WithSecure™ Consulting eBook

Purple teams with wings

Measuring detection efficacy in the cloud

Authors:
Alfie Champion and Nick Jones

Introduction

At its core, a successful purple team will ultimately enable the uplift of an organization’s people, processes, and technologies. When the dust settles, security operations center (SOC) analysts should have a better understanding of offensive tradecraft, how it might surface in their tools, and how they might better scrutinize this behavior. Senior stakeholders should also have an evidence-based view of the organization’s detection capabilities.

For cloud environments, the collaborative nature of a purple team pays even greater dividends, as analysts may be monitoring and developing detections for technologies and environments relatively unfamiliar to them. With cloud workloads playing a critical role for so many organizations, 2 years ago, we decided our purple team exercises needed a cloud migration of their own. And in 2020, we pioneered our first cloud purple team. Fast forward to the present day, and our team members have presented on the topic at iconic conferences globally, created their own cloud attack simulation tooling, and delivered cloud purple team exercises in 5 countries. This eBook, written with enterprise detection teams in mind, describes our learnings and approach to measuring and developing attack detection efficacy in the cloud.

The journey started with our long-term partner, a global bank with a lengthy history of purple teaming with us, who were confident they understood their on-premise detection capability. As an increasing number of their workloads were migrated to AWS, they wanted to ensure their ability to detect and respond to malicious activity was not impaired. The organization desperately needed to understand and strengthen their cloud detection capability. Drawing on their joint expertise, our Cloud and Detection teams joined forces to create the next generation of Attack Detection Capability Assessment (ACDA). Building on the client’s existing standardized deployment template for multiple application teams to host solutions securely in the cloud, our team—led by Nick Jones (Cloud Security Lead) and Alfie Champion (Detection Lead)—wanted to develop tooling and a methodology that would provide detection assurance on the template, so it could be applied to hundreds of applications going forward.

What is a purple team?

WithSecure's Attack Detection Capability Assessment (ADCA) is a highly collaborative purple team exercise performed alongside a client's detection and response personnel. It is designed to provide a definitive assessment of detection capability. While many offensive security assessments tend to be objective led, an ADCA takes things in a different direction. Executing attack techniques from across the kill-chain in a de-chained, atomic fashion, it makes an objective assessment of an organization's detective capability, all the while ensuring full exposure of, and insight into, the offensive tradecraft in use. This provides a unique opportunity for analysts to learn and workshop potential detections in a safe environment. An ADCA covers the people, processes, and technologies that comprise a detection capability by asking:

People

are the security analysts sufficiently skilled and experienced enough to be able to identify and respond to malicious activity?

Process

are the necessary building blocks in place to facilitate the creation of new use cases? Are defenders proactively and methodically hunting for suspicious activity?

Technology

is sufficient telemetry produced to allow malicious activity to be detected? Are the relevant technologies available to turn an indicator of attack (present in a log) into an actionable, high-fidelity alert?

Why test your cloud Detection capability?

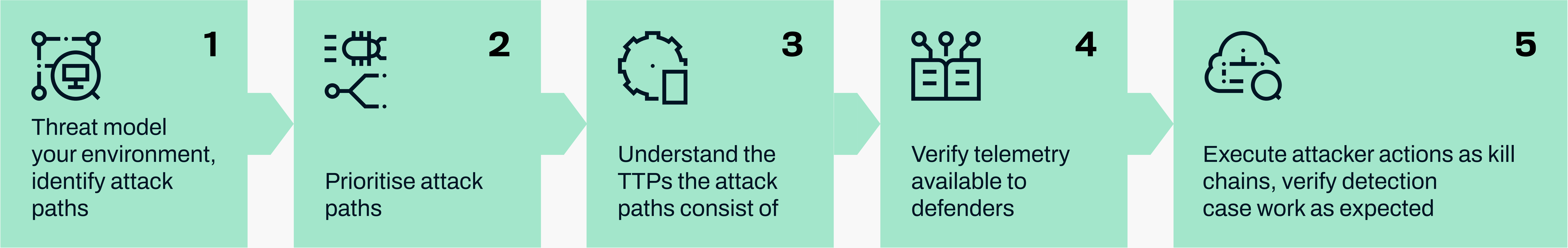
While there is undeniable overlap with on-premise attack detection, the way organizations consume cloud services presents several fundamentally different challenges, particularly when dealing with scalable, ephemeral resources and the multi-account cloud architectures often employed by organizations operating at enterprise scale. Not only are the environments you need to monitor changing all the time, attackers leveraging automation in their attacks shorten the window available to identify, contain, and eradicate the threat. Add the fact that security analysts with a detection skillset are hard to hire and retain (even moreso for cloud) and it's easy to see why maintaining consistent and effective detection capability is such a challenge.

