



Tamnoon's Ultimate Guide to Cloud-Based Remediation



Cloud Security is Evolving Fast, and it's Time for Cloud Remediation to Evolve with it.

The explosion of cloud adoption has led to a staggering rise in complexity for security teams.

Over 80% of organizations now use multiple public or private clouds, with over a quarter managing more than 20 distinct cloud security policies. This complexity has resulted in an explosion of misconfigurations and vulnerabilities - 69% of companies report that multi-cloud security gaps have led directly to data breaches or exposures.

Gartner estimates that by 2025, 99% of organizations' cloud breaches will stem from avoidable end user mistakes and misconfigurations.

Unfortunately, conventional remediation approaches fall short in the face of this complexity. The average security team experiences over 50 cloud misconfigurations per day. As per Cloud Threat Report Volume 7, "On average, security teams take 145 hours (approximately 6 days) to resolve a security alert. Well over half (60%) of organizations take longer than four days to resolve security issues."

Clearly, current human-driven processes don't scale. We leave our expanding cloud attack surface unprotected, and it is worsened by increasing volumes of vulnerabilities and misconfigurations remaining unsolved.

At the same time, fully-automated approaches lack the context needed to remediate cloud risks without negatively impacting critical production services. The results are disappointing: more outages, lower security, drained productivity, and a relationship of distrust between security and DevOps.

Cloud-native application protection platforms (CNAPP) and **cloud security posture management (CSPM)** provide a boon of opportunities – not just for developers, but for SecOps. As a security community, we see the value of a new approach to cloud remediation – one that combines human cloud expertise with AI-powered technology purpose-built for the cloud, to drive remediation at scale.

That's why we wrote this Ultimate Guide to Remediation: because **we know there's a better way.**

This Ultimate Guide shares what we've learned from helping our customers fix their cloud security – and observing the gaps and pitfalls in their remediation approaches along the way. It maps the remediation lifecycle in detail, providing a framework that any organization can follow to simultaneously enhance its cloud security posture *and* improve the resilience of production environments.

Here's where we're going:

In Part 1:

- **Chapter 1:** Where Current Remediation Approaches Fall Short
- **Chapter 2:** Sharpening Your Focus (Triage and Prioritization)
- **Chapter 3:** Mapping the Impact (Impact Analysis)

In Part 2:

- **Chapter 4:** Neutralizing the Threat (Resolution)
- **Chapter 5:** Building Resilience Against Recurrence (Prevention)
- **Conclusion:** Remediation For a New Era of Cloud Security

Our core belief is this: *Remediation is a “team sport” that requires not just the right technology – but the right processes as well.*

Let's get going.

Where Current Remediation Approaches Fall Short

The last few years have seen the emergence of a plethora of cloud security tools designed to provide increased visibility and automated guardrails for cloud environments. Solutions like cloud-native application protection platforms (CNAPP), cloud security posture management (CSPM), and cloud infrastructure entitlement management (CIEM) can detect misconfigurations, prevent identity breaches, and even auto-remediate certain issues. (You can read more on the specific goals and scope of each [here](#).)

However, while these tools identify an impressive breadth of risks, companies still struggle to operationalize remediation at scale.

As noted in the introduction, the average organization sees up to 50 misconfigurations per day, with security teams often bogged down triaging each alert manually. At this pace, critical cloud vulnerabilities persist for weeks or months, leading to data exposures and outages down the line. This persistence of vulnerability is known as **dwell time**, or the period between the introduction of a given misconfiguration and its remediation. We can represent this as **Time to Detect (TTD) + Time to Remediate (TTR)**. More so than any other, this is the metric that definitively says how long an organization was vulnerable.

It's clear that despite powerful security tooling, there are missing elements in the remediation process. Teams lack context, orchestration, owner identification and thoughtful prioritization to turn insights into resolved risks. The remediation lifecycle itself remains manual, inconsistent, and painfully slow.

Two Extremes: Neither Sufficient

There are many deployment models for cloud. For example, using Infrastructure as Code (IAC) may create opportunities to remediate misconfigurations by changing the source code. This document is focused on the process while acknowledging diversity in specific remediation techniques.

Today's landscape offers two extremes for remediation, both insufficient on its own:

Manual remediation

Leverages human expertise to carefully contextualize and fix issues, but can't handle the overwhelming volume of cloud misconfigurations.

This approach offers two models:

1

DevOps teams resolve misconfigured resources.

2

Dedicated cloud security engineers or SOC analysts own remediation.

Pure automation

Provides efficiency yet often causes incidents by blindly enforcing changes without considering unique environmental context and deployment methods.

The table below summarizes each approach and the pros and cons of each:

Manual remediation by experts

Remediation by DevOps

What it involves

In a DevOps remediation model, the team that owns the misconfigured resource is responsible for fixing it.

Benefits

Teams can more easily prioritize alerts for their workload.

Drawbacks

DevOps teams often have little insight into how the inherited environment affects the misconfiguration impact from a security perspective.

Manual remediation by experts

Remediation by cloud security experts or SOC analysts

What it involves

Cloud Security experts, or SOC analysts reviewing security alerts, making prioritization decisions, and manually overseeing the remediation process.

Benefits

Cloud Security experts are responsible for prioritization, and help focus developers on the immediate priorities. This model is thorough, ensuring that security risks are addressed holistically.

Drawbacks

Not scalable – slow and inconsistent workflow leaves critical risks unremediated. No workload context, and lack of permissions needed to evaluate that the production process is not going to be affected.

Pure automation

What it involves

Automation-based remediation uses ticket creation and API-based flows to execute simple what-if based actions – classifying, prioritizing, and automatically driving the remediation process.

Benefits

Efficient, enables teams to rapidly work through security alert backlogs and remediate at scale.

Drawbacks

Risks unexpected breaking changes due to lacking environment and process context.

The limitations of each approach on its own become clear when we look at the data.

The average security team spends over 75% of its time triaging alerts – meaning that no team of experts alone can handle the deluge of incoming cloud security alerts generated by CNAPP, CSPM, and other cloud security tooling.

And stepping through specific, familiar use cases will convince even the most automation-happy security teams that pure automation-powered approaches can't get the job done either. See the inset ("Pitfalls of a purely automated approach: RDS encryption") for an overview of the complexities involved in a seemingly straightforward remediation task.

Pitfalls of a purely automated approach: RDS encryption

The limits of infrastructure as code emerge clearly when attempting to remediate something like an unencrypted Amazon Relational Database Service (RDS) instance. While infrastructure as code is invaluable for easily replicating environments, you simply cannot retroactively encrypt a live RDS database through configuration scripts or pipelines.

The process requires:



Snapshotting the existing RDS instance



Creating an encrypted copy of the snapshot



Restoring a new RDS instance from the encrypted snapshot



Migrating applications to use the new encrypted database

This workflow has major implications on production systems, requiring careful change control processes for execution and rollback. Proper service migration also needs extensive planning and validation to avoid disruption.

In other words: securely enhancing critical foundational services demands human guidance, cross-functional impact analysis, and orchestrated automation.

The Bottom Line

Modern cloud environments require a remediation approach that balances both security and production – coordinating insights from tools with human guidance to drive outcomes.

