# Security Principles for Cloud and SOA

*A White Paper by:*

The Security for the Cloud & SOA Project of
The Open Group Cloud Computing Work Group

December 2011

Security Principles for Cloud and SOA

Document No.: W119

Published by The Open Group, December 2011.

Any comments relating to the material contained in this document may be submitted to:

The Open Group, 44 Montgomery St. #960, San Francisco, CA 94104, USA

or by email to:

ogspecs@opengroup.org

# Table of Contents

**Executive Summary** 4

**Nature of Security Principles** 5

**Setting the Principles in Context** 6

**Drivers for Security** 7
- Risk....7
- CIA....7

**Format of Principles** 8

**General Security Principles** 9
- Fundamentals....9
- Trust....11
- Data Protection....12
- Management....13
- Interoperability....15

**Cloud and SOA Security Principles** 16

**Other Sources** 19
- ISO/IEC 27001....19
- NIST....19
- Joint ISF, (ISC)2, and ISACA....19
- OECD....20
- The Jericho Forum®....20

**About the Authors** 21
- Lead Author....21
- Base Document....21
- Lead Reviewers....21
- Other Reviewers....21

**About The Open Group** 21


*Boundaryless Information Flow™*
*achieved through global interoperability*
*in a secure, reliable, and timely manner*

## Executive Summary

This White Paper is part of a series on Security in Cloud and SOA computing, which provides guidelines for Enterprise Architects involved in developing risk-based approaches to securing Cloud and SOA computing. In outline, our approach is to gather requirements and from them build architectures to meet those requirements, adhering to principles where relevant, and documenting architectural decisions as appropriate.

It is strongly based on key work deliverables from the Security for the Cloud & SOA Project in The Open Group – which is a joint project between the Cloud Computing and SOA Work Groups and the Security Forum, and aligns with the Jericho Forum Commandments.

This White Paper presents a set of Security Principles. The Security for the Cloud & SOA Project is publishing them as a White Paper to ensure that they achieve the public prominence and visibility that they merit as essential guidelines on the general requirements necessary to assure effective information security.

People use "principles" as a test to check whether their decisions are sound and address all the aspects involved. In designs, we use principles as a validation tool to check we are doing things right and have not forgotten anything that is material to achieving the design goals. They are especially valuable when there are many aspects/views to a complex design, because they remind us of issues which we find hard to bring to mind when we're deeply involved in such design complexity.

Many of these security principles are general to the secure design of all enterprise architectures, so they are widely applicable as guidance to architecting secure systems in all environments. We have also identified principles that we see as having specific relevance to securing Cloud and SOA environments, so we list these as additions to the General Security Principles.

## Nature of Security Principles

Security Principles define key design features of information security that should be applied when architecting a secure architecture (or framework, or infrastructure). These features must:

- Be defined in plain language
- Use terms that have clear meaning within the context being used
- Be technology-neutral; i.e., independent of any technologies or design implementation

They serve as a benchmark against which security concepts, designs, solutions, compliance with standards and applicable regulations, and system architectures, can be methodically assessed and measured. An example of how this may be done is the Jericho Forum's Self-Assessment Scheme.[1]

[1] Jericho Forum Self-Assessment Scheme; available at: www.opengroup.org/jericho/SAS_Guide.pdf.

## Setting the Principles in Context

As will be clear from reviewing these Security Principles, they are by no means all exclusively relevant to SOA and Cloud architectures, but they may well represent specific challenges in those environments. They are also just a part of the complete body of work being developed by the Security for the Cloud & SOA Project.

This project has also defined a set of Architecture Building Blocks (ABB) applicable to SOA and Cloud (but potentially also to other situations). In our growing series of White Papers explaining our scenario-based approach:

- An Architectural View of Security for Cloud[2] – examining policy-based security through scenarios, using Identity Entitlement and Access Management (IEAM) as an example of policy-driven security concerns, to illustrate an approach that can be applied to other aspects of Security in the Cloud
- Forthcoming Data & Information Protection White Paper
- Forthcoming Cloud Security Architecture White Paper
- Forthcoming Architectural Decisions White Paper

and more, we illustrate possible deployment models for these building blocks across various consumers and providers and show how the principles can be used to help to assess the value and risk of each model in the practical situations experienced by both consumers and providers.