Establishing and operating an effective SOC of any size is, of course, a major undertaking. The initial outlay for detection technologies and the hiring of skilled analysts is just the start, with well defined processes required to ensure continuous improvement and maximize return on investment. An ADCA provides a point-in-time, data-driven means to evaluate the efficacy of all these components. Whether it's identifying telemetry you didn't know you needed, or finding that alerts aren't firing as expected, reviewing your capability in this way provides a means of quantitatively demonstrating measurable improvement over time. With most organizations having multiple cloud technologies and service providers to consider, this approach can also help you determine whether investment in specific security controls is effective, demonstrating either proof of value or providing evidence-based justification to deprecate tooling or forgo further investment.

How can you assess cloud detection capability in the cloud?

Our approach to testing cloud detection includes 5 stages. While there is some degree of commonality with on-premise TTPs, most cloud environments we encounter are purpose-built. Also, while some cloud services are commonplace — IAM to name an obvious example — the data flows and interconnectivity between services are often bespoke.

For this reason, every exercise starts by reviewing the environment's architecture to devise environment-specific attacks to simulate, alongside other known attack techniques. With a clear understanding of what 'bad' looks like in the context of your organization's environments, you can verify the telemetry that you are able to collect, linking it to corresponding identified attack paths and attacker TTPs and developing riskprioritized use cases. Finally, organizations can safely emulate malicious activity in the cloud to regression test their detection capability, checking whether alerts fire and people respond correctly. Once this process is complete, remediation work and further detection engineering can begin. The process can be summarized as follows:



1 Threat model your environment, identify attack paths

2 Prioritise attack paths

3 Understand the TTPs the attack paths consist of

4 Verify telemetry available to defenders

5 Execute attacker actions as kill chains, verify detection case work as expected

Fig. 1. Process for building and validating detection capability for a cloud workload

How cloud detection compares with on-premise

To design an assessment that would provide a robust measure of cloud detection capability, our team began by assessing the core differences between cloud and on-premise detection. In his recent article, Detecting attacks in the cloud, Nick Jones expands upon this topic in detail. 4 key differences between the traditional attack detection approaches and the cloud appeared critical to acknowledge in our approach:

1. **There is more uncertainty around malicious intent in the cloud.** Far fewer actions in the cloud are obviously bad compared to on-premise, making generic detection rules harder to build. Attackers are exploiting known functionality in ways that aren't inherently 'bad'.

2. **Understanding the context of actions is key.** Because far fewer known-bad actions occur, and anomalies will vary by environment, it becomes critical to understand the context of an action. Behavioral analytics are important here, as is developing environment-specific alerting.

3. **At a technology level, gaining visibility is far easier in the cloud.** Organization-wide telemetry sources, like CloudTrail in AWS, make it easier to gain visibility into much of your estate. Shadow IT accounts—for example, accounts opened without central oversight— becomes the primary issue, rather than coverage of known assets.

4. **Cloud attacks can happen very quickly, aided by automation.** The majority of less sophisticated attacks leverage scripted techniques to abuse stolen or exposed credentials for things like cryptocurrency mining, with attacks like this unfolding in moments. The same API-driven control plane we rely on to quickly spin up resources is leveraged to automate targeted attacks in the cloud.

