Editors: Lori M. Kaufman, lori.kaufman@ieee.org
Bruce Potter, bpotter@pontetec.com

Monitoring Cloud Computing by Layer, Part 1

JONATHAN SPRING
Software Engineering Institute

The general characteristics of cloud computing's three service models—software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS)—include on-demand self service, broad network access, pooling of resources, rapid elasticity of provisioning resources, and service or resource monitoring.[1] On the basis of the Cloud Security Alliance's work, a cloud is modeled in seven layers: facility, network, hardware, OS, middleware, application, and the user.[2] These layers can be controlled by either the cloud provider or the cloud customer.

Table 1 identifies which layers the cloud provider controls under the three service models. Any service-level agreement (SLA) should clearly delineate these lines of responsibility. Analyzing these controls per cloud layer can make it easier to manage auditing and security monitoring. The desired controls don't change significantly for a given layer according to whether the cloud provider or customer manages that layer, but implementation and scalability differences exist. Regardless of who's responsible for the layer, the customer must ensure that the controls are enacted according to its standards.

Most clouds receiving publicity are public clouds, which are resources owned and managed by a vendor that sells or leases its services to a large industry group or the public. Public clouds carry the highest risk of data exposure and compromise owing to the less-controlled environment and must be handled with the appropriate caution. Any cloud project will have idiosyncrasies, and each requires its own risk assessment.

Here, I present a set of recommended restrictions and audits to facilitate cloud security. Although the recommendations might be overkill for deployments involving no sensitive data, they might be insufficient to allow certain information to be hosted in any public or community cloud. Owing to space constraints, I cover only the lower four layers here. Part 2 will complete the discussion by covering the middleware, application, and user layers.

The Facility Layer

Monitoring at this layer is generally in the purview of physical security. A robust physical-security policy will have many facets for surveillance, personnel, continuity of operations, and architectural resilience. Controlling and monitoring physical access to the hardware is a high priority, and surveillance should at least include closed-circuit cameras and patrolling security guards. The cloud provider must formulate personnel policies with full appreciation of the observed increase in malicious insiders' involvement in security breaches[3] and the potentially huge impact a malicious insider could have by exfiltrating or manipulating data. The provider should follow best practices in separation of privileges, least privilege, access control systems, alarm systems, administrator logging, two-factor authentication, codes of conduct, confidentiality agreements, background checks, and visitor access.

Operating a data center is a complex process that must take into account many environmental concerns that, although not inherently security concerns, could affect the information's availability. The data center should have a comprehensive continuity-of-operations plan (COOP), preferably conforming to US Federal Emergency Management Agency standards.[4] It should also, where necessary, have a liaison with customers as to how the provider integrates with the customers' COOP. Architectural security should be sufficient to protect the data center from physical attack, given the value of the data inside. The provider should regularly review and update these policies as conditions change.

The Network Layer

An essential characteristic of cloud computing is that the provider provides and controls the network access between the customer data and the users across the Internet. This border between relatively trusted space (the provider's facilities) and

 1540-7993/11/$26.00 © 2011 IEEE

the inherently untrustworthy network outside is, like all such borders, important for information security and monitoring. Firewalls, dynamic firewalls, intrusion detection systems (IDSs), intrusion prevention systems (IPSs), and network proxies are the basic network devices for protecting the network border. A firewall at the cloud border that blocks obviously troublesome packets can limit, but not eliminate, access by known malicious entities at this interface. An active IP and domain name blacklist policy is one possible security advantage of cloud models, because the cloud provider has the resources to blacklist quickly and the infrastructure to adopt changes quickly.

Operating a higher-layer device (IDS, IPS, or proxy) at the border that's responsible for all the traffic in and out of the data center might not be feasible. The traffic volume could be extremely high, preventing the available solutions from scaling to allow the required performance. Unless the data center only offers one or a very few types of services, the rules for the device would be too vague to provide much value. This is particularly true for PaaS and IaaS because these higher-layer devices leverage information from the application layer, which in those service models is controlled by the customer and therefore is unpredictable. IDSs, IPSs, and proxies could still play a role in a cloud environment in a federated manner, if the provider has logically segregated different segments for different tasks.

The network defense devices are the natural place to collect information about security events on the network. The provider should log this information; these logs are a natural place to go to audit the network's security posture. The provider should also collect further network information from some type of packet capture. Full packet capture, ideally including protocol metadata extraction and indexing, is useful; however, the retention period for such data volumes is quite short. This limits the data to a role in quick forensic analysis of recent attacks and retrospective exploration of the scope of incidents (for a somewhat longer retention period) based on application-layer indicators. For long-term auditing and more manageable data volumes, the provider should maintain, monitor, and audit network flow data.[5] The customer should request summaries of all these audits as part of its SLA verification.

Table 1. What the cloud provider controls.*

	Service model		
Layer	Software as a service	Platform as a service	Infrastructure as a service
Facility	✓	✓	✓
Network	✓	✓	✓
Hardware	✓	✓	✓
OS	✓	✓	?
Middleware	✓	?	—
Application	✓	—	—
User	—	—	—

* Question marks indicate layers in which either the provider or user could be in control.

