



Technical Steering Committee

What is the OCA Ontology?

The OCA Ontology is the up-front design rules and a baseline definition of event topics, API specifications, services and functions of information technology systems and related security technologies required to achieve an Open, Event-Driven, Heterogenous Distributed systems, service oriented architecture.

• The Ontology governs the services, exposed functions or features of things in the architecture, data, API formats, schema

•

- For Search, a repository of cyber-observables, IoC or IoB should conform to X standard.
- For repositories or technologies with repositories that do not, OCA Community will produce adapters to adapt the non- comformant technology to the OCA Ontology
 - STIXShifter
 - OpenC2 Adapt Non-Comformant Command API to OpenC2
 - SCAP v2 -

Status Quo

- What we and customers are dealing with.
- What open standards are out there today?
- The Technology Alliances Problem,
- The Product Engineering Problem
- Agile vs. Waterfall, SaaS, PaaS, Clouds, Devices, People
- The up-front design Conway's Law

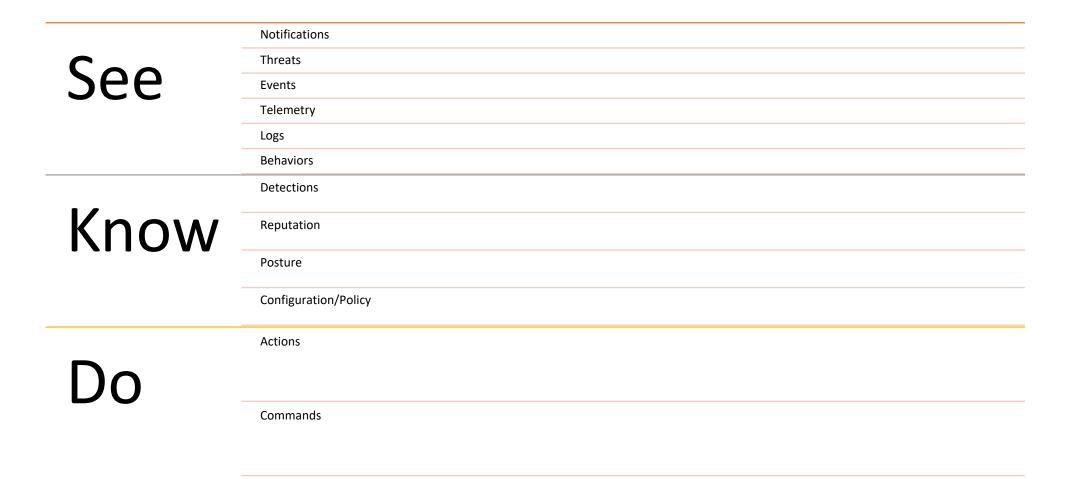
Principles for Terminology

- 1. Include in the terminology terms used in authorized glossaries
 - 2. Strive to ensure maximal consensus with the authorized usage
- 3. Identify areas of disciplinary overlap where terminological usage is not
- consistent
- 4. In terminology construction and ontology design, make use of as many existing resources (terminologies and ontologies) as possible.
- 5. Use singular nouns.
- 6. Use lowercase for common nouns.
- 7. Avoid acronyms.
- 8. Associate each term in the ontology with a unique alphanumeric identifier.
- 9. Ensure univocity of terms (unambiguous terms). 10. Ensure univocity of relational expressions.
 - 11. Avoid mass terms.

- 13. Provide all non-root terms with definitions
- 14. Use Aristotelian definitions
- 15. Use essential features in defining terms.
- 16. Start with the most general terms in your domain. 17. Avoid circularity in defining terms.
- 18. To ensure the intelligibility of definitions, use simpler terms than the term you are defining. 19. Do not create terms for universals through logical combination.
- 20. Definitions should be unpackable (Term-definition intersubstitutability)
- 21. Structure every ontology around a backbone *is_a* hierarchy.
- 22. Ensure *is_a* completeness.
- 23. Ensure asserted single inheritance.
- 24. Both developers and users of an ontology should respect the open-world assumption.
- 25. Adhere to the rule of objectivity, which means: describe what exists in reality, not what is known about

What can or should 'things':

See, Know, Do



Event Driven Thinking

Given	State	Known Vulnerabilities	
		Known Software Inventory	
		Known User Risk Context	Location
	An Frant Occurs	Sightings	
When	An Event Occurs	File Reputation Change	
		Analytic Result	SIEM Correlation UEBA User Risk State Change
		Vulnerability Discovered	
		User Download Request	
Then	Action/Command	SCAP Vulnerability Scan	
		Delete File	
		Publish Notification	
		Block Web Request	

"Show me your flowcharts and conceal your tables, and I shall continue to be mystified. Show me your tables, and I won't usually need your flowcharts; they'll be obvious" – Fred Brooks, author Mythical Man-Month

User Interface – Beginning with what the user will see

Events – Events Happen. Something is seen, something is learned or now known,

Views – Independent of storage, we can now use the event model to analyze the system from the point of view of state.

Orchestration, Choreography, Automation

Where do we start?

Technology Categories

Ref Gartner MQ and Critical Capability for Tech Categories - https://www.gartner.com/en/research/magic-quadrant

Cloud Access Security Brokers	
Hyperconverged Infrastructure	
Identity Governance and Administration	
Integrated Risk Management Solutions	
Secure Web Gateways	
Enterprise Network Firewalls	
Security Information and Event Management	
Unified Threat Management	
Web Application Firewalls	
Ticketing	
EDR?	
Vulnerability Management	
UEBA	
Threat Intelligence Platform or Service	
Endpoint Protection Platforms	

Technology Categories

SANS CIS Critical Controls

nventory and Control of Enterprise Assets	
nventory and Control of Software Assets	
ata Protection	
ecurity Configuration of Enterprise Assets and Software	
ccount Management	
ccess Control Management	
ontinuous Vulnerability Management	
udit Log Management	
mail and Web Browser Protections	
Nalware Defenses	
ata Recovery	
letwork Infrastructure Management	
letwork Monitoring and Defense	
ecurity Awareness and Skills Training	
ervice Provider Management	

Technology Categories

Endpoint Protection	EPP
	EDR
	OSQuery?
Network Protection	Firewall
	Network IDS/IPS
	Flow Sensor

Web Protection

• • •