The complexities, particularly in Cloud, of dealing with an ecosystem of other providers and consumers, may at times mean that we cannot easily comply with our own principles. The project has formalized a template for documenting and making architectural decisions in these situations as well as producing a sample illustrative set of decisions.

The completed body of work is intended to help architects develop secure and adaptable solutions for their own organizations' concrete situation and business goals.

[2] An Architectural View of Security for Cloud, White Paper (W116), May 2011, published by The Open Group; available at: www.opengroup.org/bookstore/catalog/w116.htm.

# Drivers for Security

## Risk

Risk of loss is the business driver for security. Enterprises need to perform risk assessments to understand their exposure to risk of loss. This is often referred to as their *risk profile*. Business managers then use the results from risk assessments to make informed business decisions on how to manage their risks of loss – either by accepting each risk, or by mitigating it – through investing in appropriate security protective measures judged sufficient to lower the potential loss to an acceptable level, or by investing in external indemnity.

Critical to enabling good business decision-making therefore is to use risk assessment methods which give objective, meaningful, consistent results.[3]

## CIA

Confidentiality, Integrity, and Availability (CIA) are commonly accepted as capturing the prime objectives for what security measures aim to achieve. The security architect should always keep these three key aspects of information security at the forefront of any solution where data is involved:

- Confidentiality: Data is shared only to authorized actors (the right people).
- Integrity: Data can be assured to be authentic, trustworthy, and complete.
- Availability: The solution ensures access for delivering, storing, and processing data when required.

A key aspect of information and solution security, especially in regard to data, is to preserve these three fundamentals. Failure to deliver a solution which addresses these objectives can result in losses which may even threaten corporate survival. A secure solution requires information access to be appropriately controlled; data to be complete and trusted; and data to be available when required. A documented understanding of the requirements for all three must lie at the forefront of a solution architecture. Care must be taken that the solution meets those demands, but does not become over elaborate; nor favors one over the others.

[3] Risk Taxonomy, Technical Standard (C081), January 2009, published by The Open Group; available at: www.opengroup.org/bookstore/catalog/c081.htm

## Format of Principles

It is useful to have a standard way of defining Principles. In addition to a definition statement, each Principle should have associated rationale and implications statements, both to promote understanding and acceptance of the Principles themselves, and to support the use of the Principles in explaining and justifying why specific decisions are made.

This White Paper follows the format used in TOGAF 9 Chapter 23:

| | |
|---|---|
| **Name** | Should both represent the essence of the rule as well as be easy to remember. Specific technology platforms should not be mentioned in the name or statement of a principle. |
| **Statement** | Should succinctly and unambiguously communicate the fundamental rule. For the most part, the principles statements for managing information are similar from one organization to the next. It is vital that the principles statement be unambiguous. |
| **Rationale** | Should highlight the business benefits of adhering to the principle, using business terminology. Point to the similarity of information and technology principles to the principles governing business operations. Also describe the relationship to other principles, and the intentions regarding a balanced interpretation. Describe situations where one principle would be given precedence or carry more weight than another for making a decision. |
| **Implications** | Should highlight the requirements, both for the business and IT, for carrying out the principle – in terms of resources, costs, and activities/tasks. It will often be apparent that current systems, standards, or practices would be incongruent with the principle upon adoption. The impact to the business and consequences of adopting a principle should be clearly stated. The reader should readily discern the answer to: "How does this affect me?" It is important not to oversimplify, trivialize, or judge the merit of the impact. Some of the implications will be identified as potential impacts only, and may be speculative rather than fully analyzed. |

# General Security Principles

The following Security Principles are common to all designs aiming to assure a secure IT architecture.

For ease of navigation, these General Security Principles are grouped under the following headings. Their order of appearance does not imply priority of importance.