The Hardware Layer

Because the customer interacts only with a virtualized environment, the provider is responsible for maintaining and monitoring the hardware. The provider can use software to monitor the connection topology, memory use, bus speeds, processor loads, disk storage, temperature, voltage, and so on. The provider must measure such quantities to effectively load-balance its resources. This monitoring's security implications aren't as clear. Distinguishing benign and malicious anomalies in such an environment is difficult, especially without access to customer software commands.

The provider must guarantee that the hardware is tamper-free. Appropriate physical security protocols should prevent tampering, or at least identify suspicious behavior as it occurs and remedy it immediately. However, the provider should audit hardware configurations to verify that nothing has tampered with them. Otherwise, the provider is concerned primarily with availability and should document and report as with the facility layer. Hardware security discussions with National Security Agency-level paranoia are beyond this article.

The OS

Any OS included in a cloud service must be highly secured and controlled. In general, fewer features in an OS translate to fewer failure points. Given that an OS in a cloud environment has few essential functions, nonessential functionality should be removed. The remaining functionality should be checked thoroughly for vulnerabilities before the OS is deployed. Securing the host OS is vitally important because if it's compromised, any customer data on it is compromised.

The host OS monitors and arranges all system calls between the virtual machines (VMs) and the hardware, so it can access any data passing to or from the VM. The host OS also can access all data

"stored" on the VM because that data is stored on the disks that the OS controls. Storage of sensitive data can be mitigated with encryption in some cases, because an OS in a service that merely stores the data wouldn't have the key. However, this isn't true about a service that actually will process the encrypted data in the cloud. Because the OS manages all the VM's functions, it could access any plaintext in the hosted VM, including plaintext keys the processor uses.

The cloud provider should deploy a single, hardened, pared-down OS throughout its cloud. It can then monitor these systems' images for any binary changes. If it found any, it could log the offending system and return it to a known good state from read-only media. The provider could also procure memory dumps from machines and check them against a white list for the processes known to be included in the OS. This process has been demonstrated,[6] and some providers are using it. Again, if anything has been compromised, the provider could capture the system and memory image and return the running system to a known good state. The provider should further investigate all such events to try to identify the initial cause. This will help identify where the security policy or configuration might be lacking or need updating to adjust to the threat and prevent future infections.

In an IaaS model in which the customer provides the main OS, the customer probably doesn't have the benefit of a highly homogenized, pared-down OS. In this case, the customer should implement all the standard, non-cloud-based best practices in monitoring. However, because the customer can't access the hardware, there won't ever be a guarantee that no malicious process resides between the OS and the hardware modifying or copying data. The provider's hypervisor software, which provides the characteristic cloud elasticity, is surely a candidate for this function. In theory, the provider could compare reports on software and hardware performance and, if they didn't match, investigate further. However, even if the customer could negotiate with the provider to share these reports, correlating them might be infeasible owing to the distributed nature of cloud computing. Any risk assessment should seriously consider this inability to technically ensure against espionage.

Furthermore, in environments in which the customer might execute arbitrary code, the provider must account for the eventuality of malicious customers and intentional attempts to compromise the OS through the customer's VM. Most VM software has proven vulnerable to hostile virtualized hosts to the extent of permitting compromise of the host OS.[7] Any provider monitoring plan should account for this eventuality because, even if the cloud is in a private environment, a malicious insider or disgruntled employee in a customer organization might attempt such attacks.

In cloud computing's lower layers, it's clear how and why SLAs and contracts are the primary instruments of customer control. This might be partly because facilities, networks, and hardware are physical items that people are accustomed to leasing. As we saw in the discussion of OSs in the cloud, as the layers grow more complicated, there are more potential points of failure. The next installment will continue this investigation, discussing controls for the middleware, application, and user layers. □

Acknowledgments

The information and views in this article don't necessarily reflect the view or opinion of the Carnegie Mellon University Software Engineering Institute.

References

1. P. Mell and T. Grance, "The NIST Definition of Cloud Computing," ver. 15, Information Technology Laboratory, US Nat'l Inst. of Standards and Technology, 7 Oct. 2009; http://csrc.nist.gov/groups/SNS/cloud-computing.
2. *Security Guidance for Critical Areas of Focus in Cloud Computing v2.1*, Cloud Security Alliance, Dec. 2009; www.cloudsecurityalliance.org/csaguide.pdf.
3. *2010 Data Breach Investigations Report*, Verizon RISK Team; www.verizonbusiness.com/resources/reports/rp_2010-data-breach-report_en_xg.pdf.
4. "Directives, Guidance, and Planning," US Federal Emergency Management Agency, 2011; www.fema.gov/about/org/ncp/coop/planning.shtm.
5. J. Quittek, *Requirements for IP Flow Information Export (IPFIX)*, IETF RFC 3917, Oct. 2004; http://tools.ietf.org/html/rfc3917.
6. A. Walters and N.L. Petroni Jr., "Volatools: Integrating Volatile Memory Forensics into the Digital Investigation Process," presented at Black Hat DC 2007, 2007; www.blackhat.com/presentations/bh-dc-07/Walters/Paper/bh-dc-07-Walters-WP.pdf.
7. T. Ormandy, "An Empirical Study into the Security Exposed to Hosts of Hostile Virtualized Environments," presented at CanSecWest 2007; http://taviso.decsystem.org/virtsec.pdf.

Jonathan Spring *is a member of the technical staff in the Software Engineering Institute's CERT program. Contact him at jspring@sei.cmu.edu.*