A CSIRT Process Model for Improving Information Sharing & Knowledge Capture in Cybersecurity

US-CERT

Tom Millar
November 2010

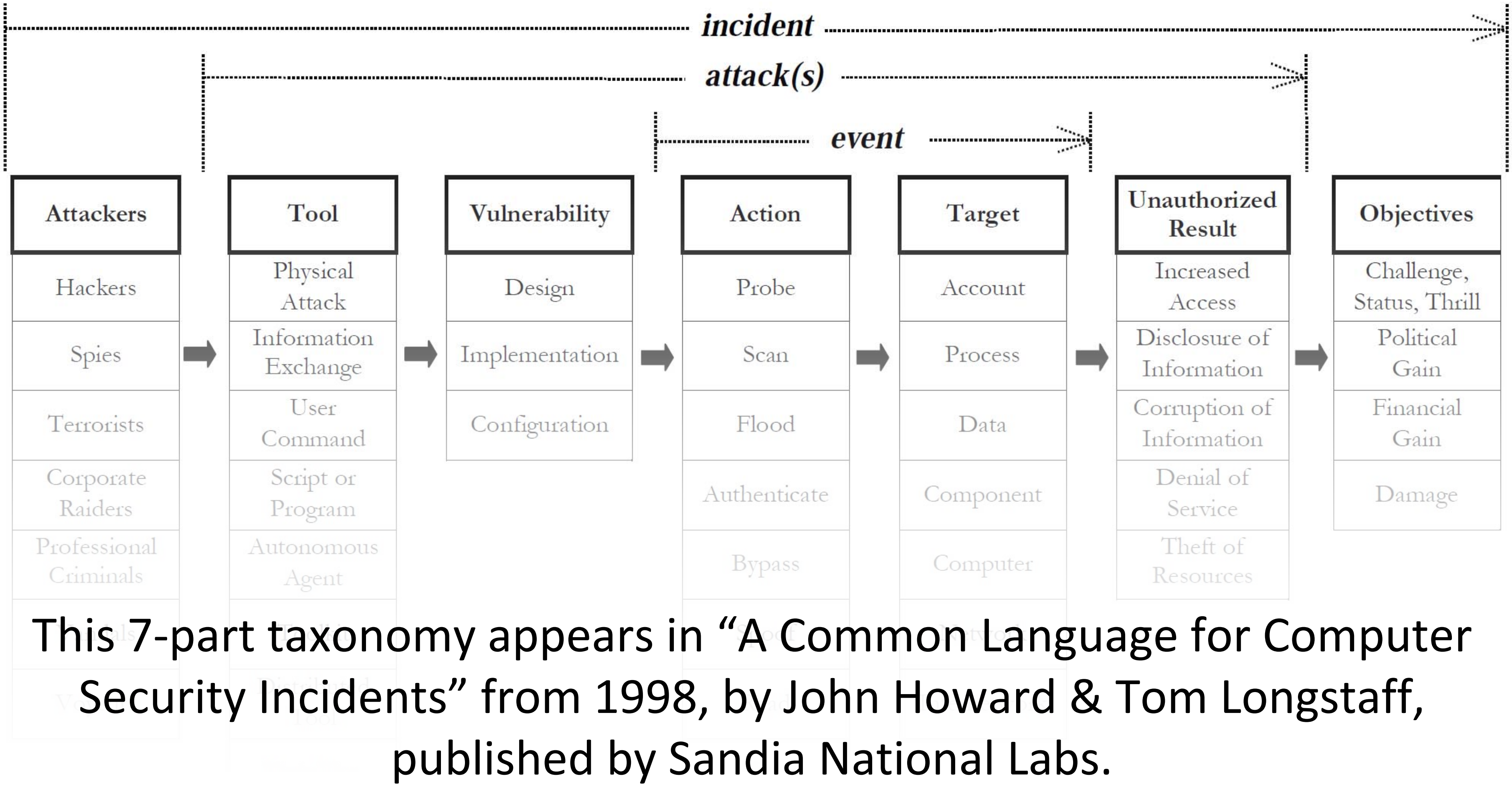
thomas.millar@us-cert.gov

Problem

Information sharing is a priority for all cyber security organizations – not only between internal functional groups, but externally with partners both public and private, at home and abroad.