Phase 1: Threat modelling cloud attack paths

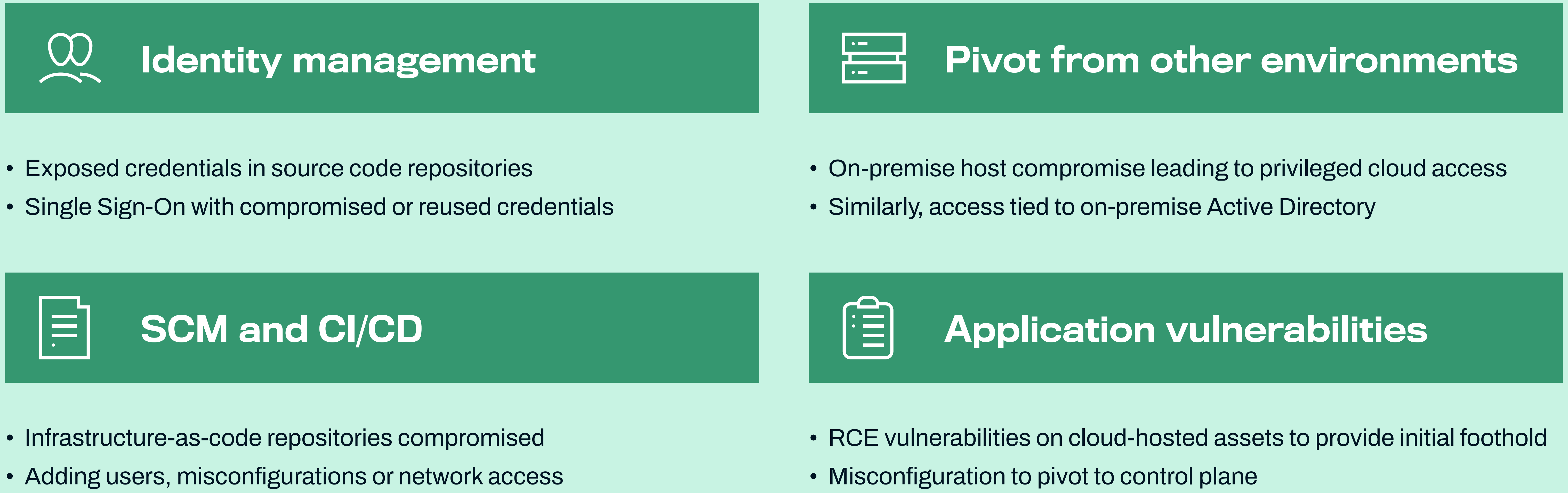
Threat modelling ultimately enables you to identify the most likely tactics, techniques, and procedures (TTPs) attackers would employ in your environment. The first step in this process is to review the environment's architecture. Put simply: what does it do and how does it do it? Does it handle customer data? How does data flow within the environment? We can start to consider what attackers may be looking to achieve within the environment and the TTPs that could be employed to achieve it.

It's important to consider different attacker types too. As discussed above, the context of an action plays such a pivotal role in developing high fidelity use cases. For instance, in many cases we should consider the activities and attack paths of an external unauthenticated attacker, as well as an internal application developer or a system administrator (i.e., an insider threat). There may be several different actors or 'threat agents' to consider, each with their own points of access into the system and assigned privileges.

It's pivotal here to consider the upstream systems and services that could affect the environment in question. For many organizations, the deployment of infrastructure and applications is managed by some form of Continuous Integration and Continuous Deployment (CI/CD). While a meticulously architected environment might be a veritable fortress when considered in isolation, the malicious modification of Infrastructure as Code (IaC) or application code, or indeed a pipeline itself, could have downstream impact on the production environment.

Typical external attack paths we see in the cloud include (fig. 2.):

Fig. 2. Developing and prioritizing your cloud test cases.



