Snyk Top 10:

Strategies When Migrating to AWS

Snyk Top 10

Strategies When Migrating to AWS

Organizations use Amazon Web Services — the dominant cloud service provider — to innovate faster and accelerate their digital transformation efforts. But traditional security processes will become the rate-limiting factor for speed and success in the cloud. Transforming how you do security at the beginning of your AWS journey will empower teams to deliver applications and features faster and more securely — and help ensure a successful digital transformation.

1. Approach AppSec and CloudSec together — holistically

Cloud infrastructure is 100% software and a component of the applications that use it. Attackers don't recognize arbitrary boundaries between application and infrastructure, and they will exploit vulnerabilities to move up and down the stack in order to get what they're after. Avoid security silos and assess the security of your cloud environment and software development lifecycle (SDLC) in full context.

2. Embed security teams with AppDev and DevOps teams

Security teams need a complete understanding of their environment in full context involving the system architecture, applications, and all data and how that data is used and stored. When security teams work closely with developers and cloud engineers when designing and developing cloud-based systems, they can build security into every phase of the SDLC to maximize speed and productivity.

3. Modernize app development with security across the SDLC

Moving to AWS presents the opportunity to transform how application development works at your organization to innovate faster than ever with end-to-end security coverage across the SDLC. Build in automated security checks for every pull request to identify security issues prior to integration and establish high-level security visibility as the application is built and tested to confirm the state of its security.

4. Design cloud architecture with security in mind

AWS security is a function of design, so take measures to minimize the potential blast radius of any incursion into your environment. Level up on AWS security architect skills and think like an attacker to identify architectural weaknesses that could be exploited. AWS provides valuable training programs and certifications to help your security team attain the knowledge needed to keep your environment and data safe.

5. Empower developers and cloud engineers to build securely

Developers are in the best — and often only — position to secure their application code and infrastructure as code (IaC) templates (e.g. CloudFormation; AWS CDK, Terraform) prior to deployment. When cloud misconfigurations arise, developers understand best how to fix them without breaking functionionality. Give developers the tools they need to identify and remediate their own code weaknesses and move forward.

6. Use infrastructure as code right from the start — and secure it

Using infrastructure as code is not only a more efficient and consistent way to build and manage AWS environments at scale, it also provides the opportunity to check cloud security pre-deployment. Checking the security of IaC results in a 70% median reduction of cloud misconfiguration — and a 70% median improvement in engineering productivity and deployment speed.

7. Use cloud security guardrails in deployment pipelines

Use automated checks in CI/CD to prevent misconfigurations pre-deployment and continuously monitor your environment to catch security issues that slip through. By applying a DevSecOps approach to cloud security, engineers get automated feedback and clear guidance on how to correct issues quickly and safely. And when IaC is used, engineers can avoid repeating the deployment of similar misconfigurations later.

8. Build cloud security on a foundation of policy as code

Policy as code (PaC) makes it possible to express security and compliance rules in a language that applications can use to validate correctness. Use PaC to align all cloud stakeholders — including developers, security, DevOps, and compliance — under a single source of truth on policy. Policy-driven automation ensures consistent and efficient enforcement and helps security teams scale their effort, without scaling up headcount.

9. Use Identity and Access Management services securely

Identity and Access Management (IAM) services are about much more than managing team access and privileges — IAM effectively serves as the network in cloud-based systems. Attackers exploit overly-permissive IAM configurations to gain access to the cloud control plane for discovery and movement. Continuously evaluate IAM configurations for weaknesses, and help engineers use IAM services securely.

10. Decide what matters — and measure continuously

AWS security efforts fail when teams aren't tracking important metrics and don't know how much they don't know. Teams that get AWS security right operationalize it and are disciplined about measuring what matters. They know the state of their security posture and can demonstrate it at any time. They measure their progress not only by reduction of risk, but by how much faster and more productive developers and DevOps teams are.