- Fundamentals
- Trust
- Data Protection
- Management
- Interoperability

## Fundamentals

| **Name** | **Policy-driven** |
|---|---|
| **Statement** | Security must be driven by security policy. |
| **Rationale** | Business managers use the results from risk assessments to make informed business decisions on how to manage their enterprise's risks of loss – either by accepting each risk, or by mitigating it through investing in appropriate security measures judged sufficient to lower the potential loss to an acceptable level, or by investing in external indemnity.<br>Security policy should be maintained on untrusted networks; e.g., the Internet. |
| **Implications** | A security policy defines the rules for protection of the enterprise's IT systems. Security architects must work from the security policy of the enterprise to design security measures that are necessary and sufficient for that enterprise. |

| **Name** | **People, Process, and Technology** |
|---|---|
| **Statement** | All people, processes, and technology must have declared and transparent levels of trust for any transaction to take place. |
| **Rationale** | Trust in this context is establishing understanding between contracting parties to conduct a transaction, and the obligations this assigns on each party involved.<br>Trust models must encompass people/organizations and devices/infrastructure. |
| **Implications** | Trust level may vary by location, transaction type, user role, and transactional risk. |

| **Name** | **Open-ness** |
|---|---|
| **Statement** | Information security solutions should depend on open systems mechanisms. |
| **Rationale** | Open-ness is of primary importance in an enterprise environment. This includes support for all major platforms, run-times, languages, support for major industry standards, published interfaces and algorithms, no security by obscurity, documented trust and threat models, and support for Common Criteria and similar formal security validation programs. |
| **Implications** | Components which are not available through open systems mechanisms or fair-use license should be deprecated. |

| Name | Security by Design |
|---|---|
| **Statement** | Security should be designed-in as an integrated part of the system architecture. |
| **Rationale** | Security should not be an afterthought in IT solutions, but should be incorporated as part of those solutions. Coherent security mechanisms must span all tiers of the architecture, and be scalable – from small objects to large objects. This is helped by a consistent definition and management of configurations, a consistent set of security roles and persona across products, and a consistent security management user interface. |
| **Implications** | Ensuring continued compliance and regulatory alignment should be seen as a requirement and must be considered when completing any security solution.<br>To be both coherent and scalable, interoperable security "building blocks" need to be capable of being combined to provide the required security mechanisms. |

| Name | Sharing |
|---|---|
| **Statement** | Security solutions should include management mechanisms to accommodate sharing. |
| **Rationale** | Multiple solutions can share a single IT environment, such as in a shared service center. To achieve this goal, security services and management must be able to span multiple domains, each domain potentially providing its own and independently set security policy, identity, models, and so on. Architectures must explicitly document the assumptions and limitations made in terms of span of control. |
| **Implications** | Interoperability is key to sharing, and is enabled by using open systems standards wherever possible. |

| Name | Defense in Depth |
|---|---|
| **Statement** | Multiple levels of protection, especially if they use different mechanisms, should be used to provide effective defense in depth. |
| **Rationale** | Defense in depth can be achieved by multiple levels of enforcement and detection. Resources must be designed to protect themselves as a first layer of defense. Intrusions can be contained through isolation and zoning. Multiple levels also minimize the attack surface to the outer-most accessible layer.<br>Designs should incorporate fail-safe techniques.<br>Different (but complementary) protection mechanisms should be used whenever available. A single type of security solution or application should not be solely relied upon as a protection mechanism. If one is compromised, similar mechanisms may be compromised in the same manner.<br>In the same context, while two independent layers of protection for one resource may improve security, using two different mechanisms for the same purpose for two resources increases the chances that if one of them gets broken the other can remain unbroken. |
| **Implications** | Multiple layers of protection clearly affords increased protection, especially if each layer uses a different defense mechanism. These mechanisms, however, must be carefully deployed to avoid reduced operational efficiency in enabling authenticated users to gain timely access to resources to which they are entitled and authorized.<br>In all cases, the closer the protection is to the resource, the more effective is that protection. Integrating protection with the resource so that it is self-protecting adds even greater protection. |