Identity management

- Exposed credentials in source code repositories
- Single Sign-On with compromised or reused credentials

Pivot from other environments

- On-premise host compromise leading to privileged cloud access
- Similarly, access tied to on-premise Active Directory

SCM and CI/CD

- Infrastructure-as-code repositories compromised
- Adding users, misconfigurations or network access

Application vulnerabilities

- RCE vulnerabilities on cloud-hosted assets to provide initial foothold
- Misconfiguration to pivot to control plane

Phase 2: Developing and prioritizing your cloud test cases

With attack paths devised, the individual techniques that comprise them can be identified. These individual actions, or test cases, should cover all your relevant cloud services, for example: host-based tests on EC2 instances and modification of users in IAM. Understanding the offensive techniques applicable in the cloud is a common problem, and trusted frameworks like MITRE ATT&CK are still helpful in navigating this. It may also be valuable to talk through 'evil user stories' with development and architecture teams—the people that know the environment best—to ascertain what 'normal' looks like in the context of a user and the environment (see Fig. 3).

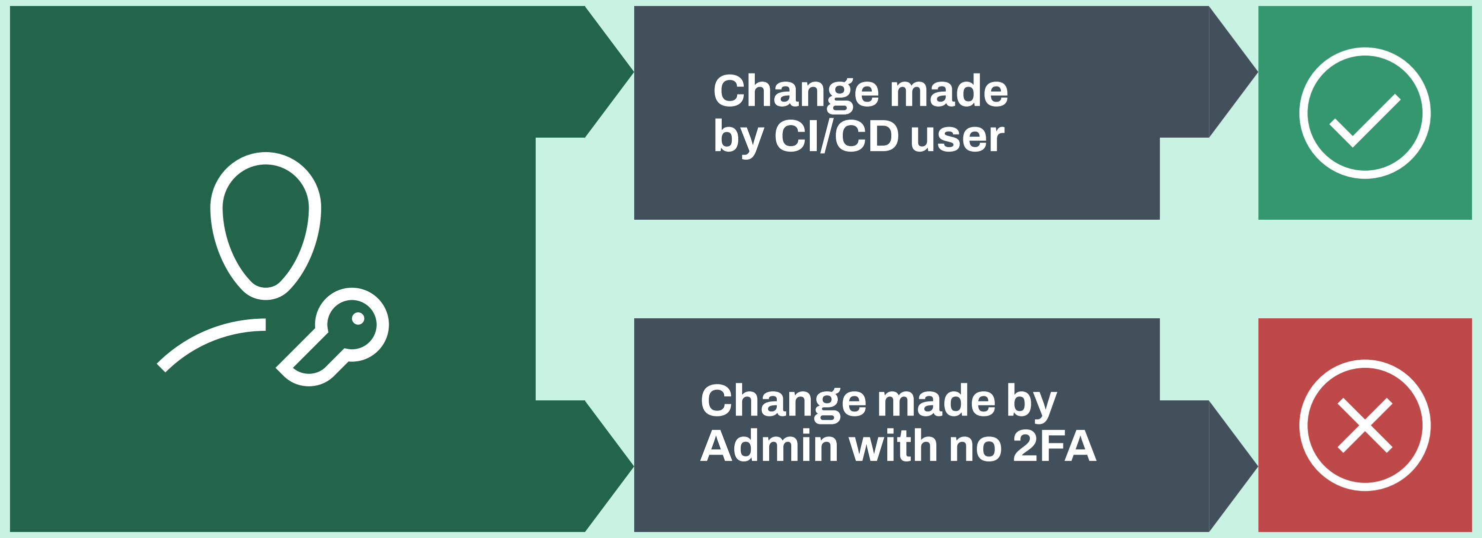


Fig. 3. Context is key. An action performed by one user entity may be benign, while the same action by another may be malicious.

When prioritizing test cases, and by extension use cases, start at the objective end of the killchain and work backwards, mapping the TTPs attackers will use to each identified attack path. That could be the deployment of high-powered virtual machines for crypto mining, or the access and exfiltration of sensitive data from a database. Particularly in enterprise organizations with multi-cloud environments, the sheer complexity and volume of context-based test cases means it's often not practical to test everything. When deciding what to prioritize, consider:

- What is business critical information/ functionality?
- How likely is this type of attack to happen?

