This article is a review of the security section of the "Cloud Computing Use Cases Whitepaper" Version 3.0 — posted by the Cloud Computer Computing Use Cases Discussion Group — to highlight the security issues that architects and developers should consider as they move to the cloud.

This article takes a look at the "Cloud Computing Use Cases Whitepaper," Version 3.0 from the Cloud Computing Use Case group — an information storehouse created by an open web community of more than 900 participants. It started with a group of supporters from the Open Cloud Manifesto and grew to include representatives from large and small companies, government agencies, consultants, vendors, and users.

The group agreed on three principles:

• The need for cloud users to work together.
• The need for cloud standards development to be open and customer-driven.
• The need to use existing standards whenever possible.

The scope of the "Cloud Computing Use Cases Whitepaper" is comprehensive, so we won't try to cover the entire document in one reading. For this review, we'll focus on the group's evaluation of security issues and scenarios in the cloud since security is one of the first considerations when it comes to enterprise-level cloud computing.
Cloud security topics in general

The "Cloud Computing Use Cases Whitepaper" covers the security issues that developers and architects should consider as they plan a move to the cloud. It emphasizes that, like other system environments, cloud computing as a whole can be considered an ideal example demonstrating the need for a "consistent, transparent, standards-based security framework." The individual cloud deployment model shouldn't matter.

If there were one major difference when you think about security in the cloud as opposed to other environments, what would that be? It's not primarily a technical challenge ... it is the enterprise's perceived loss of control over sensitive data and applications since the cloud service provider controls the infrastructure. Discussion of the following topics address this issue:

- **Laws and regulations**, while not technical issues, can determine which security requirements have priority over functional requirements.
- There is a minimum list of **security controls** a cloud provider should be able to deliver to make you feel that his infrastructure is secure enough for you.
- There is also a minimum list of **security federation patterns** (mechanisms) through which the infrastructure provider becomes capable of delivering security controls.

The issue of regulations and laws is not exclusively technical in nature and it's relatively simple, so let's take that up now. It is a fact of life is that many governments have strict data privacy laws that can impact the physical and logical disposition of certain data. A similar situation may exist with corporations and non-governmental organization in the form of policies or industry-specific mandates. These can also apply to applications running in the cloud. Following these laws and regulations will take precedence over all other requirements. There's no way around these laws, regulations, and policies (after all, the owners of the data and apps may decide to not let you use it) — these may be non-technical considerations that will impact on your technical choices to use the cloud at all.

Security controls

The "Cloud Computing Use Cases Whitepaper" discusses the following security controls necessary to adequately secure your cloud environment (plus the relevant standards when they exist).
Asset management. You have to be able to manage all physical/virtual hardware, network, and software assets, including being able to provide access to assets for audit and compliance purposes.

Cryptography: Key and certificate management. Kind of a no-brainer for anyone who’s developed for the web, this should include being able to use standards-based functions to assure the security of information at rest and in motion. Standard: KMIP, the OASIS Key Management Interoperability Protocol.

Data/storage security. You've got to be able to store data in an encrypted format. It’s important to recognize that some users will need to store their data physically separate from other consumers’ data. Standard: IEEE P1619, developed by the IEEE Security in Storage Working Group.

Endpoint security. Users must be able to secure their cloud resource endpoints (which means being able to restrict endpoints by network protocol and device type).

Event auditing and reporting. It might seem too obvious to mention, but one of the major keys to security is the ability to know what happened. Especially when it comes to systems failure, intrusions, and outright attacks. In this case, timeliness is crucial.

Identity, roles, access control, and attributes. This speaks strongly to the federated aspects of cloud computing. Just as cloud cannot exist without the ability of all the resources to be accessible as if they were on an interoperable, single system, security on the cloud cannot be efficient as long as the attributes of individuals and services are not defined in a "consistent, machine-readable way." Standards: SAML, the OASIS Security Assertion Markup Language and X.509 Certificates, part of the ITU Public Key and Attribute Certificate Frameworks Recommendation.

Network security. You have to be able to secure network traffic at the switch, router, and packet level; the IP stack itself should be secure.

Security policies. To make access control and resource allocation effective, you have to be able to define, resolve, and enforce security policies in a consistent, robust manner. Consistent and robust so it can be enforced automatically. Standard: XACML, the OASIS eXtensible Access Control Markup Language.

Service automation. You should have an automated way to manage and analyze security control flows and processes — like reporting events that violate security policies or customer licensing agreements — to support security compliance audits.

Workload and service management. You'll want to be able to configure, deploy, and monitor services in line with defined security policies and customer licensing agreements. Standard: SPML, the OASIS Service Provisioning Markup Language.