| Name | Security is Model-driven |
|---|---|
| **Statement** | Models are reflective of the operating environment, common models, and consistent formats for identity and trust, data, policy, applications, security information and events, and cryptographic keys. |
| **Rationale** | Models are consistently interpreted across the stack (for example, network identities are linked to application-level identities) and across units (for example, policies and trust are negotiated and understood within a federation). Models are consistently validated against reality (feedback from policy and model discovery). |
| **Implications** | Using models is a valuable tool to check for consistency and discover any gaps or flaws in a design. |

| Name | Simplicity |
|---|---|
| **Statement** | Security mechanisms should be pervasive, simple, scalable, and easy to manage. |
| **Rationale** | Designs for security solutions should be as simple as is needed to make them effective, and be coherent across the whole architecture. Unnecessary complexity is a threat to good security; a complex security system solution increases the likelihood of unknown and unforeseen weaknesses in its security, and makes it more difficult to understand. A solution cannot be protected sufficiently if it is not sufficiently understood and easy to manage. This simplicity principle overlaps with the Security by Design principle. |
| **Implications** | Secure architectures are most effective when based on a thorough understanding of the requirements, leading to a design that provides effective adequate protection while minimizing complexity and impact on operational performance. |

| Name | Protection Against Insider & Outsider Attacks |
|---|---|
| **Statement** | Security measures should maintain their intended effectiveness irrespective of the source credentials of a principal claiming access to a resource. |
| **Rationale** | Security solutions should cater to mitigate both internal and external threats, because insider attacks account for the majority of all attacks. Security defence should be implemented in a manner that is not differentiated based on the threat source. Also, critical information assets may require additional levels of authentication and authorization regardless of where they are accessed. |
| **Implications** | Protection of resources – including digital identities and personas, as well as assets – should be applied strictly, irrespective of whether the claim for access originates from inside or outside the corporate perimeter. |

### Trust

| Name | Trust Assurance |
|---|---|
| **Statement** | Mutual trust assurance levels must be determinable. |
| **Rationale** | Devices and users must be capable of appropriate levels of (mutual) authentication for accessing systems and data.<br>Authentication and authorization frameworks must support the trust model. |
| **Implications** | Trust assurance must be interoperable to support business collaborations. |

| Name | Weakest Link |
|---|---|
| **Statement** | Overall security can only be as effective as the weakest link in the chain from end-to-end. |
| **Rationale** | Designs need to identify and eliminate or monitor any possible vulnerability within the security solution – which may be people or process – i.e., not necessarily technology-related. Security is only as strong as the weakest link, so intelligent attackers will seek the weakest point to attack.<br>Designs should minimize the number of possible entry points that attackers can find, and provide specific monitoring and control mechanisms at these points. This principle has particular application to SOA, where a potentially large number of services collaborate in different business processes and contexts. It encourages use of a mediation layer to control access to services, as opposed to each individual application managing its own authentication and authorization, therefore minimizing the number of potential weakest links.<br>All parties involved in the implementation of a secure chain must accept responsibility for the complete chain of security end-to-end. If one party in the chain does not conform or otherwise compromises a security component, they will introduce a weakness in the overall implementation.<br>See also the Separation of Duties principle. |
| **Implications** | While this principle is readily appreciated when envisioning the strength of a chain, it does require solution architects and designers to analyze the complete path of a system flow from one end to the other end, and assess the security of that path at each point where the continuity changes from one medium to another. Experience indicates that the endpoints are a particular point where security can be weak. |