Assessments of attack detection capability should prioritize operational resilience, focusing on the highest impact attacker objectives. Attack likelihood is also a sensible factor to consider, as is the difference in attack speed in the cloud (mentioned above). For example, if a certain detection use case could be a choke point for a rapidly unfolding automated attack, and it produces an acceptable level of false positives, you should look to prioritize it. It might even be a candidate for automated response, e.g., the containment of an EC2 instance. Fig. 4. illustrates this pyramid of attacker sophistication, moving from many premade, automated attacks, which are untargeted and leverage public credentials, to human-driven, targeted, and bespoke attacks.

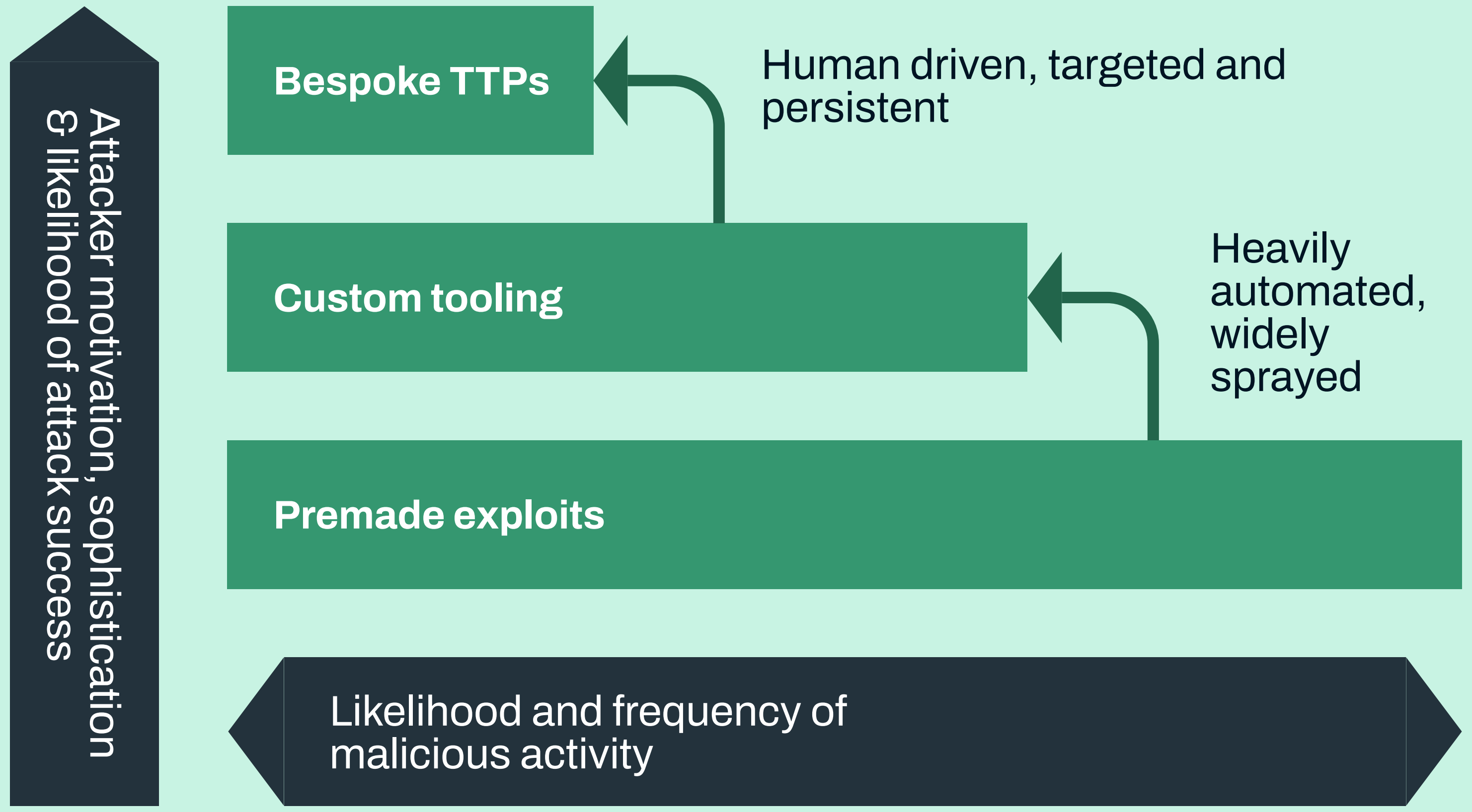

Attacker motivation, sophistication & likelihood of attack success
Bespoke TTPs
Human driven, targeted and persistent
Custom tooling
Heavily automated, widely sprayed
Premade exploits
Likelihood and frequency of malicious activity

Fig. 4. Threat actor characteristics.

