detailed attack execution flow can provide a framework for organizing real-world attack data to assist in attribution.
What should we do to prevent against attacks in the future?	Attack patterns offer prescriptive guidance on solutions and mitigation approaches that can be effective in improving the resistance tolerance and/or resilience to instances of a given pattern of attack.

CAPEC Status

- **V1.7.1 released**
 - Aligned consequences structure with CWE/CWSS
- **Actively being used within various organization processes and tools**
- **Now part of FISMA2 and DoD PPPs**
- **Continuing to be integrated into Common Criteria / ISO**
 - ISO TR 20004, ISO TR 30127

- **Additional effort and resources being applied over the next year**
 - Refine content
 - Refine schema
 - New patterns
 - Compatibility program

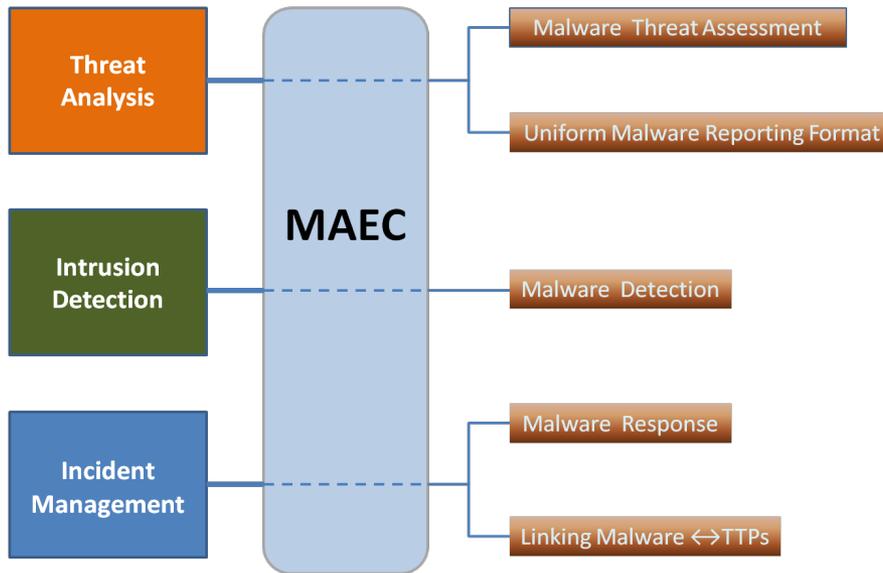
Malware Attribute Enumeration and Characterization (MAEC)



- **Language for sharing structured information about malware**
 - Grammar (Schema)
 - Vocabulary (Enumerations)
 - Collection Format (Bundle)
- **Focus on attributes and behaviors**
- **Enable correlation, integration, and automation**

