



Position Paper

Collaboration Oriented Architectures

Introduction

Collaboration Oriented Architectures (COA) describes information architectures that comply with the COA framework, which in turn is described in the associated COA Framework paper¹. COA-compliant information architectures enable enterprises that use them to operate in a secure and reliable manner in an environment of increasing information threat, and where it is the growing norm to interact without boundaries, irrespective of the location of the data or the number of collaborating parties.

This paper explains the COA concept and sets out the essential requirements that are needed in a COA-compliant information architecture. The COA Framework paper presents an architectural view of the principal components in a COA-compliant architecture. Supporting papers then describe these components as required.

While many organizations are trying to respond to the de-perimeterization issue, they often lack a framework and set of guiding principles to organize and implement specific solutions. This paper and its associated COA Framework paper, together with the Jericho Forum “design principles” (commandments²) aims to fill this gap.

This paper focuses on the need to have business processes that operate across and between multiple organizations, probably (but not necessarily) using the Internet as the common transport mechanism. In this environment, users and end-systems must securely interact with, or use services from, disparate systems that are outside any single locus of control or security domain.

Implementing a COA entails adoption of the Jericho Forum Commandments (JFC) (specifically Jericho Forum design principles #4 to #8³) covering the areas of operating in a hostile environment, trust, and authentication.

¹ COA Framework paper available from <http://www.opengroup.org/jericho/publications.htm>

² Jericho Forum Commandments: available at http://www.opengroup.org/jericho/commandments_v1.2.pdf

³ JFC#4 - Devices and applications must communicate using open, secure protocols

JFC#5 - All devices must be capable of maintaining their security policy on an un-trusted network

JFC#6 - All people, processes, technology must have declared and transparent levels of trust for any transaction to take place

JFC#7 - Mutual trust assurance level must be determinable

JFC#8 - Authentication must interoperate/exchange outside of your locus of control

Problem

The traditional electronic boundary between a corporate (or ‘private’) network and the Internet is breaking down in the trend which we have termed de-perimeterization.

Traditional approaches to architecting security solutions are aimed at securing organizational borders, and then the network, reinforcing a ‘perimeterized’ perspective. This is contrary to the future business needs of most organizations:

- Business is demanding more connectivity outside the enterprise
- Commoditization of technology is driving towards any-to-any connectivity on every electronic device, with those devices having ever lower cost with more ‘intelligent’ functionality built-in
- Business ‘relationships’ of every type, from subsidiaries to relationships with other business that are also competitors in other areas, all require connectivity
- Pervasive, fast, reliable, cheap Internet connectivity is becoming available everywhere

Responding to the trend of de-perimeterization with a COA-compliant architecture allows the business aspirations to be met by positioning processes and security controls appropriate to risks and needs, away from the traditional firewalls or gateways that organizations have turned to in the past to ‘solve’ the business demand for secure collaboration.

The COA framework defines the key components within which interoperable, secure solutions can be provided to meet the needs of the business. Thus, systems, networks and whole ‘enterprise architectures’ can be considered to be compliant with the COA framework if all the components defined in the framework are present.

A COA-compliant architecture enables provision of IT systems that are secure in a global networked world, able to keep pace with the growing threats and the business need for faster and more flexible collaborative business arrangements. These range from outsourcing to joint ventures, from merger today to divestment tomorrow, all within a global working, global manufacturing and global procurement environment.

Why I should care?

De-perimeterization describes a problem driven by business and commercial pressures. It does not, in itself, suggest any solutions. The COA framework allows appropriately architected business-driven solutions to be developed and delivered. De-perimeterization is happening now, will continue to happen, and will inevitably impact virtually all networked IT systems. Implementing a COA-compliant architecture will ensure that de-perimeterization does not magnify the risks to your organizations.

Recommended Solution/Response

The COA framework generalizes conventional architectures. It provides:

- increased emphasis on the requirements listed in the COA framework as ‘principles’. These are traditionally only seen as external or ‘boundary’ interface concerns in enterprise architectures.
- a user repository (keyed on people identifiers), generalized into a contract repository (keyed on relationship, or obligation identifiers). A contract repository records agreements, and the obligations and capabilities that ensue from them.
- an accounting log (keyed on system events), generalized into a reputation repository (keyed on business events). A reputation repository records user actions and compares

them to applicable contracts, and, depending on whether or not the actions are in accordance with the contract, upgrades or downgrades a reputation.

The architecture formed by combining SOA (Service Oriented Architectures) with available security protocols (SAML or other XML) is insufficient to support COA. The following elements are also valuable⁴:

- The Standard Security Management System ISO/IEC 27001.
- Business processes that manage the collaborations founded on practises found in COBIT.
- Service Management capabilities detailed in ITIL.
- The architecture capabilities defined in TOGAF.
- A powerful language for describing access policies and delegations (XACML version 3.0 is a promising candidate.)
- Access managers that will enforce an externally-required or end-to-end policy. Current access management systems are beginning to gain this capability.
- Attribute brokers that will establish a requester's identity, credentials and attributes to an appropriate degree of confidence, based on information from multiple authoritative sources (e.g. attribute authorities).
- Performance managers that will record what a user or system does at the level of business events, judge whether the user or system has acted in accordance with a contract or other agreed obligation, and report on their compliance profile. Today, this is a rather neglected field. It includes audit log managers and reputation systems.
- Contract brokers that will negotiate and agree new collaborative understandings between collaborating individuals in ways which do not violate their 'owning' organization's and jurisdiction's existing policies and contracts. These new contracts must be expressed in an open-standard language which can be interpreted by performance managers and access managers – eBXML is a strong candidate. The contract brokers must be able, in turn to read the open-standard output language of the performance managers and attribute brokers.

Conclusion

Implementing a COA-compliant architecture builds a high level business framework that uses the capabilities of SOA, in addition to other relevant standards and practises, to enable effective and secure collaboration. While a SOA meets many of the functional and non-functional requirements of COA, other standards and practises such as TOGAF, COBIT and ITIL also need to be engaged.

A fundamental shift in thinking is required to implement a COA-compliant architecture, moving from the thinking of a hedgehog - an animal that rolls into a tight prickly perimeterized ball at any sign of threat - to that of a strawberry plant, which puts all its key genetic material securely on its outside, as well as sending out suckers to extend the plant's domain. The COA framework paper also provides a high level pattern for how a previously developed information system can be re-architected to support effective and secure collaborations across corporate boundaries. Enterprises that want to operate in a network of business partners will do well to implement a COA-compliant architecture, and encourage their partners to do likewise.

⁴ Note that we include mention of brokers and repositories. While these are not strictly within the intended scope of this paper, they are mentioned because of their importance in the complete picture.

The way forward

Jericho Forum members encourage the development and definition of appropriate open-standard interfaces between the COA framework's architectural elements. A key development area is to define at the semantic level the meaning of trust/confidence and Trust Management to assure secure business collaboration between corporations.

These dependencies are addressed in the associated COA Framework paper, in terms of identifying the principal components in an "architects view" diagram, and providing a requirements description for each component. Where necessary, additional supporting COA component papers provide more detailed descriptions for specific components.