Phase 3: Assessing your cloud telemetry

What can we collect logs for? Where should we put them?

Threat modelling and prioritization of attack paths will give you a clear understanding of how your cloud workloads may be targeted at a given point in time. Before you can build or validate any environment-specific alerts, you need telemetry. Therefore, as in any on-premise Purple Team exercise, it's important to conduct a review of your existing cloud telemetry at this stage, identifying which log sources are being ingested, and mapping this to critical assets and probable attack vectors as identified in phase 1.

For defenders ultimately tasked with developing detection and responding to produced alerts, having this telemetry in a centralized security information and event management system (SIEM) is a huge advantage. This is especially true for larger organizations looking to feed logs into a SOC with well-defined processes and technologies. Even more crucially though, defenders must have knowledge of, or access to, the context of the telemetry they're working with. Consider an organization with more than a handful of cloud workloads: what's 'good' in one deployment might be highly suspicious in another. This drive for visibility—looking at all cloud-related telemetry in aggregate — can present problems for developing workload-specific alerts.

All hope is not lost though. A middle ground can be reached in several ways that aren't necessarily mutually exclusive:

- Feed telemetry into a pipeline that correlates and analyzes it against a workload-specific ruleset, forwarding these alerts (either with or without the original telemetry) to the SIEM.
- Enrich the logs with additional metadata that can enable defenders to operationalize logs in the SIEM. Information such as whether the account is a production or development account, what application it's hosting, and if it's internet-facing might be useful.
- Give analysts the toolchain necessary to inspect activity at a per-account level. Consider the first point where solely alerts are forwarded to the SIEM. An analyst should be able to pivot and scrutinize the original logs that triggered an alert. Cloud-native tools such as AWS Athena can provide a means to query logs stored in S3.