Security federation patterns

Federation is the underlying concept that makes cloud computing possible. It is the ability of many independent resources — assets, identities, configurations, etc. — to act like a single one. The
paper outlines the following federation patterns that help define how your provider implements the security control requirements.

**Trust.** An authentication authority provides two parties the ability to define a trust relationship. The authority should be able to exchange credentials (like X.509 certificates) and then secure messages and create signed security tokens (like SAML tokens) using those credentials. This is the basis for all security federation patterns.

**Identity management.** This is the ability to define an identity provider that can accept user credentials (ID, password, certificate, and so on) and return a signed security token to identify the user. Even without knowing the user, a token from a trusted identity provider makes it OK for the service provider to allow access.

**Access management.** This capability lets you write policies (in XACML, for example) to examine security tokens, thereby refining your ability to manage access to resources. You can use multiple factors to determine multiple levels of access to resources.

**Single sign-on and sign-off.** You are able to federate logins based on credentials from the same trusted authorities. The identity management pattern enables this one.

**Audit and compliance.** You need federated audits from compliance data spread across multiple domains to document compliance with service-level agreements and other regulations.

**Configuration management.** The ability to federate configuration data for services, applications, and virtual machines.

Existing security best practices are exactly what the phrase implies — "security best practices." Since best practices usually end up in standards, the authors suggest that a designer or developer look to existing standards first as a mechanism to deliver federation patterns.

**Security use case scenarios**

The authors of the "Cloud Computing Use Cases Whitepaper," Version 3.0 crafted common scenarios that cover a range of application types, deployment models, patterns, and roles to provide the following:

*customer cloud computing experiences + security requirements = successful cloud usage*

The use cases in the whitepaper are designed to

- Provide practical, customer-experience-based context to enable discussion on interoperability and standards.
- Delineate where to use existing standards.
- Highlight where standards need to be created.
- Demonstrate the importance of open cloud computing to business.

Each section starts with the general scenario and:
• Describes the problem scenario using the language directly from the "Cloud Computing Use Cases Whitepaper," Version 3.0.
• Discusses how the problem was solved using a cloud solution.
• Provides a list of the requirements, controls, and federation patterns employed to achieve the solution.

**Sensitive data, private infrastructure overwhelmed**

The scenario:

An insurance company has a claims application used to capture data about their policy holders and any property damage they suffer. A hurricane is projected to strike the Gulf Coast region of the US, likely causing massive property damage. This will create a huge spike in claims which will in turn create an enormous load on the corporate IT infrastructure.

The company’s decision is to use a public cloud provider to deliver virtual machines to handle the expected demand.

The company must control access between the enterprise system and the virtual machines hosted by the cloud provider, limiting access to only authorized agents of the company.

The company must securely transmit any data created by cloud-based instances of the application back inside the corporate firewall.

The cloud provider must ensure that no traces of the application or its data remain whenever a virtual machine is shut down.

**Customer problem solved:** The public cloud environment will let the company handle the workloads that may be as much as an order of magnitude greater than it is used to. Off-loading the capital costs of this one-time event is much less expensive than buying the physical capabilities to handle this load in on a long-term basis.

**Requirements and controls:**

- Requirement: Access to application limited to particular roles.
  Security controls: Identity, roles, access control, and attributes; asset management; and network security.
- Requirement: All traces of the application or data must be deleted when a VM is shut down.
  Security controls: Workload and service management.

**Federation patterns:** Trust, access management, configuration management.

**Limited resources, needs new apps**

The scenario:

An online retailer needs to develop a new Web 2.0 storefront application, but does not want to burden its IT staff and existing resources.
The company chooses a cloud provider to deliver a cloud-based development environment with hosted developer tooling and a source code repository. Another cloud provider is chosen to provide a testing environment so that the new application can interact with many different types of machines and huge workloads. Choosing two providers to handle cloud-based development and testing means federation is crucial.

**Customer problem solved:** Cloud-hosted development tools means you're not responsible for installing, configuring, synchronizing, and managing tools on your developers' machines. If you've got a large product build, the cloud infrastructure can scale up to meet the size requirements. If there's a newer version of a file in the cloud to test against, your test environment can access the latest.

As for testing, testing against the more highly interactive Web 2.0 interface (instead of a static Web page) will better determine your application's elasticity in real-world situations (its ability to stretch and accommodate higher loads and a wider variety of VM images).