The 4 Pillars of Cloud Remediation

In working with our customers to systematically strengthen their cloud security posture, we've observed that the single biggest "failure mode" is treating the remediation *process* itself as an afterthought.

What does this mean? Traditional cloud remediation experiences four primary challenges. First, the disjointed nature of security tools and compliance standards may duplicate security alerts in a non-obvious way. For example, one compliance standard might issue an alert on an AWS security group open to all ports, protocols, and IP addresses, while another issues an alert because the security group allows access on a database port. They are separate alerts, but alerting about the same security misconfiguration. The inflated volume of alerts creates additional noise for remediation teams to sift through.

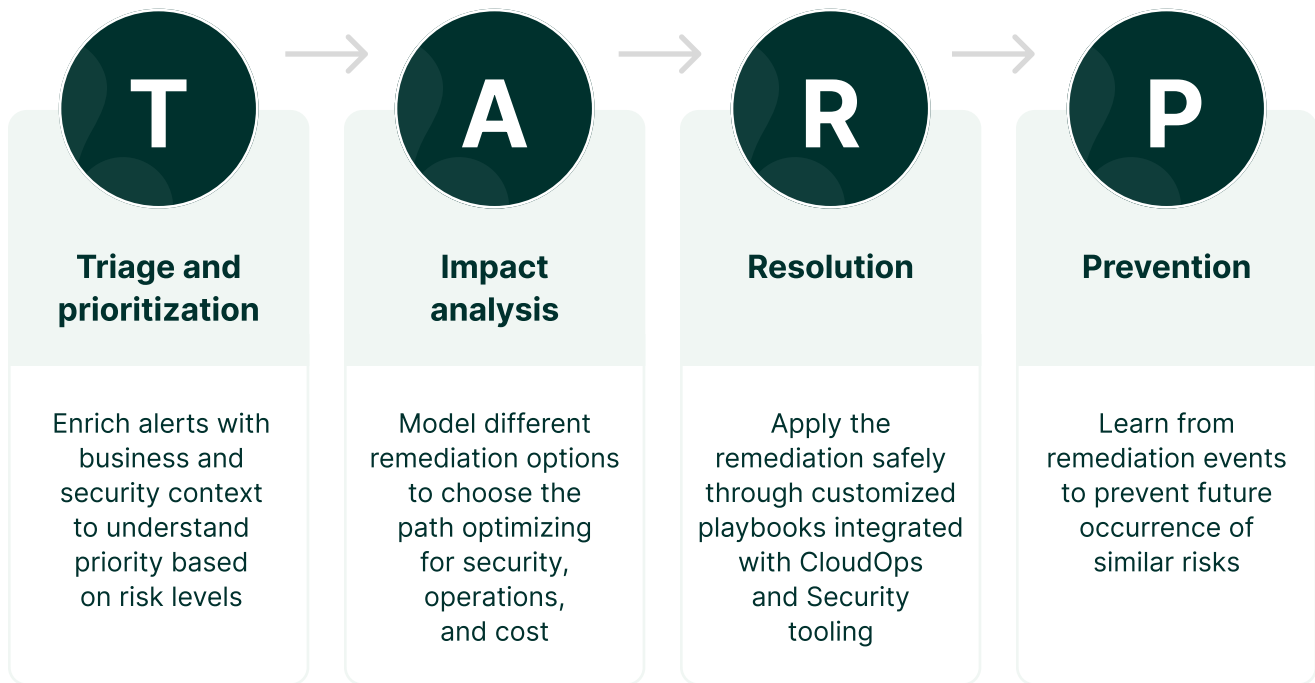
Second, traditional cloud remediation typically resources from teams that may be reprioritized, resulting in increased dwell time. Few organizations raise cloud security remediation to a priority that keeps those resources dedicated to the mission.

Third, recruiting and retaining cloud and cybersecurity talent continues to be a challenge for most organizations. Dwell time increases as teams take on unfamiliar challenges. High impact remediation efforts may be deprioritized because teams lack the confidence to deploy without operational impact.

Last, the process for cloud security remediation differs from traditional remediation. At a high level, the processes are similar, but the details about how the cloud is deployed and managed interrupt traditional remediation processes.

We believe that an effective cloud remediation workflow rests on four pillars. We call this Tamnoon's TARP process for the full remediation lifecycle.

Tamnoon's TARP Process for the Remediation Lifecycle



Delivering on all four pillars requires tight humans-in-the-loop guidance coupled with automation to create a coordinated, scalable process.

The following chapters will explore TARP in depth – with the goal of helping every organization (regardless of what specific technologies or solutions they use) enhance their remediation processes.

- What does the stage involve? Why does it matter?
- What are traditional approaches to each stage, and where do they fall short?
- How should organizations think about objectives, process, and technology for the stage?

Sharpening Your Focus (Triage and Prioritization)

Cloud adoption is growing and security tooling is getting better at creating visibility into cloud vulnerabilities. As a result, the volume of security alerts generated by your CNAPP and/or CSPM likely exceeds your capacity to respond to them. The alerts from your cloud-native application protection (CNAPP), cloud security posture management (CSPM), and other tools, report issues ranging from misconfigurations to vulnerabilities. Security tools do their best to express risk to help inform prioritization efforts, but lack the business context to be useful without human triage. It is common for security teams to be unable to immediately deal with every alert and its corresponding misconfiguration – and when you're dealing with thousands of alerts daily, fatigue can settle in as well.

In order to focus a team on what is most critical and urgent, the cloud remediation process frequently begins with a triage and prioritization step. Triage and prioritization provide a systematic way to optimize how your team handles this flood of notifications.

What Does Triage and Prioritization Involve?

Triage and prioritization aim to ensure your team works on the most critical risks first. This prevents spending time fixing low-impact problems while major threats go unaddressed. An effective triage and prioritization process involves:

- Gathering context to understand the scope and impact of each alert within the business context
- Ranking assets based on criticality to the business in order focus on safeguarding your "crown jewels"
- Estimating risk levels associated with each issue
- Identifying patterns in alerts to address systemic weaknesses
- Assigning ownership of the issue to the right individual or team
- Performing robust alert validation to audit for false positives, behaviors by design, and accepted risk
- Streamlining and improving incident response flows
- Leveraging automation and human expertise to enhance analysis

Establishing a rigorous triage and prioritization workflow is essential for working through your cloud environment's security challenges efficiently and effectively. Let's explore the key steps involved.

Key Steps for Effective Triage and Prioritization

1. Add Context to Alerts

Typically, CNAPP, CSPM, and other tools provide alerts with basic context. However, understanding the scope and impact of each notification, as well as the business relevance of the underlying asset is crucial for assignment and prioritization. For example, an open security group is bad – but an open security group sitting in front of an EC2 instance with an application vulnerable to RCE/SSRF (remote code execution / server-side request forgery) is catastrophic. Enriching raw alerts by gathering details like affected resources, configurations, compliance, and potential business impact will help remediation teams to either understand the broader impact of any changes they implement, or conduct further analysis that may alter their remediation plan.

For example, assume you are using a CNAPP to monitor for cloud vulnerabilities. You receive a cluster of related alerts around improper bucket permissions allowing public access. The CNAPP provides the affected resources – in this example, S3 buckets – but not the importance of those resources to the business or the data sensitivity. Your team would need to dig deeper to determine the scope of the exposure and potential business impact.