However, increased information sharing is of little benefit without a shared perspective and vocabulary among the participants.

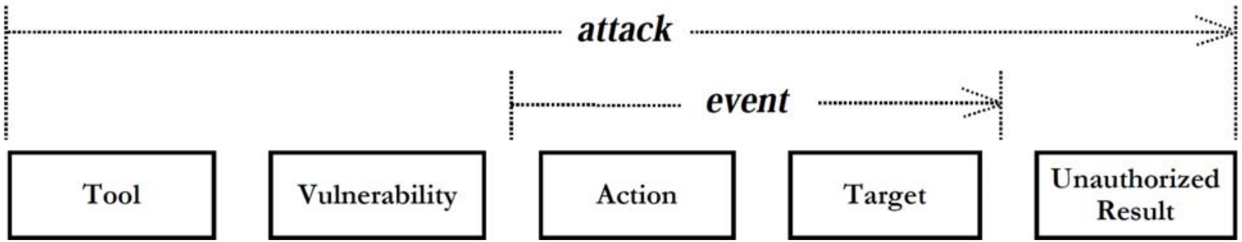
Starting Point: Howard & Longstaff

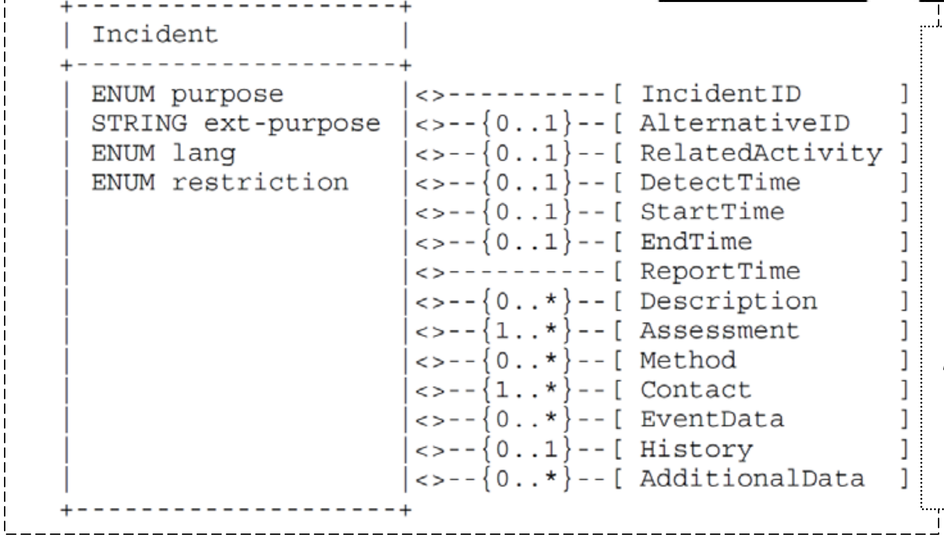


Attackers	Tool	Vulnerability	Action	Target	Unauthorized Result	Objectives
Hackers	Physical Attack	Design	Probe	Account	Increased Access	Challenge, Status, Thrill
Spies	Information Exchange	Implementation	Scan	Process	Disclosure of Information	Political Gain
Terrorists	User Command	Configuration	Flood	Data	Corruption of Information	Financial Gain
Corporate Raiders	Script or Program		Authenticate	Component	Denial of Service	Damage
Professional Criminals	Autonomous Agent		Bypass	Computer	Theft of Resources	
Vandals	Toolkit		Spoof	Network		
Voyeurs	Distributed Tool		Read			

This 7-part taxonomy appears in “A Common Language for Computer Security Incidents” from 1998, by John Howard & Tom Longstaff, published by Sandia National Labs.