## Data Protection

| Name | Security Context |
|---|---|
| **Statement** | Validate the security context for which the security solution(s) is designed. |
| **Rationale** | Security-critical resources must be aware of their security context. Assume security context at your peril, because security solutions designed for one environment may not be transferable to work effectively in another. Thus, it is important to understand the limitations of any security solution.<br>Use-Cases:<br>1. Problems, limitations, and issues can come from a variety of sources, including geographic, legal, technical, acceptability of risk, etc.<br>2. Resources and actors should be kept aware of their environment (including physical location and logical co-location), and their security status and context. |
| **Implications** | Architects and designers need to analyze the complete domain they are designing in, to validate that they are not making invalid assumptions when proposing to use a security resource that is familar in one domain into another domain – where perhaps one, but importantly at least one, critical criterion is different. |

| Name | Data Access Control |
|---|---|
| **Statement** | Access to data should be controlled by security attributes of the data itself. |
| **Rationale** | Attributes which facilitate access can be held within the data (Digital Rights Management/metadata) or they could be held on or consist of a separate system. Access/security could be complemented by encryption. Access/security controls can be enhanced through encryption. Some data may have, or require, classifications for wider audiences such as “public, non-confidential” attributes. Access and access rights may have, or require, a temporal component. |

| Name | Data Access Control |
|---|---|
| **Implications** | Design for access to data – in fact any system asset – should include consideration of what attributes the asset owner has provided to protect access to that asset. |

| Name | Data Protection |
|---|---|
| **Statement** | By default, data must be appropriately secured when stored, in transit, and in use. |
| **Rationale** | Must maintain security policy for protection of data at all stages.<br>Removing a default must be a conscious act.<br>High security should not be enforced for everything.<br>"Appropriate" in the statement of this principle means varying the level of restricted access according to the sensitivity of the data. For example, the G8 "traffic light" protocol for data sensitivity classification defines red for highly sensitive, amber for sensitive, green for normal business, and white for public. |
| **Implications** | Information should be classified at levels of restricted access which define how much security they need – from none (for public data) to highly sensitive. |

### Management

| Name | Accountability |
|---|---|
| **Statement** | Security solutions should include collection of audit information on system operations.<br>(See also the Accountability in Service-based Architectures principle.) |
| **Rationale** | In today's environments, with many requirements in the compliance area, it is important that all security-relevant actions can be logged and audited, the audit infrastructure should be scalable to handle these events, and audit information must be immutable and not easily repudiated. |
| **Implications** | Business operations must demonstrate effective security management over regulatory compliance, audit, and good business practices. Failure in compliance or audit poses severe risks to the continued existence of the business. |

| Name | Regulation/Compliance |
|---|---|
| **Statement** | Security solutions should include mechanisms to configure and monitor systems for regulatory compliance. |
| **Rationale** | Regulations drive many requirements in IT security projects, and regulations change over time. To handle this, flexible support is required for the constraints set by government regulations and industry standards and traceability between regulations, standards, and business policies and the security policies used to implement them. |
| **Implications** | Ensuring continued compliance and regulatory alignment must be considered when completing any security solution.<br>Use of automated compliance set-up and monitoring will facilitate updates to keep track of regulatory changes and monitoring to maintain continuity of compliance. |

| Name | Privacy |
|---|---|
| **Statement** | Security solutions should include mechanisms to implement policy on privacy. |

| Name | Privacy |
|---|---|
| **Rationale** | In the current age of data sharing, privacy becomes increasingly important. Solutions should highlight the use of private information and corresponding data protection control mechanisms, and enable the principles of notice, choice, and access. Administrator access must also be subject to these controls. This principle demands particular attention in Cloud and SOA requirements, where privacy must be applied to all points in the provision of each service. |
| **Implications** | The enterprise needs to implement applicable privacy policies, both regulatory and best practice, and monitor whether they conform to them. |

| Name | Compartmentalization |
|---|---|
| **Statement** | Resources should be protected at separated levels appropriate to their value, confidentiality, integrity, and accountability classification. |
| **Rationale** | Information assets should be isolated based on data classifications, with access to these assets based on similar criteria. Networks should be segmented based on user communities. Compartmentalizing assets and networks provides an additional layer of defense and control. |
| **Implications** | Mechanisms providing separation of impact of loss in the event of a security breach add significantly to the overall security level afforded in any architecture. |