This context is difficult to automate entirely. While tools can provide some supplementary information, human oversight is invaluable for accurately assessing the potential impact of a fix. Augmenting alerts with additional context sets the foundation for effective responses.





2. Assign Ownership of Remediation

With thousands of alerts each day, determining exactly who handles each one is essential for streamlining the remediation process.

Some common types of ways companies identify ownership include:

- Logs
- Tags
- Organizational charts
- Dedicated identity provider
- Continuous integration and continuous deployment (CI/CD), i.e. who originally deployed the infrastructure

Broadly, companies attempt to use either a manual or automated process to assign ownership of remediation tasks. Each has pros and cons:

APPROACH	APPROACH
Manually assign alerts to relevant devs	Automate assignment of alerts based on predefined rules or heuristics
 Ensures issues are being appropriately assigned based on domain expertise and sensitivity	 Saves time and provides useful audit trails showing who has taken ownership of each incident
 Time consuming and impossible at scale in complex environments	 May incorrectly assign alerts due to business logic that cannot be defined by policies

The most effective approach tends to involve a blend of these approaches. Machine learning and automation can help achieve process efficiencies at scale – while human oversight ensures appropriate routing, especially for alerts with complex business logic associated with them.

3. Prioritize Your Assets

Not all cloud resources and data are created equal. To focus triage efforts, you need to identify and prioritize your most sensitive assets - the "crown jewels" requiring heightened protection.

Examples include databases containing PII, healthcare information, financial data, and intellectual property. You can also prioritize based on resource exposure, encryption status, and other vulnerability factors.

For example, a healthcare company would prioritize systems containing patient health records, clinical trial data, and other highly sensitive information. Similarly, a financial services firm would focus on securing financial transaction data, account information, and intellectual property.

While tools can help classify assets, intricate knowledge of your business, architecture, and data flows are needed to accurately determine sensitivities and priorities. This process is difficult to automate completely and benefits greatly from human input.

4. Categorize Your Assets

In addition to ranking individual assets, higher-level categorization provides further guidance for triage. You can segment your cloud resources and data by factors like importance, sensitivity, and compliance requirements.

For instance, you may categorize assets as mission critical, business critical, and non-critical. Similarly, labeling them by data classifications (PII, financial, public) allows appropriate security handling.

Tools can suggest categories, but only organizational stakeholders truly understand complex business needs. The most effective methods combine machine learning with human judgment.

5. Evaluate Asset Risk

Too often, we conflate asset risk with alert risk. Alerts do their best to assess risk using the context available, but the asset risk may be different than the alert risk.

For example, an alert on an asset in the corporate network may be assigned a lower risk without consideration that the asset is used by a developer to commit code to the company's flagship product. A robust remediation process must prioritize the asset risk, which may be informed by the alert risk.

The disastrous Capital One data breach from 2019 exemplifies the importance of prioritization that evaluates the risk posed to assets based on their context within the environment. In the C1 instance, S3 buckets containing sensitive data were being stored in the same account as DMZ security appliances running on vulnerable, internet-facing EC2 with instance profiles that allowed S3 access.¹

With your prioritized and categorized assets mapped, you can assess the risk associated with each to guide the order of response. Resources containing highly sensitive data will warrant rapid triage of related alerts.

¹Novaes Neto, N. Madnick, S. Moraes G. de Paula, A. ["A Case Study of the Capital One Data Breach."](#) Cybersecurity at MIT Sloan, MIT Sloan School of Management, Massachusetts Institute of Technology.

For example, an alert related to improper access controls on databases storing customer credit card data would be assigned a high risk score demanding prompt investigation. On the other hand, an alert about a public-facing content bucket without sensitive data may carry lower risk.

Certain categories like unencrypted PII may require immediate investigation, while public-facing content with exposure risks might permit a slower approach. Repeated failures to address risks in a timely manner would escalate the triage priority.

Here again, automated risk scoring provides a useful starting point but human experts add crucial nuance based on business impact. Use the best of both worlds.

6. Streamline Incident Response

Triage also involves eliminating duplicate alerts and refining categorization to streamline incident response. Deduplicating notifications, classifying them by severity and type, and resolving false positives improves efficiency.

A CNAPP with multiple compliance frameworks configured may generate an alert for each framework on the same misconfiguration, on the same resource. For example, your CNAPP may generate multiple alerts around the same public bucket access issue. Your security team would need to analyze and deduplicate these related alerts, rather than waste time investigating each one separately.

Tools can deduplicate basic alerts, but humans identify more complex duplicate incidents spanning multiple systems. For categorization, the technical expertise of engineers is invaluable — automation alone cannot reliably classify intricate cloud security notifications.

7. Identify Patterns

Effective triage looks beyond individual alerts to identify broader patterns and systemic weaknesses. Analyzing alerts collectively reveals issues like recurring misconfigurations requiring updated configurations, security guardrails, or user training.

For instance, frequent alerts around improperly configured bucket permissions may point to a need for better cloud security training on access controls. Improper SSH key rotation alerts may indicate a gap in administrator practices that should be addressed through better training on key lifecycle management.

Without examining the bigger picture, you cannot address the root causes behind volumes of alerts.

While machine learning has become quite adept at finding patterns, human security experts interpret these signals best in terms of potential root causes and solutions. This high-level insight is very difficult for machines to match. A robust triage and prioritization process leverages both machine learning to classify and trend while a skilled human uses their knowledge and experience to operationalize the machine learning outputs.

The Key Role of Human Experts

As the examples above demonstrate, although automation provides a solid starting point, human expertise is indispensable for effective security triage and prioritization. The contextual understanding, architectural intuition, and risk insights of experienced cloud security professionals remain impossible for tools to replicate independently.

APPROACH

Expert



Allows nuanced human judgment to assess and prioritize complex issues



Custom prioritization based on intimate knowledge of architecture and data flows



Effective communication of insights across teams



Dependent on individual specialized expertise – human behavior generates inconsistencies, even between two great security engineers



Very time consuming and labor intensive, cannot scale



Inconsistent processes prone to human error and oversight



Fails to account for constant cloud evolution, missing new resources/services



Provides rapid, consistent prioritization at scale



Automates repetitive tasks to save time



Aggregates huge data sets to reveal insights faster



Automatically suggests assignments and priorities



Lacks nuanced oversight, potentially leading to inaccurate prioritization



Standard tools fall short of tailored solutions



May incorrectly assign/prioritize complex issues

By combining smart software with human oversight, you can optimize focus on your cloud environment's most critical risks and streamline response efforts. The future lies in this potent blend of machine and human capabilities - amplifying the strengths of each.

Triage and Prioritization in Action: A Case Study

Many enterprises using AWS have a mix of Elastic Block Store (EBS) volumes, some encrypted and some unencrypted. How might the security team go about the process of triage and prioritization for the remediation of unencrypted volumes to strengthen its security and data protection?

Typically the team would use detection tools such as AWS Config or off the shelf CSPM tools to continuously detect which EBS volumes are unencrypted. This generates alerts for any volumes not complying with encryption requirements.