One Implementation: The IODEF





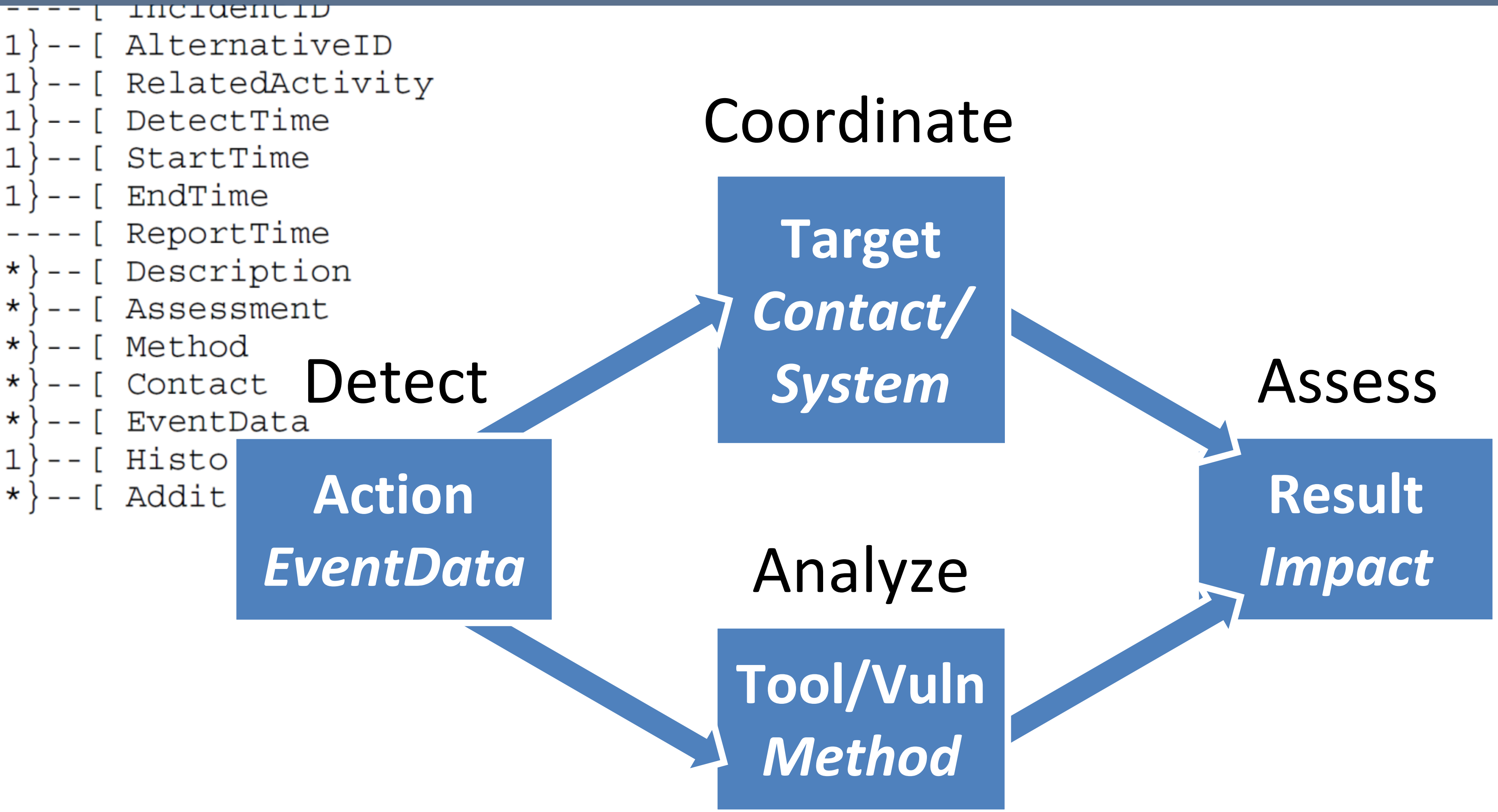
```
RFC 5070                                  IODEF

   +------------------+
   | Incident         |
   +------------------+
   | ENUM purpose      |<>----------[ IncidentID        ]
   | STRING ext-purpose|<>--{0..1}--[ AlternativeID     ]
   | ENUM lang         |<>--{0..1}--[ RelatedActivity   ]
   | ENUM restriction  |<>--{0..1}--[ DetectTime        ]
   |                   |<>--{0..1}--[ StartTime         ]
   |                   |<>--{0..1}--[ EndTime           ]
   |                   |<>----------[ ReportTime        ]
   |                   |<>--{0..*}--[ Description       ]
   |                   |<>--{1..*}--[ Assessment        ]
   |                   |<>--{0..*}--[ Method            ]
   |                   |<>--{1..*}--[ Contact           ]
   |                   |<>--{0..*}--[ EventData         ]
   |                   |<>--{0..1}--[ History           ]
   |                   |<>--{0..*}--[ AdditionalData    ]
   +------------------+
```

