Overview of AWS Security - Compute Services

June 2016

(Please consult **http://aws.amazon.com/security/** for the latest version of this paper)

Notices

This document is provided for informational purposes only. It represents AWS' current product offerings and practices as of the date of issue of this document, which are subject to change without notice. Customers are responsible for making their own independent assessment of the information in this document and any use of AWS' products or services, each of which is provided "as is" without warranty of any kind, whether express or implied. This document does not create any warranties, representations, contractual commitments, conditions or assurances from AWS, its affiliates, suppliers or licensors. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

AWS Service-Specific Security

Not only is security built into every layer of the AWS infrastructure, but also into each of the services available on that infrastructure. AWS services are architected to work efficiently and securely with all AWS networks and platforms. Each service provides extensive security features to enable you to protect sensitive data and applications.

Compute Services

Amazon Web Services provides a variety of cloud-based computing services that include a wide selection of compute instances that can scale up and down automatically to meet the needs of your application or enterprise.

Amazon Elastic Compute Cloud (Amazon EC2) Security

Amazon Elastic Compute Cloud (EC2) is a key component in Amazon's Infrastructure as a Service (IaaS), providing resizable computing capacity using server instances in AWS' data centers. Amazon EC2 is designed to make web-scale computing easier by enabling you to obtain and configure capacity with minimal friction. You create and launch instances, which are collections of platform hardware and software.

Multiple Levels of Security

Security within Amazon EC2 is provided on multiple levels: the operating system (OS) of the host platform, the virtual instance OS or guest OS, a firewall, and signed API calls. Each of these items builds on the capabilities of the others. The goal is to prevent data contained within Amazon EC2 from being intercepted by unauthorized systems or users and to provide Amazon EC2 instances themselves that are as secure as possible without sacrificing the flexibility in configuration that customers demand.

The Hypervisor

Amazon EC2 currently utilizes a highly customized version of the Xen hypervisor, taking advantage of paravirtualization (in the case of Linux guests). Because paravirtualized guests rely on the hypervisor to provide support for operations that normally require privileged access, the guest OS has no elevated access to the CPU. The CPU provides four separate privilege modes: 0-3, called rings. Ring 0 is the most privileged and 3 the least. The host OS executes in Ring 0. However, rather than executing in Ring 0 as most operating systems do, the guest OS runs in a lesser-privileged Ring 1 and applications in the least privileged Ring 3. This explicit

virtualization of the physical resources leads to a clear separation between guest and hypervisor, resulting in additional security separation between the two.

Instance Isolation

Different instances running on the same physical machine are isolated from each other via the Xen hypervisor. AWS is active in the Xen community, which provides awareness of the latest developments. In addition, the AWS firewall resides within the hypervisor layer, between the physical network interface and the instance's virtual interface. All packets must pass through this layer, thus an instance's neighbors have no more access to that instance than any other host on the Internet and can be treated as if they are on separate physical hosts. The physical RAM is separated using similar mechanisms.

Customer instances have no access to raw disk devices, but instead are presented with virtualized disks. In addition, memory allocated to guests is scrubbed (set to zero) by the hypervisor when it is unallocated to a guest. The memory is not returned to the pool of free memory available for new allocations until the memory scrubbing is complete.

AWS recommends customers further protect their data using appropriate means. One common solution is to run an encrypted file system on top of the virtualized disk device:

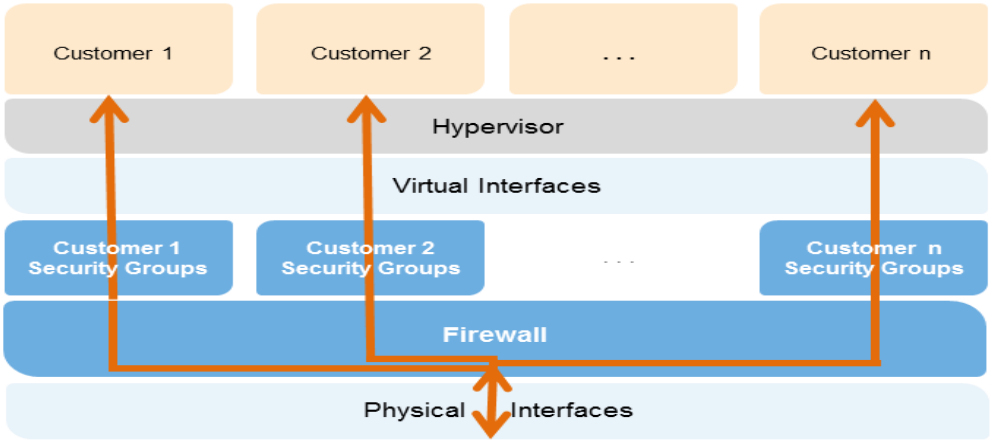

Figure 3: Amazon EC2 Multiple Layers of Security

Host Operating System: Administrators with a business need to access the management plane are required to use multi- factor authentication to gain access to purpose-built administration hosts. These administrative hosts are systems that are specifically designed, built, configured, and hardened to protect the management plane of the cloud. All such access is logged and audited. When an employee no longer has a business need to access the management plane, the privileges and access to these hosts and relevant systems can be revoked.