| Name | Separation of Management Services |
|---|---|
| **Statement** | Security services for management, enforcement, and accountability should be delivered as separate functions through separate authorities. |
| **Rationale** | Security management services (identity, authentication, authorization, audit, etc.) should be shared by multiple applications within the security infrastructure, enabling consistent monitoring and enforcement. The enforcement itself (through cryptography, or policy enforcement, or physical isolation) is typically distributed and kept as close as possible to the resources. |
| **Implications** | Segregation of provision of security services is an important security control mechanism, in the same way as it is in provisioning services in the physcal world. |

| Name | Separation of Duties |
|---|---|
| **Statement** | Security operations should enforce separation of duties. |
| **Rationale** | Security operations should enforce separation of duties, so that, for example, software developers should not have access to running production systems. Similarly, broad administrative access should be restricted as much as possible (the Least Privilege principle), with individuals holding administrative access to only the those subsystems where they have appropriate roles.<br>Permissions, keys, privileges, etc. must ultimately fall under independent control, otherwise there will always be a weakest link at the top of the chain of trust. |
| **Implications** | Segregation of allocation of security functions is an important security control mechanism, in the same way as it is when designing operating procedures for humans. |

### Interoperability

| Name | Least Privilege |
|---|---|
| **Statement** | A principal should have only the privileges required to carry out its specified task. |
| **Rationale** | An entity granted entitlement to access an asset should only be given sufficient access rights to perform the operation for which rights were granted, and no others. |
| **Implications** | Requires clear classification of resources and permission levels for constraining access and authorization to perform operations only to the required extent. |

| Name | Agility and Extensibility |
|---|---|
| **Statement** | Security solutions should include agility and management mechanisms to accommodate extensibility. |
| **Rationale** | Business demands that security enables business agility and is cost-effective to implement. Component-based solutions support separation of the management of mechanisms themselves, to support a variety of mechanisms under the same framework. Already deployed solutions should allow for the addition and extension of new mechanisms within the existing management framework. |
| **Implications** | Globalization and scalability are key considerations in architecting systems, particularly for use in business collaborations, including multi-nationally. |

| Name | Consumability |
|---|---|
| **Statement** | Security solutions should include management mechanisms to accommodate consumability. |
| **Rationale** | All security services must be easily consumable by a variety of audiences. This includes programmers who develop and integrate applications with the security services, management systems that create, update, and manage security policies and other security artifacts, and people who manage security activities, audit security activities, and manage access to protected resources. |
| **Implications** | Consumability is best achieved by security following human behaviors. |

# Cloud and SOA Security Principles

Beyond the General Security Principles, the following Security Principles are considered specifically relevant to assuring security in architecting Cloud and SOA environments.

| Name | Weakest Link |
|---|---|
| **Statement** | Adding to the Weakest Link principle, this principle has particular application to Cloud and SOA. |
| **Rationale** | This principle has particular application to Cloud and SOA, in that it encourages use of a mediation layer to control access to services, as opposed to each individual application managing its own authentication and authorization. |
| **Implications** | This principle has specific impact on the approach to architecting for Cloud and SOA. |

| Name | Off-line Backup |
|---|---|
| **Statement** | It must be possible for Cloud tenants to make a back-up of their data on another environment of their choice. |
| **Rationale** | A back-up provides the tenant extra certainty to be able to access his data in case of an emergency such as inaccessibility of the Cloud system. |
| **Implications** | Availability is a key requirement in the CIA objectives for effective security. Accordingly, an enterprise needs to protect itself from external failures resulting in access to its data and processing resources becoming unavailable. |

| Name | Policy-based Access to Services |
|---|---|
| **Statement** | Service consumption will be controlled by policy. Policies must be held externally from applications. |
| **Rationale** | Where services are accessed in a distributed environment, the policies applying to those services should not be accessible from those services. |
| **Implications** | Separate policy from implementation. |