Here's how the team might then go about triage and prioritization for this issue:

Triage and prioritization step	Why it matters
The security team enriches these alerts by identifying which applications and data sets depend on the affected EBS volumes.	Provides crucial context on potential impact
The team uses an automated ticketing system to assign unencrypted volume alerts to the appropriate AWS owner for remediation. Ownership can be identified based on AWS tagging strategies or external sources such as CMDB tools.	Ensures that the most sensitive problems are being assigned to the teams best equipped to address them
The security team has categorized the crown jewels data sets stored on EBS, like PII, financial data, and intellectual property. Related encryption alerts get high priority scores.	Aligns the team's activities around the most business-critical risks
To streamline response, duplicate EBS encryption alerts are merged into a single ticket. And false positives are filtered out.	Prevents duplicate work and improves efficiency of remediation workflow
Analyzing encryption alerts collectively revealed an upward trend due to new volumes getting created without encryption enabled by default. To address this pattern, the team now uses a policy to block creation of unencrypted volumes.	Reduce risk by preventing recurrence of the issue

By leveraging both automation and human expertise, an organization can stay on top of encrypting EBS volumes across its dynamic AWS footprint. The triage and prioritization process has been key to this success.

The Bottom Line

Cloud remediation is a complex undertaking, with new alerts and issues constantly arising that require a well thought out, documented, and repeatable process. Triage and prioritization are just the first step in this process.

For Practitioners

For security analysts and engineers, optimized triage and prioritization improves workflow efficiency, team communication, and clarifies response steps.

Automating repetitive enrichment and assignment tasks enables practitioners to focus expertise on complex judgment calls. Well-defined processes also create smoother hand-offs between different teams. With mature triage fundamentals in place, practitioners gain better leverage over the daily flood of security alerts.

For Management

For leadership, effective triage and prioritization deliver measurable improvements in risk reduction over time. Metrics around response times, security ticket backlogs, and overall security posture will steadily improve. Automated enrichment provides management more insight into response workflows. Leadership can scale cloud security with confidence knowing that the highest priority threats are addressed first.

Mapping the Impact (Impact Analysis)

What is Impact Analysis?

Performing an impact analysis is a critical step in the cloud remediation process that employs methodical techniques to answer the questions: **"What might go wrong if we implement this fix?"** and the equally-important **"What might go wrong if we *don't*?"**

A comprehensive impact analysis not only highlights the possible ramifications of altering cloud-based resources and configurations, but also offers a holistic view of the risks and benefits to your operations, cost, security, and compliance with each remediation task.

Some fixes may have no broader impact at all while others may crash the production environment for anywhere ranging from minutes to hours, so it's vital to understand the potential consequences *before* implementing any changes. This way, you can ensure that security fixes don't unexpectedly disrupt your business operations, put you at risk of noncompliance, or cost your organizations.

The overarching goal of impact analysis is to enable organizations to make informed, data-driven decisions about the best approach to cloud remediation, while fully grasping the business tradeoffs. When done well, it sets the stage for a seamless remediation process.

Key Steps

Planning

Once you've identified security issues, your very next step should be to make a plan. A well-crafted remediation plan should unfold in distinct phases, starting with matters that have minimal production impact first and then, as the process progresses, delving into more complex challenges. Each of these phases should be mapped out and detail the specific steps you or your team needs to take to address each security concern.

This structured approach ensures that your remediation efforts are both comprehensive and well-organized with minimal business impact (see [Chapter 2](#) for more on how to do this systematically).

For example, let's say you need to encrypt an existing Amazon Relational Database Service (Amazon RDS) for PostgreSQL DB instance in the Amazon Web Services (AWS) Cloud with minimal downtime.

After it's created, you can no longer add encryption to an Amazon RDS DB instance. But you *can* encrypt a snapshot of your unencrypted DB instance and then restore it from the snapshot to get an encrypted copy of your original DB instance.

In the planning stage, the security team would meet with the database admins and application owners to plan the process of encrypting the RDS database instance. They would review the current database architecture, security requirements, and potential downtime. The plan would need to outline the steps for creating a snapshot, encrypting it, spinning up a new encrypted instance, and cutting over.

It should also cover validating data replication, disabling constraints, and updating applications.

Documenting Your Plan

Documenting your remediation plan is essential. Documentation not only serves as a roadmap for the remediation process but also fosters clear communication across various teams. A detailed plan will act as both a guide you can reuse in the future and a communication tool, playing an indispensable role in successful remediation.

In our Amazon RDS example, the security lead would document the encryption plan in a shared wiki or document, with details on timelines, resources required, risks, and mitigation strategies. This would be circulated to all stakeholders for review and sign-off before starting the work.

Roles and Responsibilities

Define roles and responsibilities as part of the plan. By meticulously laying out the required steps and procedures, all members — from SecOps to DevOps — gain clarity on their roles and responsibilities. This type of transparency ensures that teams are aligned, enhancing your remediation effort's effectiveness and efficiency.

Consider again the Amazon RDS example. A clear plan would layout each team members role, such as:



Remediation Impact Analysis

Each fix has its own unique set of repercussions. Some might seamlessly integrate with no noticeable effect on your business environment, while others may introduce downtime ranging from a few minutes to several hours — halting production and impacting trust with your users or customers. By systematically working from tasks with the least impact to those with the most, you can prioritize actions, safeguard operations, and maximize your uptime.

Because potential outcomes vary greatly, conducting a detailed impact analysis is the first line of defense against unintended outcomes and is indispensable for informed decision-making.

Continuing with Amazon RDS, the team would analyze the impact of the RDS encryption on overall database performance, application latency, and potential downtime. They would test encryption on development environments first and then they would assess the app's ability to failover and point to the new endpoint.

Role of the Expert vs. Automation

A common dilemma for organizations is choosing between fixing risks manually or deploying fixes automatically. When it comes to impact analysis, exclusively relying on either method poses considerable benefits and challenges.

Manually testing every scenario is impractical, leaving a vast majority of risks unaddressed. On the other hand, automation without human oversight can't accurately predict the unique business implications of changes for your specific business. This may risk destabilizing your production cloud environment.

APPROACH

Expert



Invaluable perspectives and nuance that automated systems may overlook



Anticipate potential challenges and craft customized solutions aligned with business priorities and regulations



Exercise judgment in complex trade-off situations to evaluate broader implications



Devise custom scripts tailored to specific needs



Effectively communicate scenarios across teams



Manually testing every scenario is impractical, leaving many risks unaddressed



Lack of automation makes response times slower



Rapidly scan expansive infrastructures to identify vulnerabilities consistently



Automatically trigger alerts and remedial actions to expedite response times



Aggregate vast amounts of data to reveal insights that would take much longer manually



Implement fixes uniformly at scale



Automation without oversight risks destabilizing environments and compromising production



Inability to appreciate nuances that human experts would identify



Standard tools may fall short of tailored solutions

When encrypting an existing Amazon RDS, the database admins would leverage their expertise to thoroughly test and validate the data replication and app connectivity. Automation could be used for snapshotting, spinning up new instances, running data validation checks, and routing app traffic. But the DBAs' skills would be critical for troubleshooting issues.