Guest Operating System: Virtual instances are completely controlled by you, the customer. You have full root access or administrative control over accounts, services, and applications. AWS does not have any access rights to your instances or the guest OS. AWS recommends a base set of security best practices to include disabling password-only access to your guests, and utilizing some form of multi-factor authentication to gain access to your instances (or at a minimum certificate-based SSH Version 2 access). Additionally, you should employ a privilege escalation mechanism with logging on a per-user basis. For example, if the guest OS is Linux, after hardening your instance you should utilize certificate- based SSHv2 to access the virtual instance, disable remote root login, use command-line logging, and use 'sudo' for privilege escalation. You should generate your own key pairs in order to guarantee that they are unique, and not shared with other customers or with AWS.

AWS also supports the use of the Secure Shell (SSH) network protocol to enable you to log in securely to your UNIX/Linux EC2 instances. Authentication for SSH used with AWS is via a public/private key pair to reduce the risk of unauthorized access to your instance. You can also connect remotely to your Windows instances using Remote Desktop Protocol (RDP) by utilizing an RDP certificate generated for your instance.

You also control the updating and patching of your guest OS, including security updates. AWS-provided Windows and Linux-based AMIs are updated regularly with the latest patches, so if you do not need to preserve data or customizations on your running Amazon AMI instances, you can simply relaunch new instances with the latest updated AMI. In addition, updates are provided for the Amazon Linux AMI via the Amazon Linux yum repositories.

Firewall: Amazon EC2 provides a complete firewall solution; this mandatory inbound firewall is configured in a default deny-all mode and Amazon EC2 customers must explicitly open the ports needed to allow inbound traffic. The traffic may be restricted by protocol, by service port, as well as by source IP address (individual IP or Classless Inter-Domain Routing (CIDR) block).

The firewall can be configured in groups permitting different classes of instances to have different rules. Consider, for example, the case of a traditional three-tiered web application. The group for the web servers would have port 80 (HTTP) and/or port 443 (HTTPS) open to the Internet. The group for the application servers would have port 8000 (application specific) accessible only to the web server group. The group for the database servers would have port 3306 (MySQL) open only to the application server group. All three groups would permit administrative access on port 22 (SSH), but only from the customer's corporate network. Highly secure applications can be deployed using this expressive mechanism. See diagram below:

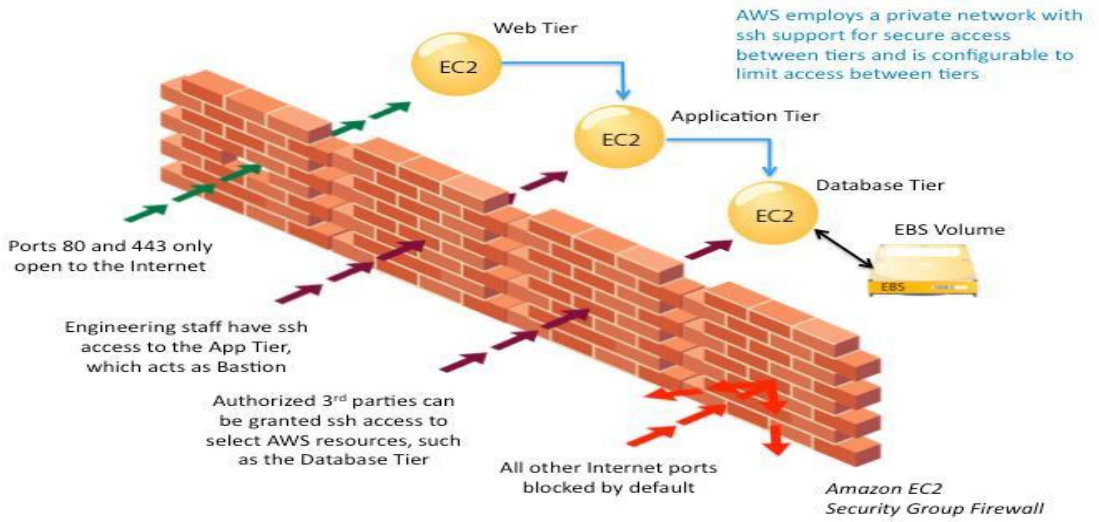

Figure 4: Amazon EC2 Security Group Firewall

The firewall isn't controlled through the guest OS; rather it requires your X.509 certificate and key to authorize changes, thus adding an extra layer of security. AWS supports the ability to grant granular access to different administrative functions on the instances and the firewall, therefore enabling you to implement additional security through separation of duties. The level of security afforded by the firewall is a function of which ports you open, and for what duration and purpose. The default state is to deny all incoming traffic, and you should plan carefully what you will open when building and securing your applications. Well-informed traffic management and security design are still required on a per- instance basis. AWS further encourages you to apply additional per-instance filters with host-based firewalls such as IPtables or the Windows Firewall and VPNs. This can restrict both inbound and outbound traffic.