| Name | Data Protection |
|---|---|
| **Statement** | Data protection should allow compliance with corporate or regulatory compliance standards and practices, implemented in a manner that supports the other principles for Cloud and SOA, such as policy-based access, federation, multi-tenancy, etc. |
| **Rationale** | Specific attention should be paid to the end-to-end data flows to ensure no gaps are present. Collection and provision of data in transit must also be appropriately protected. Data protection should also include consideration for back-up and recovery, e-discovery, etc. |
| **Implications** | In distributed environments, data flows are often complex, so need to be handled in flexible ways which provide acceptable compromises with other applicable data protection principles. |

| Name | Privacy |
|---|---|
| **Statement** | Extending the Privacy principle, protection of private information must demonstrate compliance with the enterprise's requirements for such protection across all points providing each service. |
| **Rationale** | Adding to the Privacy principle, this principle demands particular attention in Cloud and SOA requirements, where privacy must be applied to all points in the provision of each service. |
| **Implications** | This principle has specific impact on the approach to architecting for SOA. |

| Name | Multi-tenancy |
|---|---|
| **Statement** | A Cloud Computing model must support tenant and solution isolation among multiple tenants of the Cloud. |
| **Rationale** | Cloud Computing stores clients' assets (information and operational processes applications) in servers distributed "who knows where", so it is critical that each client's assets are kept securely separated from the assets of other clients, irrespective of the storage media and processing resources that each client may also use in the Cloud. |
| **Implications** | Cloud providers offer assurances that they provide secure isolation between the assets of each of their clients. While this is difficult for them to evidence, their isolation control mechanisms seem to demonstrate success over this capability. |

| Name | Data Evacuation |
|---|---|
| **Statement** | A user of Cloud Computing must be able to request its data be removed in its entirety from the Cloud on terminating use of it and be assured that no data is left behind in an accessible state. |
| **Rationale** | A principle of security is that once an asset leaves your locus of control, you cannot control where it goes, and can then only exercise "management" over its security. One aspect of this management is gaining assurance from a Cloud Provider that when you have finished using their Cloud, you can take back all the assets of value that you put into it. Against a background where forensics on stored media demonstrates how meaningful data can be retrieved even though it was supposed to have been deleted, clients expect their Cloud Providers to demonstrate adequate assurances that their operations in the Cloud are fully removed. |
| **Implications** | A situation where small remnants of data tracing past storage and processing activities remain; those remnants are called "breadcrumbs".<br>Forensics can rebuild meaningful information from those breadcrumbs, which then compromises the confidentiality of the party whose breadcrumbs were left. |

| Name | Intellectual Property |
|---|---|
| **Statement** | A Cloud Computing model must support the notion that a user's intellectual assets (capital/property) and individuals' or organizations' innovations are protected contractually and where possible also technically, and respected by its Cloud hosting providers and/or their associated supply chain, including residual knowledge and experience-based knowledge. |
| **Rationale** | Intellectual Property Rights (IPR) is a high-value asset for any enterprise, so any operation in the Cloud where IPR applies over the client's data or over the client's processing operations performed on that data, must remain in the exclusive ownership of the client, with none residing at any point with the Cloud or service provider. |

| Name | Intellectual Property |
|---|---|
| **Implications** | An enterprise's business managers and their legal advisors must be consulted on every occasion where a proposal to use the Cloud involves moving IPR assets into the Cloud, and decide whether their IPR is at acceptable risk before approving doing so. |

| Name | Accountability in Service-based Architectures |
|---|---|
| **Statement** | Security design in SOA/Cloud architectures should include collection and provision of audit information on system operations.<br>(See also Accountability.) |
| **Rationale** | Service consumers must provide service providers with sufficient information to allow them to fulfil their obligations in this regard both to the consumer and to regulation. Conversely, providers are responsible for logging all information required by the consumers for business or regulatory audit purposes. |
| **Implications** | Business operations must demonstrate effective security management over regulatory compliance, audit, and good business practices. Failure in compliance or audit poses severe risks to the continued existence of the business. |

# Other Sources

Most organizations who develop their own IT architectures have their own in-house Security Principles which guide their IT architecture designers.