**Requirements and controls:**

- Requirement: Developer tools installed and maintained in one central location.  
  Security controls: Asset management.
- Requirement: All traces of the application or data must be deleted when a VM is shut down.  
  Security controls: Workload and service management.
- Requirement: Single sign-on across development and testing clouds.  
  Security controls: Cryptography; endpoint security; identity, roles, access control, and attributes; and network security.
- Requirement: Controlled access to source code and test plans.  
  Security controls: Asset management and identity, roles, access control, and attributes.
- Requirement: Builds and tests must start and shut down VMs automatically.  
  Security controls: Service automation.
- Requirement: Builds and tests must report statistics on VM usage and performance.  
  Security controls: Event auditing and reporting.

**Federation patterns:** Trust, identity management, access management, single sign-on, audit and compliance, configuration management.

**Storing and accessing trade secrets**

The scenario:

A financial investment company is launching new investment products to its agents and affiliates. A number of videos have been created to teach the company's agents and affiliates about the benefits and features of the new products. The videos are very large and need to be available on demand, so storing them in the cloud lessens the demands on the corporate infrastructure. However, access to those videos needs to be tightly controlled. For competitive reasons, only certified company agents should be able to view the videos. An even
stronger constraint is that regulations require the company to keep product details, including the videos, confidential during the quiet period before the launch of the product.
The company's decision is to use a public cloud storage provider to scale the secure hosting and streaming of the videos.
The cloud solution must control the videos with an auditable access control mechanism that enforces the company's security policies.

**Customer problem solved:** With public cloud storage, this company doesn't have to increase its own data center resources to manage tremendous amounts of data. The level of government regulation involved in this case (above and beyond the business concerns) means that the cloud provider must be able to guarantee compliance or it cannot be considered.

**Requirements and controls:**

- Requirement: Access to the videos limited to particular roles.
  Security controls: Identity, roles, access control, and attributes; asset management; network security; and policies.
- Requirement: Data stored in the cloud must be secured.
  Security controls: Cryptography and data/storage security.
- Requirement: Data stored in the cloud must be transferred back inside the company's firewall.
  Security controls: Cryptography, data/storage security, endpoint security, and network security.

**Federation patterns:** Trust, identity management, access management, and audit and compliance.

**Cross-referencing security controls, federation patterns, and the scenarios**

The following two tables summarize the relationships among security controls, federation patterns, and the scenarios. The first summarizes the relationship between the security controls and the customer scenarios and the second does the same between the federation patterns and the scenarios.

**Table 1. Security controls and the scenarios**

<table>
<thead>
<tr>
<th>Security control</th>
<th>Peak load/ Insurance</th>
<th>Dev &amp; test/ Retail</th>
<th>Secure storage/ Finance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asset management</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>Cryptography</td>
<td></td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>Data/storage security</td>
<td></td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>Endpoint security</td>
<td></td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>Event auditing and reporting</td>
<td></td>
<td></td>
<td>+</td>
</tr>
<tr>
<td>Identity, roles, access controls, attributes</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
</tbody>
</table>
Network security + +
Policies +
Service automation +
Workload and service management + +

Table 2. Security federation patterns and the scenarios

<table>
<thead>
<tr>
<th>Federation pattern</th>
<th>Peak load/ Insurance</th>
<th>Dev &amp; test/ Retail</th>
<th>Secure storage/ Finance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trust</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Identity management</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Access management</td>
<td>+</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>Single sign-on</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>Audit and compliance</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>Configuration management</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
</tbody>
</table>

In conclusion

The authors of the "Cloud Computing Use Cases Whitepaper," Version 3.0 note that "security consistently raises the most questions as consumers look to move their data and applications to the cloud."

The conclusions the "Cloud Computing Use Cases Whitepaper," Version 3.0 reaches about security in the cloud are clear:

- Cloud computing doesn't introduce new security issues that administrators, planners, and developers don't already face when considering general IT security.
- The difference in how you have to approach implementing and enforcing general IT security versus cloud security is that there is always a third-party involved in public cloud use. (On-premise cloud is another story.)
- Meaningful transparency and disclosure from cloud providers is a necessity.
- If there's an existing standard to fulfill a security requirement, cloud users must insist providers use it; if there's not, insist the community develop one.

This summary and review offers a baseline and an overview of scenarios that illustrate cloud security regulations and controls. We encourage you to study the original "Cloud Computing Use Cases Whitepaper" in its entirety for Cloud Computing Use Case Discussion group's analysis of what developers and planners should demand from their cloud providers to deliver a secure environment for precious data and applications.

© Copyright IBM Corporation 2010
Trademark
(www.ibm.com/developerworks/ibm/trademarks/)