

The Ten Riskiest AWS Misconfigurations



Introduction

Properly, securely-configured cloud accounts are critical to keeping pace with dynamic infrastructure requirements for a cloud native deployment. The number of possible configuration options in AWS can be overwhelming at times, with [dozens of key security configurations possible for EC2 alone](#). The challenge of ensuring that services are properly configured is compounded by the sheer number of AWS services available, each with its own requirements – so much so that CSPM (Cloud Security Posture Management) products have emerged to help teams keep pace.

But not all AWS configurations are created equal in terms of security impact, not all AWS services are used as commonly as others, and some configurations interact across services to provide configurations with capabilities to override or impact others. It can be confusing to figure out how to prioritize efforts or know where to begin.

The following are the most common AWS services in the market that are representative of the basic services of a fairly large cloud native deployment.

The misconfigurations were chosen based on potential security impact as determined by:

- The **popularity** of the specific service
- The **size** of the attack surface covered
- Any **'overriding' effect** on other security controls
- **Potential impact** on other AWS services or security controls (e.g., if logging is not occurring, incident response is impossible)

Knowing the riskiest configurations can guide teams to understanding how to pinpoint the potential risks of their own environments and choice of AWS services.

The 10 Riskiest AWS Misconfigurations

Amazon EC2

The ability to access cloud-based servers, [EC2](#) is Amazon's classic offering, underlying many of AWS's other services. As such, many of the configuration policies for EC2 apply to other AWS services. Therefore, it is critical to check if there are any configuration nuances when running another service (e.g., EKS) on top of EC2.

1 Access Control: Open SSH

While it's generally not a good idea – minus a very specific business need – to open any ports to the public, some connections are meant to be more secure than others and carry sensitive information. SSH is used for management at a command level, so leaving SSH open would amount to a very serious misconfiguration, where an attacker could effectively take over the command line interface.

The SSH port must not be left open. Closing it eliminates a very risky attack vector.

[Configuration Details ›](#)

Amazon IAM

[Amazon IAM](#) is not a service that is charged for separately; rather, it is a feature included in all AWS accounts, governing Identity and Access Management for those services. There is some interplay in IAM policies across different kinds of services, so it is crucial to keep an eye to these policies when adding AWS services.

2 Access Control: Using the Root Account instead of IAM users

In AWS accounts, root users have access to all AWS services and resources across the entire account. [Best practice](#) is to only use the root user in order to set up the first IAM user, and then to use IAM roles for all other normal tasks. [Only a few use-](#)

cases, for example viewing tax documents, require root access. If this configuration option is abused, or improperly set up, the consequences of exposing root user credentials are severe. It is even recommended not to create access keys for the root account because the availability of keys increases the chance of compromise. Keeping root account credentials safe and using IAM roles for most tasks ensures that the wrong person cannot access an account's AWS services and resources.

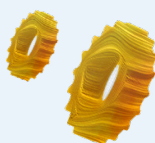
[Configuration Details ›](#)

Amazon S3

One of several storage options Amazon provides, [S3 buckets](#) have become popular with easy, out-of-the-box configurations to make data publicly accessible. This is convenient when S3 buckets are used alongside web and application servers run in EC2. However, this means there are more configurations that need to be managed to prevent the data in the S3 buckets from being inadvertently accessible by the public.

3 Access Control: S3 Bucket Public Access Block

S3 buckets can be configured to allow anyone, regardless of whether they are an AWS user or not, to write objects to a bucket or delete objects. [The Public Access Block](#) configuration will essentially override any public permissions granted via ACL or policy, serving as a central control that will block any public access to an S3 bucket. This configuration, when enabled, will not change depending on how an object has been added or a bucket is created. If there is one critical Access Control to add to an S3 bucket, the Public Access Block is the one, because of its broad protection and ability to override other potential loopholes to accessing the S3 bucket.

[Configuration Details ›](#)

RDS

One of Amazon's more popular database options, [RDS](#) is commonly used because of the out-of-the-box, automated options for configuration, management, maintenance and security.

4 Access Control: RDS Public access

Sometimes an administrator wants to make RDS available from a public endpoint for trouble-shooting purposes. But public access to RDS, when left open, can cause unnecessary exposure and risk. It is best to disable the public access option by default, only enable it for specific purposes (if absolutely needed) and, in general, set up RDS access from within a Virtual Private Cloud (VPC).