- **Tool** and **Vulnerability** become ***Method***
- **Action** becomes ***EventData***
- **Target** is part of both ***Contact*** (the PoC) and ***EventData*** (the system)
- **Unauthorized Result** becomes ***Assessment***

5 of the 7 parts from Howard & Longstaff's taxonomy were adopted as data element classes in the IETF's Incident Object Definition Exchange Format, an XML schema for cyber incident reporting.

Map Taxonomy to Process



```
----[ IncidentID
1}--[ AlternativeID
1}--[ RelatedActivity
1}--[ DetectTime
1}--[ StartTime
1}--[ EndTime
----[ ReportTime
*}--[ Description
*}--[ Assessment
*}--[ Method
*}--[ Contact
*}--[ EventData
1}--[ Histo
*}--[ Addit
```

Coordinate

Target
Contact/
System

Detect

Action
EventData

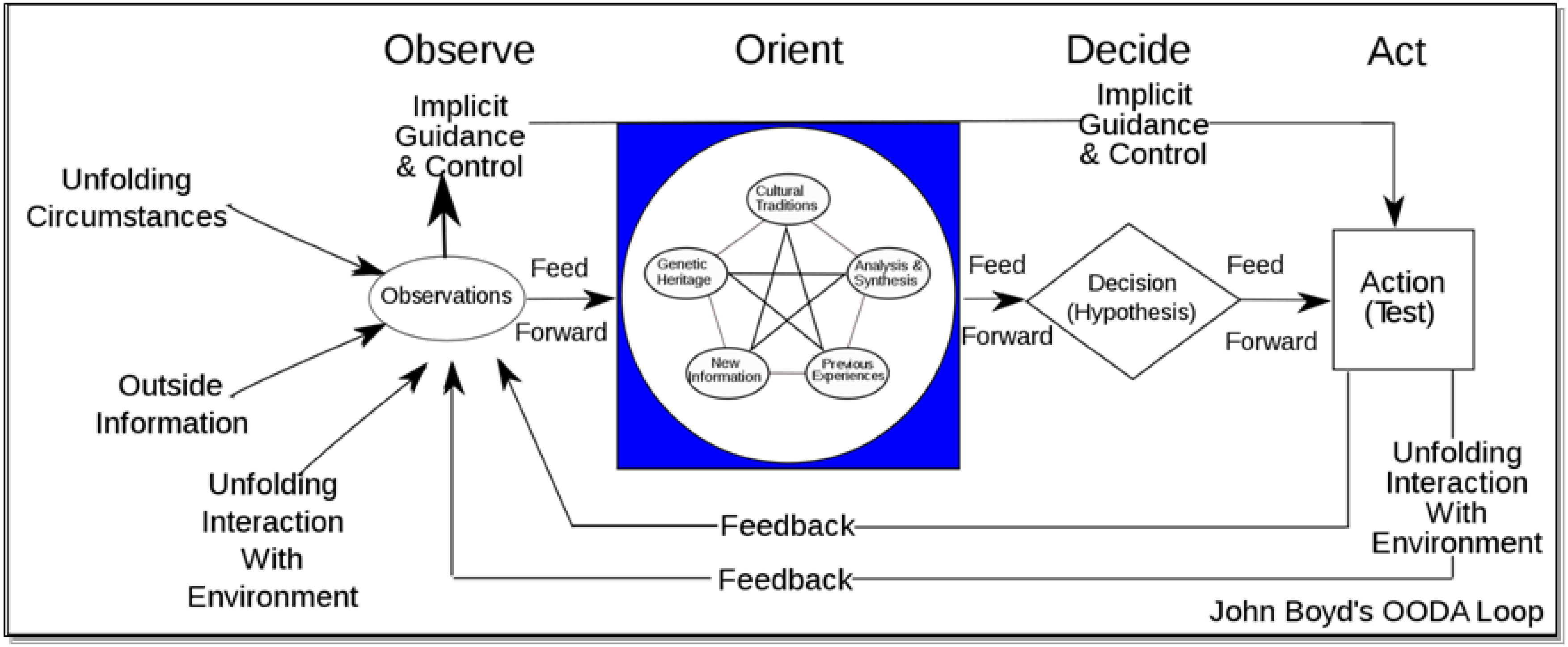
Assess

Result
Impact

Analyze

Tool/Vuln
Method

…and Rediscover the OODA Loop



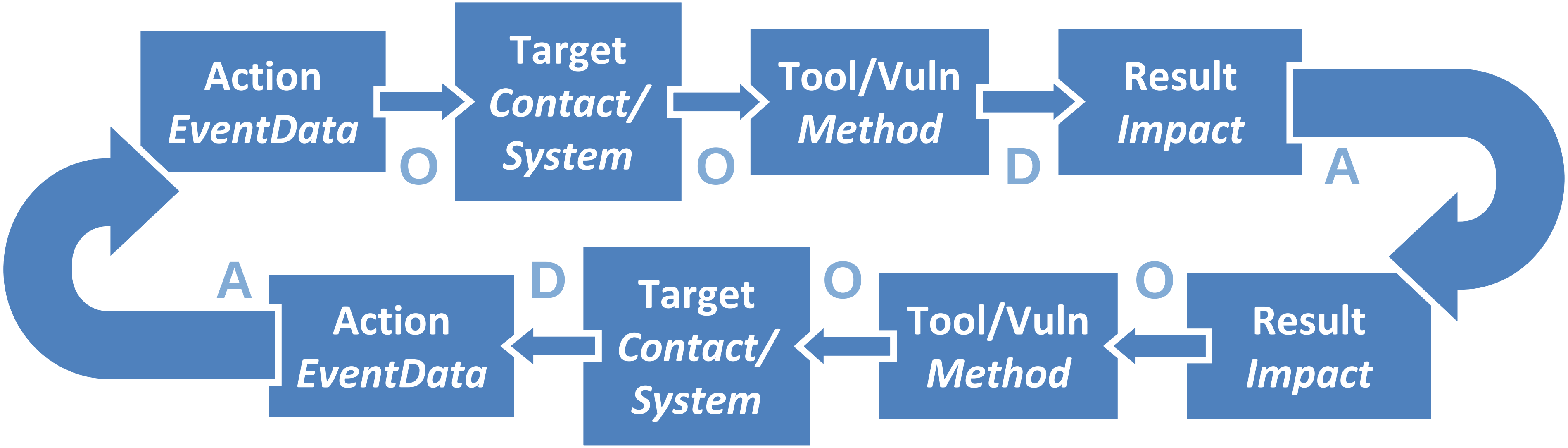
John Boyd's OODA Loop

Detect – Coordinate & Analyze – Assess…

<u>That's Nice, But:</u> Now what?