A web search will find many references to sources of Security Principles. We suggest the following as recommended published sources.

### ISO/IEC 27001

ISO/IEC 27001:2005: Information Technology – Security Techniques – Information Security Management Systems – Requirements. This standard lists Security Principles under the following headings:

- Security Policy
- Security Organization
- Asset Classification and Control
- Personnel Security
- Physical & Network Management
- Access Control
- Systems Development & Maintenance
- Business Continuity
- Audit & Compliance

### NIST

NIST Special Publication 800-27: Engineering Principles for Information Technology Security, by Gary Stoneburger, Clark Hayden, & Alexis Feringa, June 2004; refer to: http://csrc.nist.gov/publications/nistpubs/800-27A/SP800-27-RevA.pdf. This document describes a number of good engineering principles.

NIST Special Publication 800-14: Generally Accepted Principles & Practices for Securing Information Technology Systems, by Marianne Swanson & Barbara Guttman, September 1996; refer to: http://csrc.nist.gov/publications/nistpubs/800-14/800-14.pdf.

### Joint ISF, (ISC)2, and ISACA

Principles for Information Security Practitioners, 2010; refer to: www.isaca.org/Knowledge-Center/Standards/Pages/Security-Principles.aspx. This document sets out security principles under three main headings:

- Support the Business
- Defend the Business
- Promote Responsible Security Behavior

### OECD

Guidelines for the Security of Information Systems and Networks – Towards a Culture of Security, July 2002; refer to: www.enisa.europa.eu/act/rm/cr/laws-regulation/corporate-governance/oecd-guidelines. This document presents guidelines to any OECD entities (governments, businesses, other organizations, and individual users) who develop, own, provide, manage, service, and use information systems and networks.

### The Jericho Forum®

The Jericho Forum® Commandments (December 2006) define both the areas and the principles that must be observed when planning for a de-perimeterized future. Whilst building on "good security", the commandments specifically address those areas of security that are necessary to deliver a de-perimeterized vision. In this regard, these "commandments" build on the good Security Principles in this White Paper.

The Jericho Forum® Identity, Entitlement, and Access Management (IdEA) Commandments (May 2011) define the principles that must be observed when planning an identity eco-system. Whilst building on "good practice", these commandments specifically address those areas that will allow "identity" processes to operate on a global, de-perimeterised scale; this necessitates open and interoperable standards and a commitment to implement such standards by both identity providers and identity consumers.

# About the Authors

### Lead Author

Ian Dobson (The Open Group) – lead editor during development of this White Paper. Ian is the Forum Director for The Open Group Security Forum, and the Jericho Forum, a Forum of The Open Group.

### Base Document

Tony Carrato (IBM) – provided the base document from which these Security Principles were developed. Tony is Chief Product Architect – Smarter Cities, Industry Solutions Development, IBM Software Group.

### Lead Reviewers

Omkhar Arsasratnam (IBM) – Chair of the Cloud & SOA Security Project of The Open Group Cloud Computing Work Group, and joint project leader for this White Paper.

Stuart Boardman (Getronics) – Vice-Chair of the Cloud & SOA Security Project of The Open Group Cloud Computing Work Group, and joint project leader for this White Paper.

### Other Reviewers

Notable among the project members who contributed review feedback on these Security Principles were Stephen G. Bennet (Oracle), Peter Johnson (Eli Lilly), Michiel Perdeck (Logica), and Robert S Tucker (Shell).

# About The Open Group

The Open Group is a global consortium that enables the achievement of business objectives through IT standards. With more than 400 member organizations, The Open Group has a diverse membership that spans all sectors of the IT community – customers, systems and solutions suppliers, tool vendors, integrators, and consultants, as well as academics and researchers – to:

- Capture, understand, and address current and emerging requirements, and establish policies and share best practices
- Facilitate interoperability, develop consensus, and evolve and integrate specifications and open source technologies
- Offer a comprehensive set of services to enhance the operational efficiency of consortia
- Operate the industry's premier certification service

Further information on The Open Group is available at www.opengroup.org.