The combination of human expertise and automated scanning delivers comprehensive, tailored cloud remediation. Experts contextualize and customize, while automation provides speed, consistency and an information-rich landscape. Together, they offer robust defense.

Impact Analysis in Action: A Case Study

It's not uncommon for an organization to discover unencrypted EBS volumes through security tools like AWS Config. And when they do, it's crucial to perform an impact analysis before remediating, to assess how reliant each affected workload is on the EBS volume, and the potential effects of downtime during re-encryption.

For example, if a volume stores data for a customer-facing application, the security team needs to plan the remediation carefully to avoid disrupting service. To build a well-defined execution plan that addresses each scenario, they may consider the following questions:

1. Is the volume in use?

- a. If not, do we need it?

2. Is it part of an Auto Scaling Group or launch template?

If it is in use, they may schedule the re-encryption during off-peak hours and have a rollback plan ready. On the other hand, if the EBS volume is rarely used, the team can encrypt it anytime with minimal impact.

By considering the unique dependencies and risks for each unencrypted volume before acting, they can remediate this security issue efficiently and optimize uptime. By conducting a thorough impact analysis instead of blindly fixing problems, they prevented unnecessary disruptions to critical business activities.

The Bottom Line

Thorough impact analysis is indispensable for safe and smooth cloud remediation. Carefully evaluating fixes upfront through multiple lenses paves the way for successful changes.

For Practitioners

For security and ops teams, impact analysis prevents disruption by identifying risks pre-implementation. Practitioners can design tailored remediation plans leveraging impact insights across tools, technologies, and processes resulting in a smooth hand-off between teams.

For Management

For leadership, impact analysis provides confidence in change success by quantifying risks. Management gains visibility into the business justifications guiding each remediation decision while also increasing compliance assurance by assessing regulatory impacts.

In Conclusion

Current cloud remediation practices have challenging limitations; even the most advanced tools fall short without strategic human intervention. Through a deep dive into triage and prioritization, we have been able to advocate for a more methodical approach to sift through the noise and emphasize the indispensable role of human expertise in discerning the significance of each alert. As we delve into impact analysis, we advocate for collaboration between automated processes and the judgment of seasoned professionals. Together, we shared a blueprint for a resilient cloud security posture, championed a holistic strategy that marries the precision of automation with the depth of human experience, and set the stage for a more secure and agile cloud ecosystem.

How Tamnoon Helps

Powered by AI and curated by CloudPro experts, Tamnoon's human-centric AI solution connects to your cloud security tools, prioritizes security efforts and focuses your team on the risks that matter. With Tamnoon, SecOps and DevOps teams fix more risks in less time, while limiting the negative impact that configuration changes may cause to their environments. Tamnoon clients see an average 95% reduction in investigation time for alerts and an 85% immediate reduction in critical cloud risks.

Neutralizing the Threat (Resolution)

This is the execution phase that brings your remediation plan to life. In this stage, you'll follow the timeline, steps, and assigned responsibilities that were outlined earlier in the triage and impact analysis phases. Additionally, integrating security checks into CI/CD pipelines is an important part of execution, allowing you to catch vulnerabilities early in the development lifecycle before they reach production.

Smooth remediation requires meticulous coordination across tools, teams, and schedules. The complexity and scale of the remediation process may suggest that only a manual or an automated process can deal with it. Ideally, an organization can leverage the best of each process where appropriate. With so many different dimensions to coordinate, intricate planning and cross-functional excellence is critical to be successful.

The Security / DevOps Relationship

Your security and DevOps teams must work hand-in-hand during this process. And while each team has their distinct domains, they also have shared goals to safeguard data and maintain uptime, meaning that a tight partnership is paramount.

This collaboration does not always come easy. Some common scenarios that can strain relations and undermine progress might be:

- Competing priorities
- Siloed teams
- Different mindsets
- Technical disconnects
- Lack of clear policies
- Unrealistic demands
- Blame culture
- Poor monitoring
- Lack of trust (or confidence)

Nevertheless, it's critical that security and DevOps teams develop a common "language" to achieve their shared goals, which include:

Align | on severity levels, compliance needs, and business justification.

Agree | on optimal coordination model and cadence of communications.

Establish | validation criteria to confirm successful remediation.

Maintain | clear documentation of roles, responsibilities, and risks.

By regularly communicating their priorities, constraints, and risks from both perspectives, and emphasizing their unified mission, DevOps and Security can align on the best approach to streamline remediation while minimizing disruptions. The organizations that have been most successful at bridging the security/DevOps gap follow a framework like the one below for disambiguating roles and forging common ground.

TEAM

Security

Role and required inputs

- Gather intelligence to understand risks, impacts, and technical constraints related to remediation
- Provide clarity on severity levels, compliance implications, and data exposure risks
- Supply guidance to DevOps on safe remediation methods to mitigate risks
- Recommend testing procedures to validate remediation before deployment
- Collaborate with DevOps to optimize coordination and communication

Key questions to ask counterparts

- What risks do you foresee that we need to mitigate?
- What dependencies with other teams or resources do we need to map out?
- How complex will rollback be if any issues do arise? Are there ways we can simplify it?
- Are there any technical limitations that could complicate this fix? If so, are there ways we can work around them?
- What testing procedures do you recommend before we deploy any changes?

Role and required inputs

- Share insights on potential system stability and uptime impacts
- Map out dependencies with other teams and resources
- Devise reliable rollback procedures in case issues arise
- Outline technical limitations and, if any, options to work around them
- Continuously monitor for early warning signs of emerging risks




Key questions to ask counterparts

- How severe and what is the potential business impact of this vulnerability?
- How urgent is remediation? Is there a deadline?
- Will we violate any compliance requirements if this is not addressed?
- Does this issue expose sensitive information?

Key Steps for Secure and Seamless Remediation

Follow the Plan

Smooth execution requires a detailed plan mapping out phases, steps, timelines and responsibilities:

-  Document the steps to be taken to address specific types of security findings (public storage accounts, exposed services, unencrypted volumes, unrotated keys, etc).
-  Define timelines for each phase to provide clarity on the order and pace of remediation.
-  Assign clear ownership and responsibilities for each step to ensure accountability.

It's crucial that security and DevOps teams collaborate closely on constructing this plan. Security provides expertise on managing risk and prioritizing assets while DevOps offers insights on the potential impacts of any changes made.

With both perspectives, the plan can sequence remediation intelligently by starting with high-risk, mission-critical items before moving towards low-impact items.

For example, addressing improperly configured S3 buckets exposing sensitive data in phase 1 before rotating IAM keys in phase 2. By tackling high risk issues first, well-planned remediation systematically strengthens security posture while minimizing disruptions through incremental progress.

Sample Remediation Plan

In this sample plan, security teams have identified a high-risk misconfiguration; open-access S3 buckets which expose sensitive data. Unencrypted RDS instances are generating a moderate risk, threatening a service disruption during cutover. Lastly, IAM keys need to be rotated, which could cause minor operational delays if left undone.

Intelligent remediation prioritizes the reconfiguration of S3 bucket permissions; AWS-authenticated threat actors can read, write, and manipulate objects within the bucket, creating a high-severity risk to the asset and the business.

Armed with guidance on prioritization, the remediation team plans three 2-week sprints to tackle these misconfigurations.