Extending the Loop:

Detect – Coordinate – Analyze – Assess



Action *EventData*

O

Target *Contact/ System*

O

Tool/Vuln *Method*

D

Result *Impact*

A

Result *Impact*

O

Tool/Vuln *Method*

O

Target *Contact/ System*

D

Action *EventData*

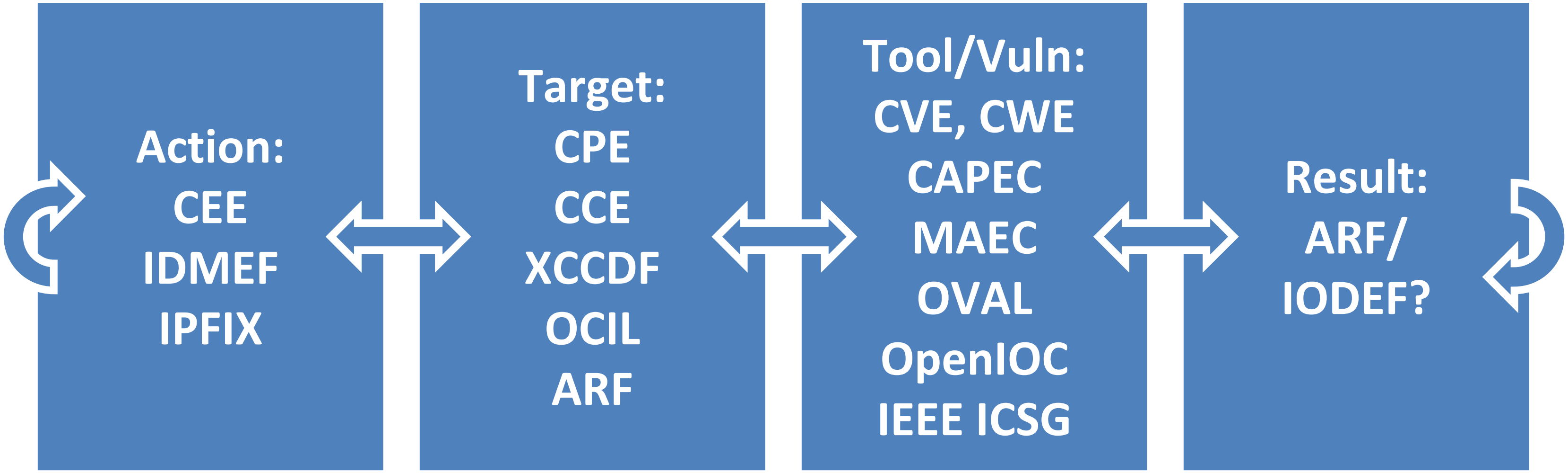
A

Monitor – Mitigate – Determine – Confirm

Chaining two “OODA” loops together gives us a better picture of a complete incident management process from identification to recovery, and most importantly, ***using the same language throughout***

Extending the Data Model:

Detect – Coordinate – Analyze – Assess



Action:
CEE
IDMEF
IPFIX

Target:
CPE
CCE
XCCDF
OCIL
ARF

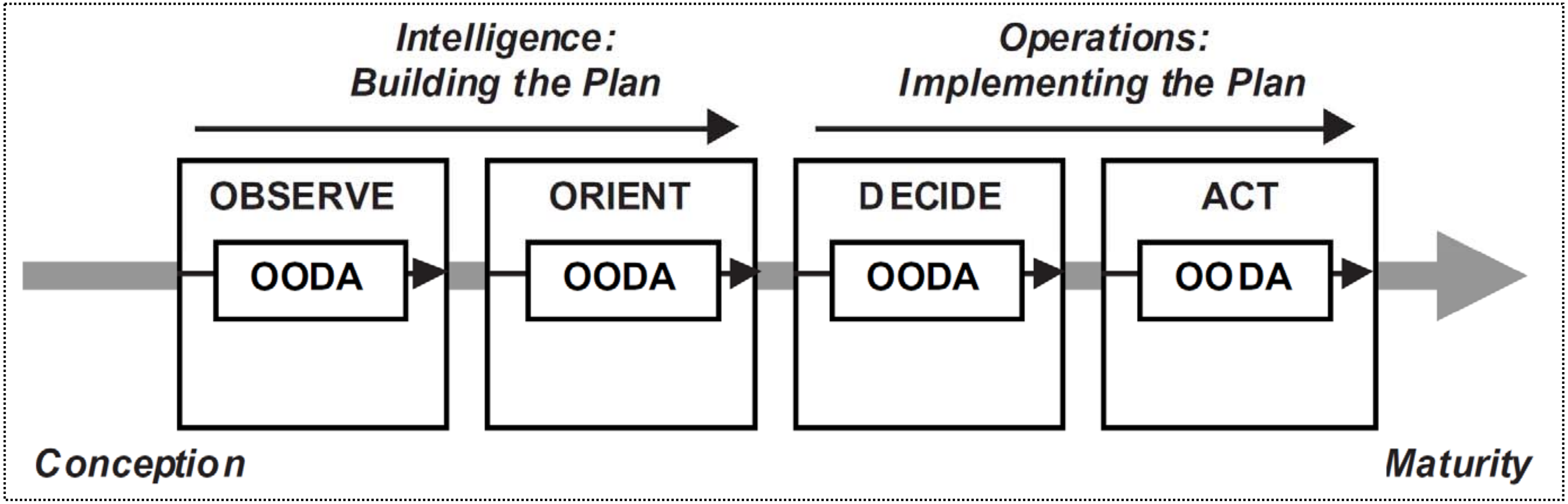
Tool/Vuln:
CVE, CWE
CAPEC
MAEC
OVAL
OpenIOC
IEEE ICSG

Result:
ARF/
IODEF?

Monitor – Mitigate – Determine – Confirm

Existing and emerging specifications could be wrapped in a larger IODEF document type – when completely “filled out” in the course of the process loop, we begin capturing **more and more reusable knowledge**

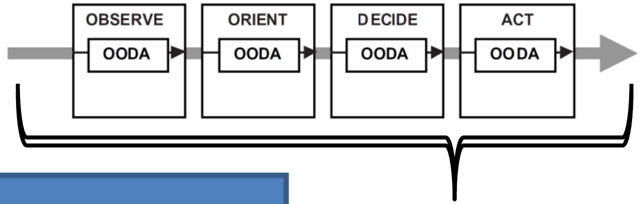
Organizational Challenges:



John Bodnar’s “Warning Analysis For The Information Age” describes how nested or chained OODA loops function in organizations.

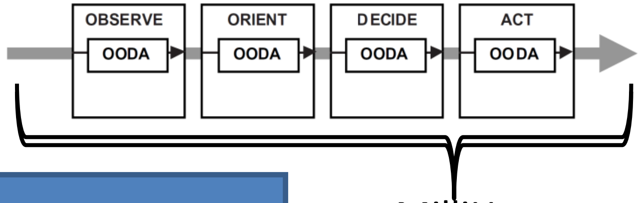
In the above example, “***Intelligence***” might be a CERT/CSIRT/SOC; “***Operations***” then becomes the NOC/ITS team. Patch this! Block that!

Differences in Octave/Register



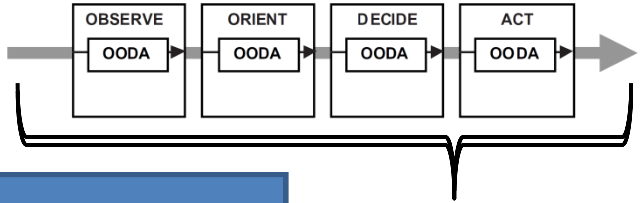
Ops

DeciHz (Fractions of Minutes)