[Configuration Details ›](#)

AWS CloudTrail

Logging for activities and events in AWS services can be achieved via [CloudTrail](#), not to be confused with [CloudWatch](#), that details health and performance metrics for AWS services

5 Policy and Config/Logging: Enable a trail

By default, CloudTrail is enabled for an AWS account – but also by default it will delete data after 90 days on a rolling basis and it will not automatically capture all types of events. [Critical additional security events to monitor](#) include sign-in failures, IAM policy changes and root account usage. Failing to create trails that log specific resources, regions, and – critically – the events that should be monitored, could result in the loss of critical security intel. In the case of a security incident, there will be no way to understand what went wrong or view the events that matter to the overall cloud security posture. This requirement applies to ECS, EKS and also Fargate (see below).

[Configuration Details ›](#)

Including configurations specific to cloud native offerings:

The specific configurations for AWS's most commonly used cloud native services demonstrate how much interplay there can be between all of these services, and how important it is to configure the most important ones correctly.

AWS Fargate

This serverless offering allows customers to let Amazon handle the provisioning, managing and configuring of containers, with no need to manually launch or manage EC2 instances.

6 Access Control: Assigning a Task Execution Role

Task Execution roles are used by [Fargate](#) to pull images from private registries (which is recommended over public registries) or to publish container logs to CloudWatch; these two tasks are absolutely critical. The goal of task executions roles is to isolate permissions for each task based on an IAM role, such that each task is also prevented from seeing all the other AWS services in the account. The task execution role would be the same as the EC2 role, but the trust relationship needs to be changed such that the CNI (container networking interface) is allowed to assume the IAM role.

[Configuration Details ›](#)

7 Policy and Config: Read-only container

To support the concept of immutability, in the task definition the configuration must define a 'read only container.' This is accomplished by configuring the 'read only root system' property to 'true.'

[Configuration Details ›](#)

8 Policy and Config: awsvpc network mode for the Task Definition

To configure a VPC, which is recommended, inside the task definition the network node must specify 'awsvpc'.

[Configuration Details ›](#)

9 Logging: VPC flow logs

[Logging flows inside a VPC](#) allows traffic to be monitored to and from tasks in a VPC. This is a critical security enabler, in addition to the other Cloudtrail logging options available.

[Configuration Details ›](#)

Amazon EKS

Running Kubernetes on AWS is possible with [Amazon EKS](#). It integrates with other Amazon services like IAM and VPC, which is helpful to keeping configurations consistent.

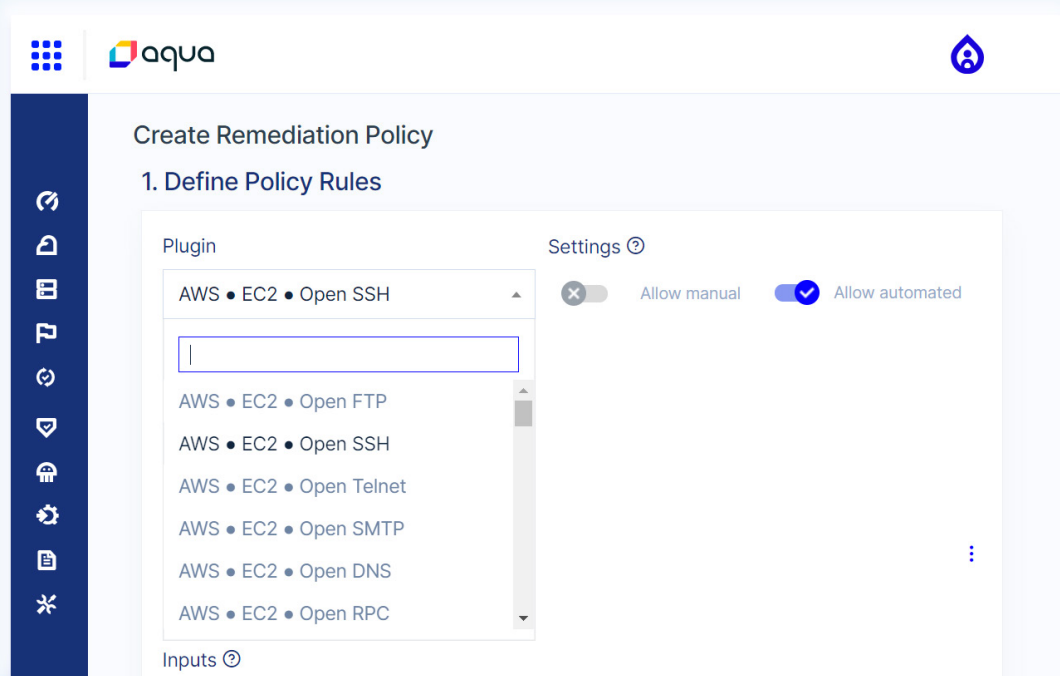
10 Access Control: Block pod access to IMDS