Phase 1 (Weeks 1-2)

Step 1:

Reconfigure public S3 bucket permissions

Task Owner: Security Team

Risks:

High – data exposure if not remediated quickly

Phase 2 (Weeks 3-4)

Step 1:

Encrypt RDS instances

Task Owner: Database Architects

Step 2:

Enable VPC flow logs

Task Owner: Security Team

Risks:

Moderate – service disruption during cutover

Phase 3 (Weeks 5-6)

Step 1:

Rotate IAM keys

Task Owner: DevOps

Risks:

Low – minor operational delays

During these six weeks, Security and DevOps should meet weekly to review:

Overall progress
and timeline

Blockers needing
resolution

Emerging risks
and mitigations

Next steps and
action items

Security and DevOps should also have contingency plans ready to adapt to changes in the remediation plan. To handle plan deviations:

Have risk mitigation strategies

ready. Being prepared with contingency plans for potential issues prevents being caught off guard.

Replan dynamically if any issues

emerge. If surprises come up, be ready to quickly adjust the plan and timelines. Don't rigidly stick to outdated plans.

Adjust timelines as needed.

Related to above, update schedules and phases if delays or changes occur.

Document all modifications.

Track all changes to the plan in a central place for visibility.

Integrate Security into CI/CD Pipeline

Remediation should integrate security checks into existing CI/CD pipelines early to catch issues before they reach production. Scrambling to fix problems right before deployment creates unnecessary risk and delays.

Shifting security left in the pipeline provides multiple touchpoints to rapidly surface and resolve vulnerabilities at each stage of the development lifecycle.

Key techniques include:



Execute infrastructure scans during continuous delivery to catch misconfigurations around encryption, permissions, network rules, and more. This surfaces risks before they impact customers.



Trigger security scans at multiple stages, not just end — failures early in CI are cheaper to fix.



Gate releases if critical vulnerabilities are detected, failing the build. This prevents broken code from being promoted.



Automate scans as much as possible to save time and money.

Role of the Expert vs. Automation

As we have seen from both impact analysis and triage and prioritization, finding the right balance between manual and automated approaches is crucial for effective cloud security. Teams often have to choose between a solely automatic solution, or solely manual. However, relying only on manual processes leaves the majority of risks unaddressed, while automation without oversight risks destabilizing your environment.

APPROACH

Manual remediation



Allows for nuanced human judgment and expertise to assess and address complex issues



Custom fixes can be devised based on intimate knowledge of systems and data flows



Better communication of risks across teams and guidance on safe fixes



Time consuming and labor intensive, cannot scale



Fails to account for constant cloud evolution, missing new resources/services



Inconsistent process prone to human error and oversight



Outdated runbooks may be followed without updates

APPROACH

Automated remediation



Provides rapid, consistent fixes at scale



Automates repetitive tasks to save time



Aggregates huge data sets to reveal security insights



Automatically triggers alerts and remedial actions to accelerate response times



Lacks nuanced human judgment and oversight leading to unintended consequences



Standard tools may fall short of tailored solutions



May incorrectly assign or prioritize complex alerts



Risks destabilizing environments by breaking systems

Remediation in Action: A Case Study

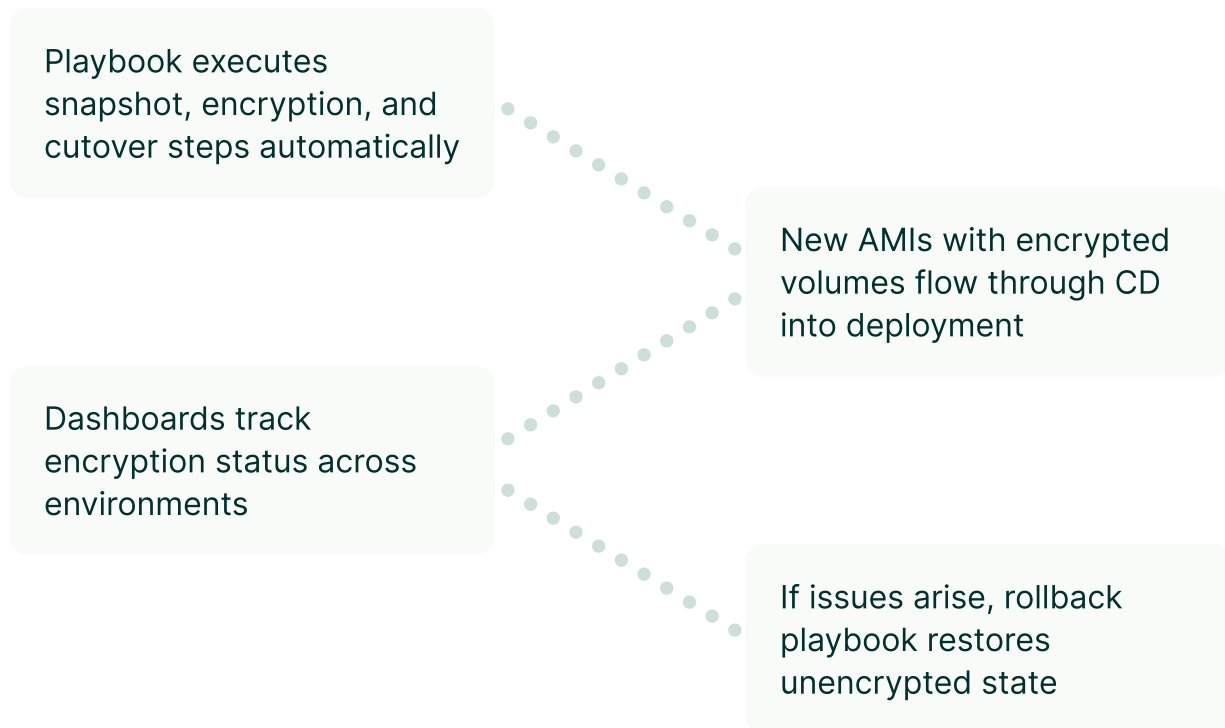
EBS Encryption

Let's walk through an example of providing DevOps with an EBS encryption playbook to fix unencrypted volumes.