API Access: API calls to launch and terminate instances, change firewall parameters, and perform other functions are all signed by your Amazon Secret Access Key, which could be either the AWS Accounts Secret Access Key or the Secret Access key of a user created with AWS IAM. Without access to your Secret Access Key, Amazon EC2 API calls cannot be made on your behalf. In addition, API calls can be encrypted with SSL to maintain confidentiality. AWS recommends always using SSL-protected API endpoints.

Permissions: AWS IAM also enables you to further control what APIs a user has permissions to call.

Elastic Block Storage (Amazon EBS) Security: Amazon Elastic Block Storage (EBS) allows you to create storage volumes from 1 GB to 16 TB that can be mounted as devices by

Amazon EC2 instances. Storage volumes behave like raw, unformatted block devices, with user supplied device names and a block device interface. You can create a file system on top of Amazon EBS volumes, or use them in any other way you would use a block device (like a hard drive). Amazon EBS volume access is restricted to the AWS Account that created the volume, and to the users under the AWS Account created with AWS IAM if the user has been granted access to the EBS operations, thus denying all other AWS Accounts and users the permission to view or access the volume.

Data stored in Amazon EBS volumes is redundantly stored in multiple physical locations as part of normal operation of those services and at no additional charge. However, Amazon EBS replication is stored within the same availability zone, not across multiple zones; therefore, it is highly recommended that you conduct regular snapshots to Amazon S3 for long-term data durability. For customers who have architected complex transactional databases using EBS, it is recommended that backups to Amazon S3 be performed through the database management system so that distributed transactions and logs can be checkpointed. AWS does not perform backups of data that are maintained on virtual disks attached to running instances on Amazon EC2.

You can make Amazon EBS volume snapshots publicly available to other AWS Accounts to use as the basis for creating your own volumes. Sharing Amazon EBS volume snapshots does not provide other AWS Accounts with the permission to alter or delete the original snapshot, as that right is explicitly reserved for the AWS Account that created the volume. An EBS snapshot is a block-level view of an entire EBS volume. Note that data that is not visible through the file system on the volume, such as files that have been deleted, may be present in the EBS snapshot. If you want to create shared snapshots, you should do so carefully. If a volume has held sensitive data or has had files deleted from it, a new EBS volume should be created. The data to be contained in the shared snapshot should be copied to the new volume, and the snapshot created from the new volume.

Amazon EBS volumes are presented to you as raw unformatted block devices that have been wiped prior to being made available for use. Wiping occurs immediately before reuse so that you can be assured that the wipe process completed. If you have procedures requiring that all data be wiped via a specific method, such as those detailed in DoD 5220.22-M (“National Industrial Security Program Operating Manual “) or NIST 800-88 (“Guidelines for Media Sanitization”), you have the ability to do so on Amazon EBS.

Encryption of sensitive data is generally a good security practice, and AWS provides the ability to encrypt EBS volumes and their snapshots with AES-256. The encryption occurs on the servers that host the EC2 instances, providing encryption of data as it moves between EC2 instances and EBS storage. In order to be able to do this efficiently and with low latency, the EBS encryption feature is only available on EC2's more powerful instance types (e.g., M3, C3, R3, G2).

Auto Scaling Security

Auto Scaling allows you to automatically scale your Amazon EC2 capacity up or down according to conditions you define, so that the number of Amazon EC2 instances you are using scales up

seamlessly during demand spikes to maintain performance, and scales down automatically during demand lulls to minimize costs.

Like all AWS services, Auto Scaling requires that every request made to its control API be authenticated so only authenticated users can access and manage Auto Scaling. Requests are signed with an HMAC-SHA1 signature calculated from the request and the user's private key. However, getting credentials out to new EC2 instances launched with Auto- Scaling can be challenging for large or elastically scaling fleets. To simplify this process, you can use roles within IAM, so that any new instances launched with a role will be given credentials automatically. When you launch an EC2 instance with an IAM role, temporary AWS security credentials with permissions specified by the role will be securely provisioned to the instance and will be made available to your application via the Amazon EC2 Instance Metadata Service. The Metadata Service will make new temporary security credentials available prior to the expiration of the current active credentials, so that valid credentials are always available on the instance. In addition, the temporary security credentials are automatically rotated multiple times per day, providing enhanced security. You can further control access to Auto Scaling by creating users under your AWS Account using AWS IAM, and controlling what Auto Scaling APIs these users have permission to call.

Further Reading

https://aws.amazon.com/security/security-resources/

Introduction to AWS Security Processes
Overview of AWS Security - Storage Services
Overview of AWS Security - Database Services
Overview of AWS Security - Compute Services
Overview of AWS Security - Application Services
Overview of AWS Security - Analytics, Mobile and Application Services
Overview of AWS Security – Network Services