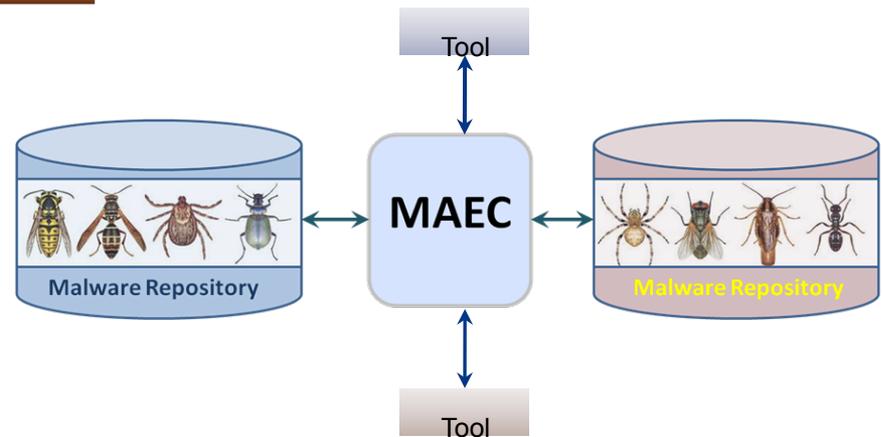
MAEC Use Cases

Operational

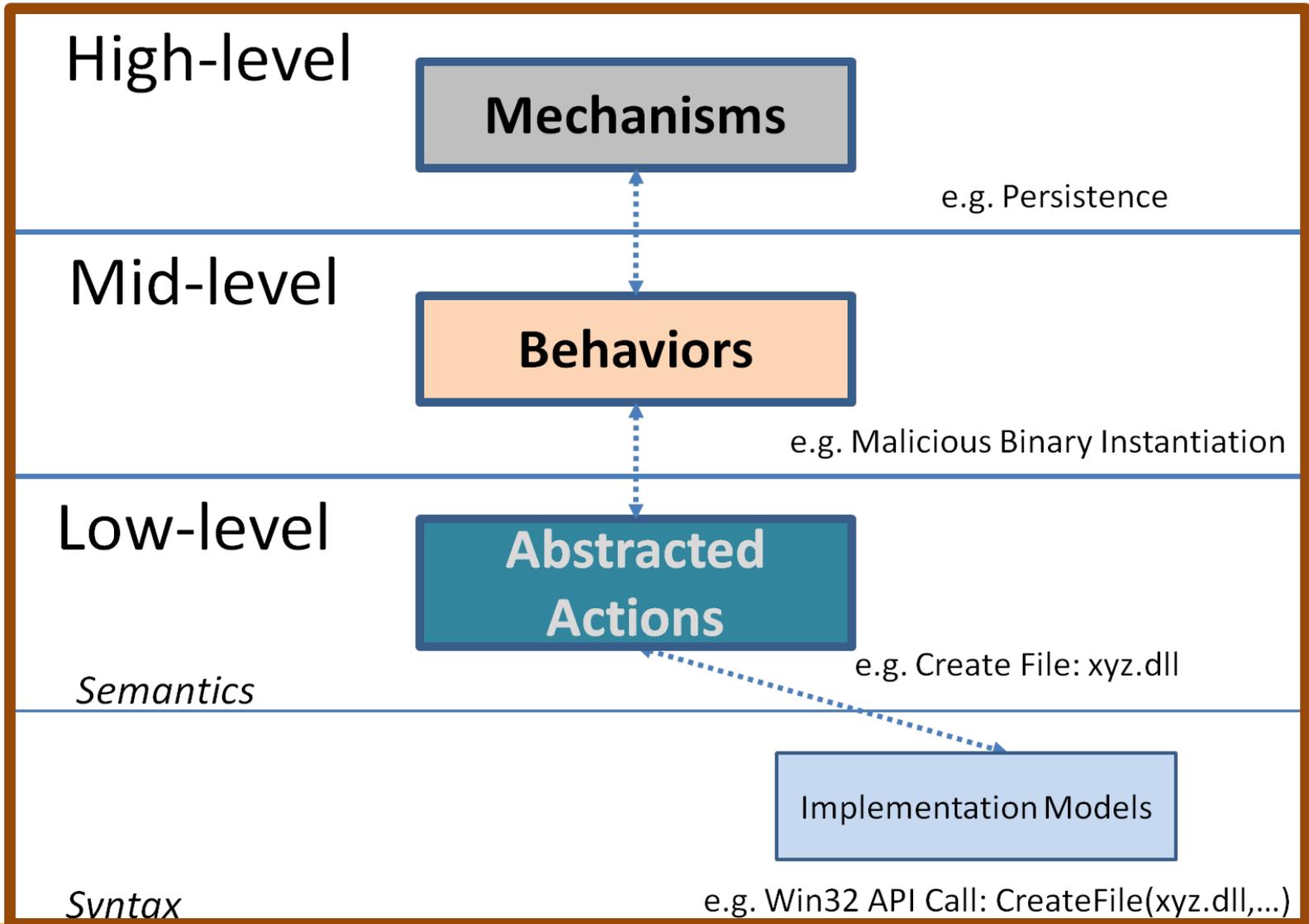


Analysis

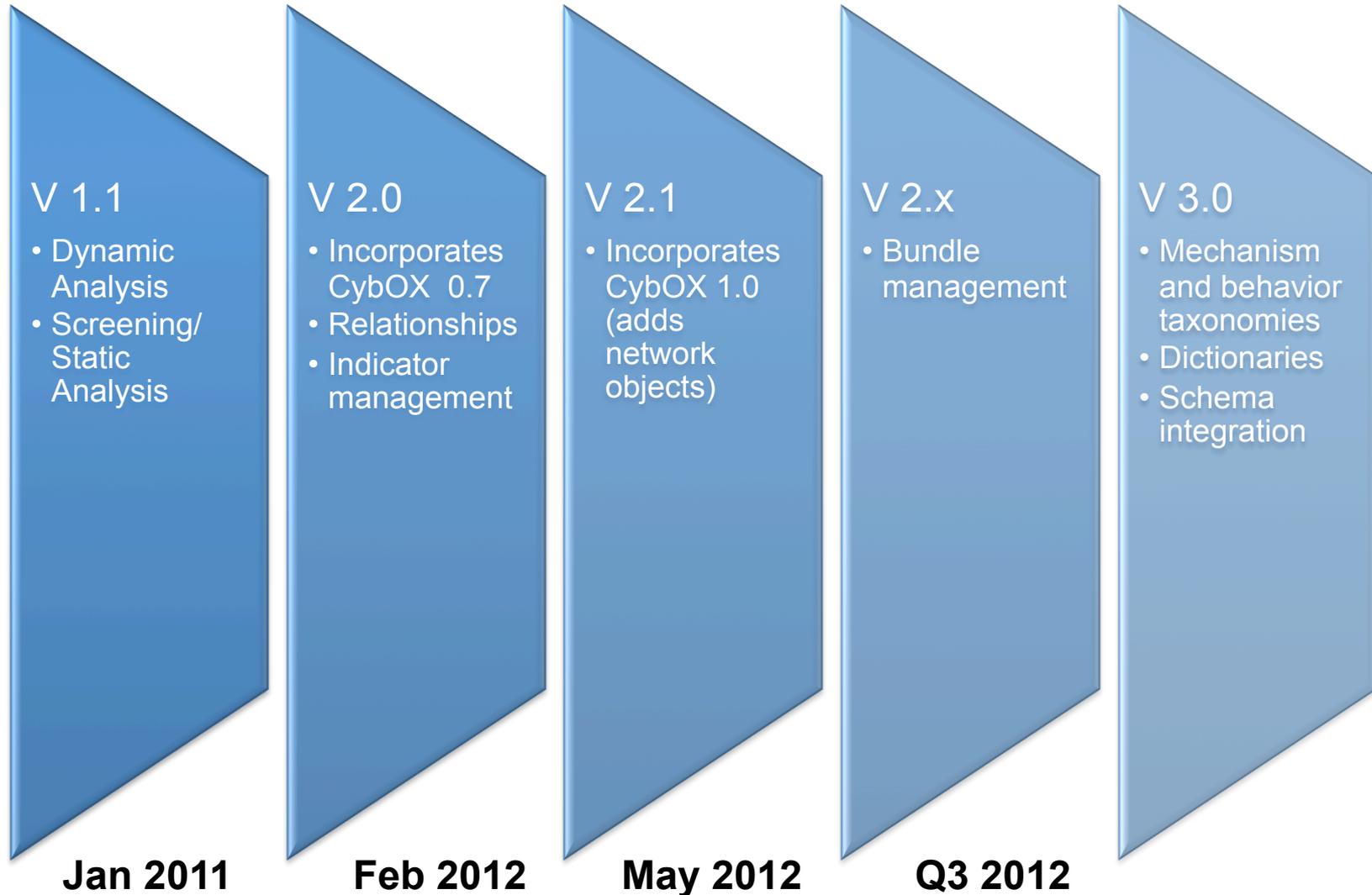
- Help Guide Analysis Process
- Standardized Tool Output
- Malware Repositories



MAEC Overview



MAEC Roadmap



■ What is a cyber observable?

– a *measurable event or stateful property* in the cyber domain

- Some measurable events: a registry key is created, a file is deleted, an http GET is received, ...
- Some stateful properties: MD5 hash of a file, value of a registry key, existence of a mutex, ...

■ **Cyber Observable eXpression (CybOX)** is a standardized language for encoding and communicating information about cyber observables (<http://cybox.mitre.org>)



CybOX Principles

■ Targeted Gestalt

- CybOX is not targeted at a single cyber security use case but rather is intended to be flexible enough to offer a *common solution* for all cyber security use cases requiring the ability to deal with cyber observables. It is primarily intended as a piece of information infrastructure that domain-specific standards and solutions can build upon.

■ Flexibility to express both instances and apriori potential patterns

- It is also intended to be flexible enough to allow both the high-fidelity description of instances of cyber observables that have been measured in an operational context as well as more abstract patterns for potential observables that may be targets for observation and analysis apriori.

