WORDSMITH

Preventing Cloud Misconfigurations

Here's an introductory look at cloud misconfigurations and what your organization can do to stop them.

Cloud computing helps security organizations of all sizes scale faster, store more, and innovate quickly. But the promise of speed and flexibility can obscure a critical risk: cloud misconfiguration.

Just how bad is the problem?

> Palo Alto's 2020 research showed that 65% of cloud network security issues also stemmed from user errors and misconfigurations.
>
> Gartner's estimations were even more brutal, projecting that by 2025, 99% of cloud security failures would be the customers' fault—either to suggest that misconfigurations are a pretty concerning issue, or that proportionally, this number will go up as fixes for other types of security breaches are improved. *Source*

You don't need to be a Fortune 500 to suffer the consequences—hackers can target any organization using cloud infrastructure. A single unchecked permission, an exposed storage bucket, or an open port can become a doorway for attackers—and you may not know it until your data is already gone.

What is a Cloud Misconfiguration?

Misconfigurations happen when cloud resources—like databases, file storage, or access controls—are set up improperly so that they unintentionally expose the organization to risk. In simple terms, ***misconfigurations happen when something is set up to work but not secured to protect.***

Common examples include:

- Public storage buckets on AWS S3, Azure Blob, or Google Cloud Storage are accessible without authentication.
- Over-permissioned IAM roles giving users more access than they need.
- Disabled logging and monitoring means there's no visibility into who accessed what, or when.
- Open ports on virtual machines or databases allow external scanning and intrusion.
- Default credentials are left unchanged or are reused across environments.

Why It Matters

Malware or phishing attacks often require user action, but *cloud misconfigurations are passive threats*. They sit quietly until someone finds them—and that "someone" is often a black hat using automated tools to scan for these precise weaknesses.

Consequences can include breaches, financial losses, brand damage and erosion of customer trust, and compliance violations.

Two Case Studies

When cloud security breaches happen, things can get very ugly.

Capital One: In 2019, a hacker exploited a server-side request forgery (SSRF) vulnerability in a misconfigured AWS Web application firewall with overly permissive identity and access management (IAM) roles, gaining unauthorized access to sensitive customer data. The breach exposed personally identifiable information (PII) from over 100 million customers, including names, addresses, credit scores, and more. The attacker exfiltrated data over several months before being detected.

U.S. Department of Defense: In 2023, a misconfigured Microsoft Azure server exposed thousands of sensitive military emails and documents to the public. The data, including personnel information, was accessible on the internet for two weeks before being identified by a non-government source.

Best Practices to Prevent Cloud Misconfigurations

By now you're hopefully convinced of the seriousness of misconfigurations and are wondering how to protect yourself and your organization. Here are some best practices.

Follow the principle of least privilege: Users and services should only be assigned the access they absolutely need—no more, no less. Over-permissioned roles create unnecessary exposure points that attackers can exploit if credentials are compromised.

Enable multi-factor authentication (MFA) everywhere: MFA adds a second layer of verification, making it significantly harder for attackers to access cloud consoles even if usernames and passwords are leaked or stolen.

Encrypt all data, in transit and at rest: Encryption ensures that data remains unreadable to unauthorized users, whether it's moving between services or stored on a disk. Enable default encryption settings in all cloud services and verify key management policies.

Use infrastructure-as-code (IaC) with policy checks: Codify cloud configurations using tools like Terraform or AWS CloudFormation. Scan your templates for misconfigurations before deployment using policy-as-code tools (e.g., Checkov, tfsec) to catch errors early.

Implement continuous configuration monitoring: Automated tools should monitor for changes and misconfigurations in real time. Services like AWS Config or GCP Security Command Center help identify and alert on policy violations immediately.

Regularly audit and pen-test your cloud environment: Security assessments—both manual and automated—help uncover gaps in configuration, access control, and visibility. Periodic penetration testing validates whether an attacker could exploit missteps. **Note:** "regularly" means *recurring pen testing*, which tests frequently—potentially quarterly or even monthly—depending on the organization's risk profile and regulatory requirements. Too many organizations only test once per year to fulfill minimal compliance requirements.

Use AWS CloudTrail, Azure Monitor, or Google Cloud Audit Logs to capture access and activity data: Logging is your forensic trail. Ensure logs are enabled for critical services, stored securely, and regularly reviewed, either manually or through SIEM tools, to detect abnormal patterns or unauthorized access.

Foster a culture of cybersecurity: Misconfigurations aren't a technology issue, they're a *people* issue, often resulting from rushed timelines, unclear responsibilities, or a lack of security training. To address this, CISOs should ensure that cloud security is a core element in onboarding. Organizations can also benefit by encouraging peer reviews and scheduling monthly security stand-ups.

Next Steps

Cloud misconfigurations are among the most common, and most preventable, causes of data breaches. And as cloud environments grow in scale and complexity, the potential blast radius of a single misstep increases.

The team should begin by asking these questions about each resource:

- Is this resource publicly accessible?
- Have we set the right access controls?
- Is data encryption enabled?
- Is logging turned on and monitored?
- Have we tested this configuration in a staging environment?
- Has it been reviewed by someone else?
- Is this service included in our backup and recovery plans?

Finally, here are a few things you can do *today* to reduce your risk:

- Audit your cloud access and storage permissions.
- Enable MFA and encryption everywhere.
- Start using IaC with policy checks.
- Deploy continuous monitoring.
- Create a team culture where security is everyone’s job.