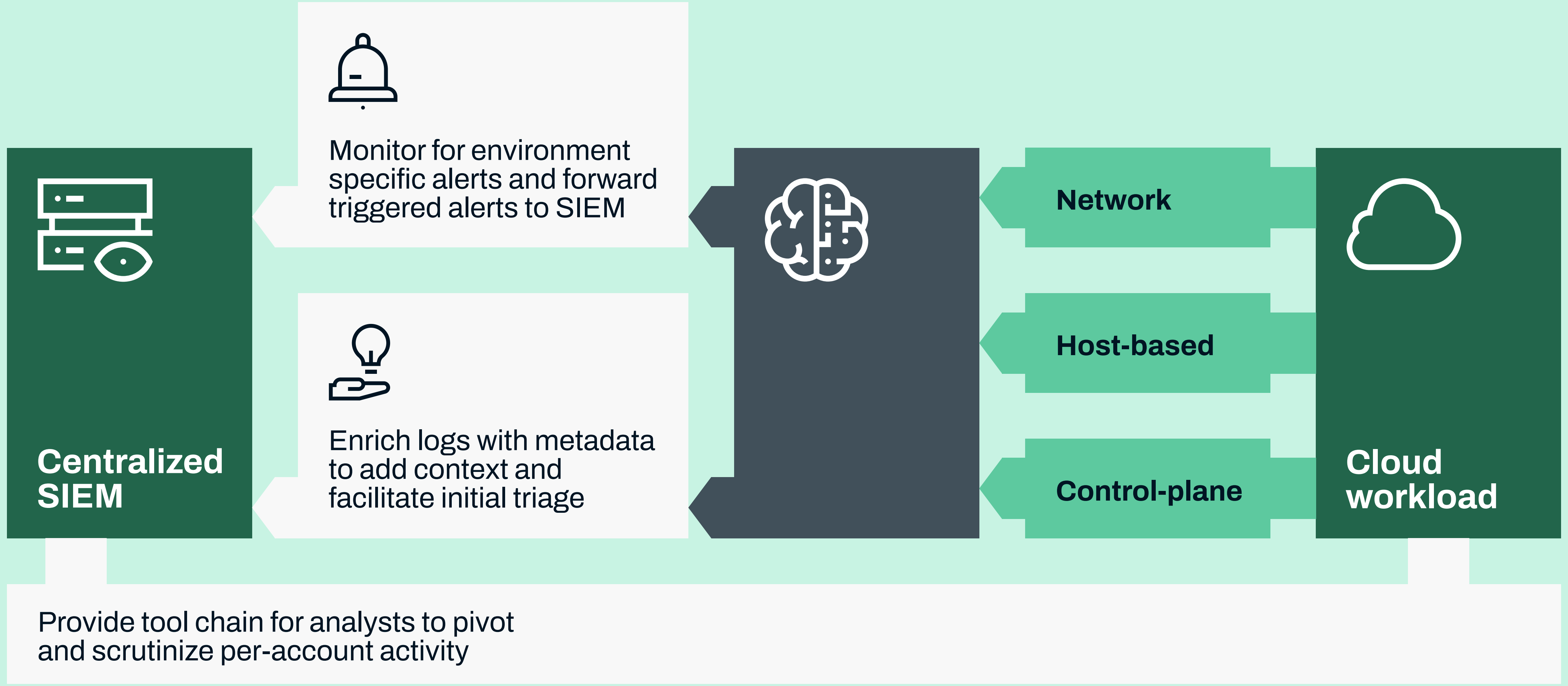

Monitor for environment specific alerts and forward triggered alerts to SIEM
Enrich logs with metadata to add context and facilitate initial triage
Centralized SIEM
Network
Host-based
Control-plane
Cloud workload
Provide tool chain for analysts to pivot and scrutinize per-account activity

Fig. 5. Approaches for scalable detection across cloud accounts, tackling the potential context loss for centralized logging.

In this article, Nick Jones, expands on the key types of cloud telemetry you can look at, summarized into 4 categories:

- Control plane activity logs provide audit logs for user-initiated activity at a control plane level within a cloud environment. These logs typically contain authentication data, access or API keys data, and information about whether the user logged in with multifactor authentication (MFA). Key indicators of credentials being misused will be found here, as will indicators of enumeration activity and attempts to escalate privileges or deploy new resources.
- Network traffic logs can be used to identify malicious traffic and provide visibility into communications between systems within the cloud and to external systems.
- Configuration change logs react to changes made within an environment. Depending on how they are configured, they can even auto-remediate specific high-impact attacker activity.
- Service-specific logs, for example: access logs for S3 buckets or logs showing which users used which keys to decrypt data. Ingesting too much data in this way can be very expensive, so the usefulness of service-specific logs needs to be considered carefully on a case-by-case basis.

Below is a table of some of the most useful telemetry services, organized by provider:

Service type	AWS	Azure	Google Cloud
Control plane logs	CloudTrail	Activity log	Cloud audit logs
Network traffic logs	VPC flow logs	NSG flow logs	VPC flow logs
Configuration change logs	Config	Policy	Cloud’s operations suite
Service-specific logs	S3 and KMS data events	Storage account access logs	Data access audit logs

Evaluating your telemetry will enable you to understand which existing and supplementary sources would best service the proactive detection of malicious activity within the environment. Even for organizations that are at the start of their cloud attack detection journey, with a relatively small number of custom alerts, simulating attacker actions can be a valuable exercise. Firstly, it allows you to exercise the end-to-end process of log capture, ingestion, and analysis. Secondly, it gives analysts an opportunity to see the log artefacts that malicious activity might surface and provides a safe setting to work with the datasets and tools at their disposal. Even for mature SOCs, this process invariably highlights development opportunities in some areas: be that log parsing issues, potential use case generation, or analyst toolset improvements.