■ Integrated automation vision

- By specifying a common structured schematic mechanism for these cyber observables, the intent is to enable the potential for detailed automatable sharing, mapping, detection and analysis heuristics.

Wide Range of Supported Use Cases

- **Use Case Area: Event Management**
 - Producing Event Data (Log-Centric and Non-Log-Centric)
 - Exchanging Event Data
 - Analyzing Event Data
 - Querying Event Data
 - Composing Events
- **Use Case Area: Attack Patterns and Threat Characterization**
 - Characterizing Observable Evidence of Granular Attacker Actions
 - Characterizing Observable Evidence of Attacker Preparatory Probing Techniques
 - Characterizing Observable Evidence of Attacker Obfuscation Techniques
 - Characterizing Observable Evidence of Abstract Attack Patterns
- **Use Case Area: Cyber Threat Indicator Sharing**
 - Generating Cyber Threat Indicators
 - Exchanging Cyber Threat Indicators

Wide Range of Supported Use Cases (cont.)

- **Use Case Area: Attack Detection**
 - Detecting Dynamic In-Progress Attacks
 - Detecting Past Attacks
- **Use Case Area: Incident Investigation**
 - Correlating Incident Initiation Data
 - Excavating Incident Context
- **Use Case Area: Malware Analysis & Management**
 - Analyzing Malware Instances
 - Analyzing Malware Patterns
 - Hunting Malware Artifacts
 - Metadata Indexing Malware Collections
 - Exchanging Malware Characterizations
- **Use Case Area: Digital Forensics**
 - Conducting Digital Forensic Analysis
 - Managing Evidentiary Process

CybOX - Expressivity

- Large number of objects defined and is user-extensible
- Each object has a rich set of (optional) elements
- Object patterns can be expressed as arbitrary Boolean expressions using AND, OR, NOT
- Comparisons supported include relational operators, InSet, InRange, regexes, etc.

CybOX v1.0 Objects

- Account
- Address
- Disk
- Disk Partition
- DNS Entry
- DNS Cache
- Email Message
- File
- GUI
- GUI Dialog Box
- GUI Window
- Library
- Linux Package
- Memory
- Mutex
- Network Connection
- Network Flow
- Network Route
- Network Subnet
- Network Packet
- Pipe
- Port
- Process
- Product
- Semaphore
- Service
- Socket
- System
- Unix File
- Unix Network Route
- Unix Pipe
- Unix Process
- Unix User Account
- Unix Volume
- URI
- User Account
- User Session
- Volume
- Win Computer Account
- Win Critical Section
- Win Driver
- Win Event
- Win Event Log
- Win Executable File
- Win File
- Win Kernel
- Win Kernel Hook
- Win Handle
- Win Mailslot
- Win Mutex
- Win Pipe
- Win Network Route
- Win Network Share
- Win Prefetch
- Win Process
- Win Registry
- Win Semaphore
- Win Service
- Win System
- Win System Restore
- Win Task
- Win Thread
- Win User Account
- Win Volume
- Win Waitable Timer
- X509 Certificate

(more on the way)

CybOX: Resources

- **Resources (released under New BSD license)**
 - Snort -> CybOX
 - OpenIOC -> CybOX and CybOX -> OpenIOC
 - CybOX -> OVAL
 - Full set of Python bindings for CybOX
 - Email -> CybOX parsing tool

Where is CybOX today?

- **CybOX Version 1.0(draft) released mid-April 2012**
 - Version 1.0 should be released in the next few weeks
- **Currently integrated into CAPEC**
- **Currently integrated into MAEC**
- **Currently integrated into STIX**
- **In process of being integrated into CEE**
- **Currently being elaborated for Digital Forensics**
- **Part of the strategic vision for IR/IM with US-CERT**
- **Currently in initial operational use for threat indicator sharing among some communities**

Summary

- **Structured understanding of cyber threat is critical to effective continuous monitoring of software assurance**

- **All resources are free and open for use**

- **Where to learn more:**
 - **<http://capec.mitre.org>**
 - **<http://capec.mitre.org/community/registration.html>**

 - **<http://maec.mitre.org>**
 - **<http://maec.mitre.org/community/discussionlist.html>**

 - **<http://cybox.mitre.org>**
 - **<http://cybox.mitre.org/community/registration.html>**