It is possible that user roles might be overridden by the credentials assigned to the node IAM role by EC2's instance metadata service (IMDS), through inheritance of the default setting. [Blocking pod access to IMDS](#) is recommended to minimize this risk as well as overall container permissions, in keeping with the concept of least privilege.

The Importance of Auto-Remediation

If the cloud infrastructure is dynamic and possibly spread across multiple services and environments, monitoring and remediation capabilities for cloud account configurations must also have some level of dynamism and self-healing. Auto-remediation is an ideal solution to cloud account misconfigurations that pose the highest risk.

Below is an auto-remediation example from Aqua CSPM for remediating Open SSH in EC2.



Setting up the auto-remediation policy

When that policy is created, if an Open SSH port is detected in EC2 services, a notification will appear that this issue has been fixed, and the report for that notification shows the details of the auto-remediation.

aquawave									
Cloud	Account	Time Remediated	Category	Plugin	Severity	Trigger	Fixed	Failed	View Report
aws	aqua-cspm-sandbox-remediations	2020-09-02 18:35:08	IAM	Minimum Password Length	Medium	Manual	0	1	View Report
aws	aqua-cspm-sandbox-remediations	2020-09-02 18:33:46	IAM	Minimum Password Length	Medium	Manual	0	1	View Report
aws	aqua-cspm-sandbox-remediations	2020-08-26 15:56:29	-	-		Manual	0	0	View Report
aws	aqua-cspm-sandbox-remediations	2020-08-25 17:05:56	EC2	Open SSH	Critical	Automated	1	0	View Report
aws	aqua-cspm-sandbox-remediations	2020-08-25 17:01:47	IAM	Minimum Password Length	Medium	Manual	0	1	View Report




Notification of fixed Open SSH configuration

The detailed report shows that it took only 2 seconds from the time the notification about the Open SSH port was received, at 17:05:54, to the time the auto-remediation was performed, at 17:05:56.

Manually, the process to alter the configuration would have required 12 discrete steps.

Remediation timeline for Open SSH

Remediation Timeline

- 
Real-Time Event Received
 Triggered by API call: ec2:AuthorizeSecurityGroupIngress
 2020-08-25 17:05:54
- 
Event-Based Scan Run
 Scan was auto-triggered by event | [View Scan](#)
 2020-08-25 17:05:56
- 
Remediation Executed
 Automated remediation was triggered
 2020-08-25 17:05:56



Conclusion

The cloud account configurations listed above have the most potential to either help or harm from a security perspective, and the most ideal way to achieve the recommended configurations is by leaving nothing to chance and setting up auto-remediation.

Every cloud implementation is different, and it's likely that you have many more misconfigured services and settings. The best way to see whether any of these configurations are issues in your cloud service accounts is to do a full scan, paying particular attention to any of the configurations listed above.

Get Cloud Security Posture Management
Try Aqua CSPM

Sign Up Free Trial



Aqua Security is the largest pure-play cloud native security company, providing customers the freedom to innovate and run their businesses with minimal friction. The Aqua Cloud Native Security Platform provides prevention, detection, and response automation across the entire application lifecycle to secure the build, secure cloud infrastructure and secure running workloads wherever they are deployed.

Aqua customers are among the world's largest enterprises in financial services, software, media, manufacturing and retail, with implementations across a broad range of cloud providers and modern technology stacks spanning containers, serverless functions, and cloud VMs.



aquasec.com



contact@aquasec.com



[@AquaSecTeam](https://twitter.com/AquaSecTeam)



[in/Aqua Security](https://in.linkedin.com/company/Aqua%20Security)