First, the security team collaborates closely with DevOps, DBAs, and application owners to create a comprehensive EBS encryption plan covering:

- ✓ Thorough testing in lower environments to validate performance impact and workflow
- ✓ Documenting the step-by-step process for snapshotting, encrypting, spinning up new volumes, cutting over apps, and validating data replication
- ✓ DBAs providing needs around potential downtime based on sizes and change windows
- ✓ Application owners assessing failover capabilities and endpoint update needs
- ✓ Rollback plans in case of unforeseen app errors after cutover

Next, DevOps integrates the playbook into the CI/CD pipeline:



This end-to-end example demonstrates:

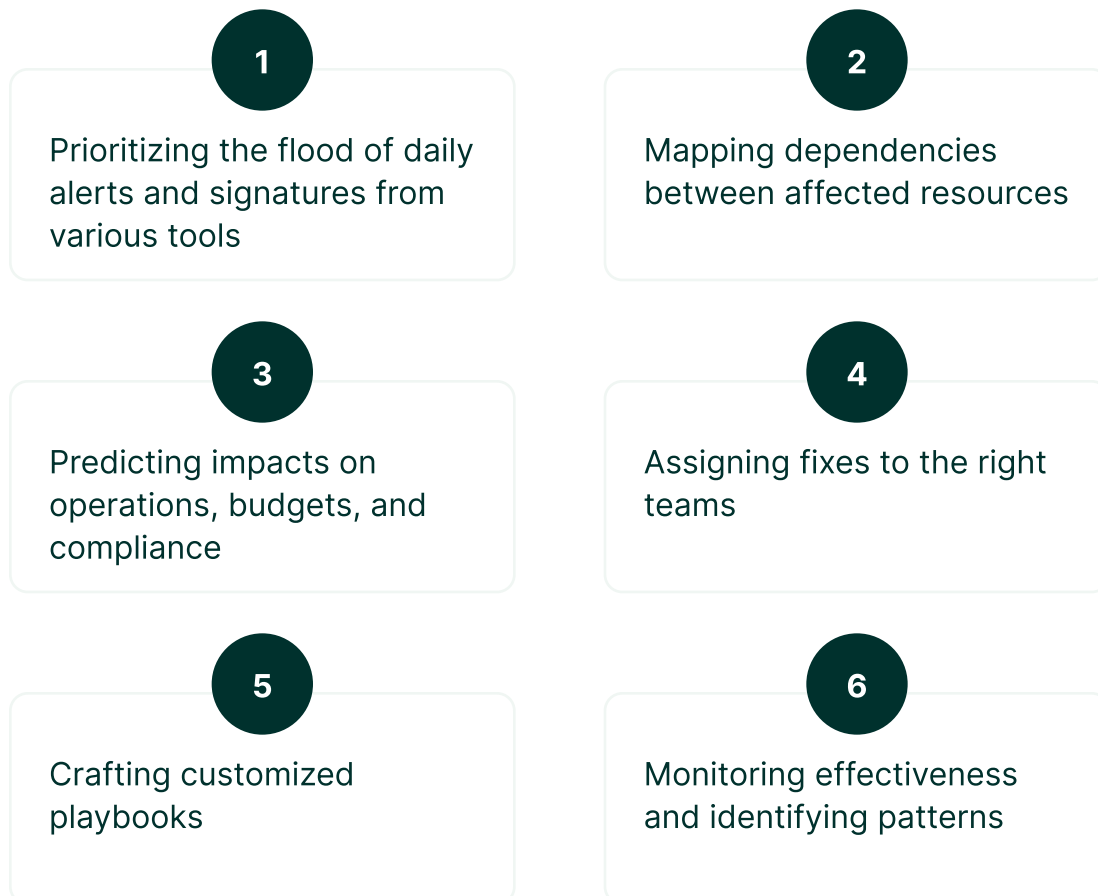
- Cross-team collaboration to create a plan balancing security and uptime
- DevOps expertise to automate playbook execution at scale
- CI/CD integration for rapid, consistent remediation
- Rollback procedures to mitigate risk

With this playbook, DevOps can remediate unencrypted EBS volumes rapidly and safely. Automation provides speed and consistency while collaboration with security ensures a smooth process tailored to the organization's needs.

The Bottom Line

Effective remediation requires careful orchestration across tools, processes, and teams to address security issues.

Consider just some of the challenges:



When addressing complex remediation, teams often face a tradeoff between relying solely on manual fixes or full automation. Manual remediation allows for nuanced human judgment but is inconsistent and doesn't scale. Automation provides speed and consistency but lacks oversight.

The most balanced approach likely integrates automation and human guidance based on the situation. By playing to the strengths of both skills and software, the potential rewards are substantial: robust security, streamlined operations, and improved compliance.

For Practitioners

For security and DevOps practitioners, an optimized remediation process improves efficiency, reduces disruption, and clarifies responsibilities. By automating repetitive tasks and detailing procedures, practitioners complete remediation faster with less overhead. They can focus their expertise on higher-value initiatives. Shared playbooks also minimize business impact and reduce friction between teams. Smooth remediation increases practitioner productivity and effectiveness.

For Management

For leadership, balanced remediation strengthens cloud security posture while optimizing IT operations. Instead of slow and inconsistent manual fixes, scalable automation remediates issues rapidly and consistently. Metrics like remediation times, security ticket backlog, and audit findings improve. Compliance assurance also increases through automated policy enforcement. Leadership gains confidence that both existing and emerging threats will be addressed quickly and effectively through mature remediation practices.

Building Resilience Against Recurrence (Prevention)

In the fast-evolving cloud security landscape, successful remediation isn't just about fixing issues when they arise – it's equally about preventing the recurrence of these issues.

Prevention is the final, critical stage of the cloud security remediation process. After a specific threat or vulnerability has been addressed, prevention focuses on reducing the likelihood of that issue happening again. The goal is to implement systematic safeguards, processes, and controls to stop the same problems from recurring.

What Prevention Is: Reducing Recurrence Systematically

Prevention is the strategic deployment of policies and practices aimed at reducing the likelihood of security issues recurring in the cloud environment. In this context, prevention is not a set of static rules but a dynamic, evolving process that adapts to the ever-changing threat landscape.

To achieve effective prevention, organizations focus on establishing policies that ensure DevOps teams do not inadvertently replicate errors or misconfigurations. This involves a collaborative effort where security works with DevOps to build and enforce policies aligned with organizational goals.

Policies: Bridging the Gap Between Security and DevOps

The relationship between security and DevOps can sometimes be fraught due to differing priorities and approaches. DevOps teams often prioritize speed, innovation, and continuous delivery, while security teams are focused on risk mitigation, compliance, and safeguarding sensitive data. Without the right processes and policies in place, these goals can be at odds, potentially leading to tension and delays in the development lifecycle.

Consider a scenario where a DevOps team is under pressure to release a critical feature within a tight deadline. In pursuit of speed, they may bypass certain security protocols, such as rigorous testing or compliance checks. While this enables rapid deployment, it introduces security vulnerabilities.

Without clear policies and communication, such actions can lead to a compromise between speed and security, creating tension between the two teams.

The challenge is exacerbated by common pitfalls in how security teams approach policy development and enforcement. Such approaches tend to rely entirely on manual work (which doesn't scale effectively) or on automated, machine learning-driven approaches (which – without proper context – can negatively impact production).

Where Current Approaches Fall Short

Manual Policy Creation is Time-Consuming

Traditionally, security teams attempt to prevent recurrence of an issue by manually creating new policies and processes. For example, after remediating incorrect IAM permissions, they may develop a policy dictating proper permission granting.

However, manual policy creation is extremely time-intensive and risks being too narrow or inflexible to prevent variants of similar issues. In a rapidly scaling infrastructure, manually updating access control policies for each resource can result in delays and inconsistencies. This lag in policy adaptation may expose the organization to vulnerabilities.

Consider a scenario where a manual policy dictates specific firewall rules for a set of applications. As the organization expands, new applications are added, but the manual policy update process lags behind. The result is an inconsistent application of security policies, leaving some resources exposed due to outdated rules.

Automation Risks Over-Enforcement Without Context

On the other end of the spectrum, some organizations use automated prevention methods based on artificial intelligence. While this increases speed, these systems often lack nuanced context about the environment and business needs. As a result, they risk over-enforcing restrictions or recommending overly broad changes that negatively impact operations.