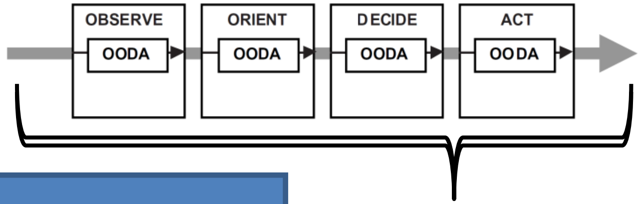
IT/Sec

MilliHz (Minutes to Hours)



Mgmt

MicroHz (Hours to Days)

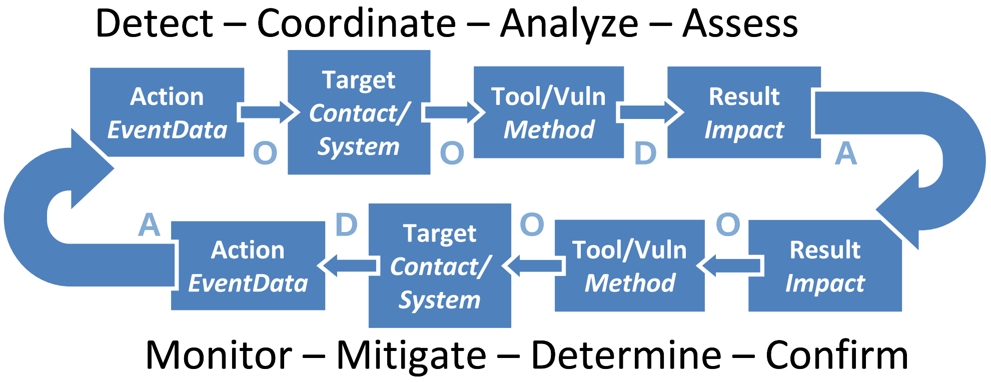


Policy

NanoHz (Days to Months)

Each layer in an organization has their own loop frequencies. Business Ops act at Push-To-Talk speed; IT & Security act at E-mail speed; Management & Policy may act at Memorandum speed or slower!

Conclusions

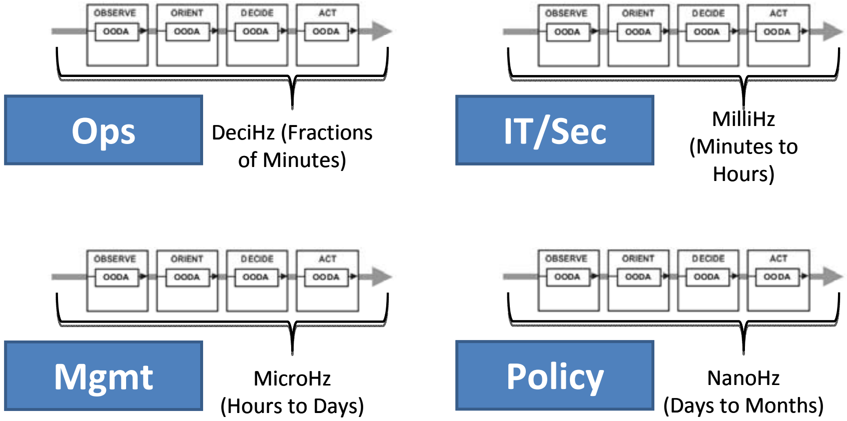


Detect – Coordinate – Analyze – Assess

Action EventData O *Target Contact/ System* O *Tool/Vuln Method* D *Result Impact* A

A *Action EventData* D *Target Contact/ System* O *Tool/Vuln Method* O *Result Impact*

Monitor – Mitigate – Determine – Confirm



Ops DeciHz (Fractions of Minutes)

IT/Sec MilliHz (Minutes to Hours)

Mgmt MicroHz (Hours to Days)

Policy NanoHz (Days to Months)

For information sharing to be minimally useful, we have to speak a common language (a language is more than a vocabulary).

For information sharing to be optimal, we have to understand how each part of that language plays its role in the greater cyber security process: incident identification, mitigation, recovery, knowledge capture and - eventually - developing safer code and protocols.