Imagine an automated system identifying a pattern of user access that triggers a security alert. Without considering that these access patterns are part of a routine system upgrade, the automation restricts user access, causing operational disruptions. This highlights the need for human oversight to understand and contextualize security alerts.

APPROACH

Expert



Tailored decision-making:

Allows for nuanced decisions based on contextual understanding



Expertise reliance:

Leverages human expertise to handle complex and context-dependent scenarios



Flexibility: Provides flexibility to adapt to unique situations and exceptions



Understanding context: Humans can interpret contextual nuances that may be challenging for automation



Time-consuming: Manual processes are often slower and can lag behind the dynamic nature of cloud environments



Inconsistencies: Prone to inconsistencies and errors, especially in large and rapidly changing infrastructures



Scalability challenges: Difficult to scale efficiently in large and complex cloud environments



Resource-intensive: Requires significant human resources for policy creation, enforcement, and updates



Limited agility: Slower response to emerging threats due to the manual nature of the process

**Speed and efficiency:**

Automation enables swift and consistent policy enforcement at scale



Consistency: Ensures policies are uniformly applied across the cloud environment

**Resource optimization:**

Reduces the need for human intervention in routine, repetitive tasks

**Understanding context:**

Humans can interpret contextual nuances that may be challenging for automation



Lack of context: May over-enforce policies without considering the unique context of specific situations



Dependency on rules: Rigid adherence to predefined rules may not accommodate evolving situations



Lack of human intuition: Lacks the nuanced understanding and intuition that humans bring to complex scenarios



Complexity challenges: May struggle to handle complex situations that require human judgment

**Potential disruptions:**

Automated processes may disrupt operations if not carefully calibrated

Keys to Prevention

The most successful prevention strategies blend automation with manual oversight – with a focus on promoting *consistency*, *automation*, and adaptable *processes*.

Consistency

Consistent application of security best practices is essential for prevention. This requires close collaboration between security teams and developers to ensure policies are followed. Security must take care not to simply dictate to DevOps teams. Instead, fostering mutual understanding and trust is key so developers buy into the importance of prevention.

EXAMPLE

Access Control Policies

Inconsistent access control policies can lead to unauthorized access. Ensuring consistent application of access controls, regardless of the resource or environment, reduces the risk of security breaches.

Automation

Automation streamlines policy enforcement, reducing manual effort and ensuring rapid responses to emerging threats. However, this automation is carefully calibrated to avoid over-enforcement, with human oversight providing the necessary context.

EXAMPLE

Patch Management

Automated patch management systems can swiftly apply security updates across the infrastructure. However, blindly applying patches without considering critical operational periods can lead to service disruptions. Human oversight ensures strategic timing for patch implementation.

Process

A well-defined and documented process is essential for successful prevention. Organizations establish a clear process for creating, updating, and enforcing policies, including regular reviews and adjustments to adapt to evolving security challenges.

EXAMPLE

Incident Response

A well-defined incident response process ensures that security incidents are handled systematically. This involves detection, analysis, containment, eradication, recovery, and lessons learned. This process helps prevent the recurrence of similar incidents in the future.

Real-World Example: Configuring AWS Config Rules

To illustrate the prevention stage, let's consider a real-world example involving AWS Config rules – and how an organization might implement a process to prevent the creation of new unencrypted EBS volumes.

Introduction

Unencrypted EBS volumes pose a security risk. Prevention involves systematically ensuring that new volumes are encrypted by default.

Policies in Place

Policies are established to dictate that all new EBS volumes must be encrypted. DevOps trusts these policies as they align with security best practices.

Consistency

The policy is consistently applied across the cloud environment, leaving no room for deviations or misconfigurations.

Automation

AWS Config rules are configured to automatically detect and alert on any new unencrypted EBS volumes, triggering a swift response.

Process

The prevention process includes regular reviews of encryption policies, updates based on emerging threats, and collaboration between security and DevOps teams.

The Bottom Line: Towards a Better Prevention Approach

Preventing recurrence of cloud security issues is essential for reducing organizational risk. Manual policy creation is too slow while automation alone risks blind over-enforcement. The most effective prevention blends consistency, automation, and adaptable processes – amplifying the strengths of both human oversight and software safeguards. With robust prevention in place, organizations can systematically strengthen their cloud security postures over time.

For Practitioners

For security teams, an optimized prevention process provides improved workflow efficiency, response agility, and clarity of responsibilities. By reducing time spent on manual policy creation and clarifying handling procedures, practitioners can focus their efforts on higher-value security initiatives. They can also respond faster to emerging threats thanks to predefined protocols. Streamlined prevention ultimately helps practitioners improve productivity and job satisfaction.

For Management

For leadership, effective prevention delivers tangible improvements in overall security posture through proactive risk reduction. Metrics around policy violations, audit findings, and incident response times will steadily improve. Automated enforcement of compliance requirements also reduces organizational exposure. Leadership gains confidence that both existing and emerging threats will be systematically addressed before impacting the business. With robust prevention practices in place, organizations send a strong message that security is an integral part of their culture.

Remediation For a New Era of Cloud Security

As cloud environments become exponentially more complex, the gap between what organizations *need* from remediation – and what current approaches deliver – is going to continue growing.

Simply put: manual methods don't scale for identifying and addressing critical threats. And purely automated approaches risk unintended impact. What's needed is an approach that brings together AI-powered technology with human expertise and context.

The best path forward lies in amplifying the respective strengths of both automation and human ingenuity – neither can do it alone. With rigorous processes, cross-functional partnerships, and the right balance of machine and manual capabilities, organizations can overcome remediation's multifaceted obstacles.

That's why we introduced Tamnoon's TARP methodology – spanning triage, analysis, resolution, and prevention. TARP provides a four-pillar process for methodical remediation attuned both to nuanced security risks and real-world operational constraints.

By codifying objectives, best practices, and metrics for each phase, TARP enables teams to optimize workflows for business-critical cloud remediation. And by emphasizing tight collaboration between security and DevOps teams, it fosters understanding, cohesion, and continuous improvement across the organization.

As practices mature, remediation transitions from a reactive bottleneck to a proactive capability delivering robust security, streamlined operations, and improved risk posture over time.

Of course, there is no “one size fits all” blueprint. Every company's cloud architecture, processes, and priorities call for a customized remediation approach. But we hope that the framework and examples shared here provide useful guardrails from which you can tailor a methodology aligned to your unique needs.

Cloud environments will only grow more complex from here – and companies can’t afford the outages, breaches, and headaches caused by inadequate remediation. By taking a step back, breaking down silos, and optimizing remediation systematically, companies can address today's threats while building long-term cloud resilience.

How Tamnoon Helps

Powered by AI and curated by CloudPro experts, Tamnoon’s human-centric AI solution connects to your cloud security tools, prioritizes security efforts and focuses your team on the risks that matter. With Tamnoon, SecOps and DevOps teams fix more risks in less time, while limiting the negative impact that configuration changes may cause to their environments. Tamnoon clients see an average 95% reduction in investigation time for alerts and an 85% immediate reduction in critical cloud risks.