Phase 4: Cloud attack detection assessment

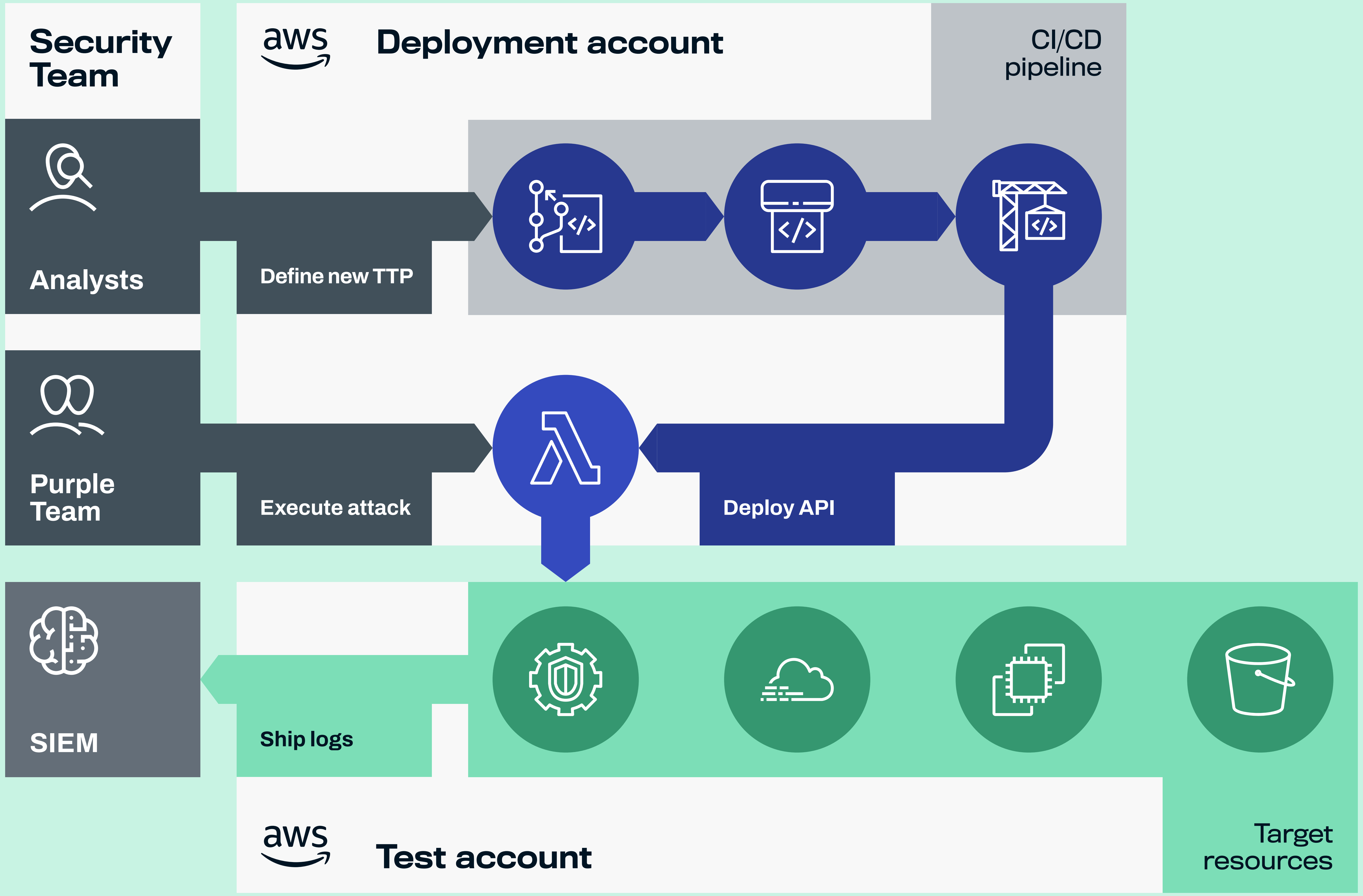


Fig. 6. The high-level interaction between security teams, Leonidas, and a cloud workload under test

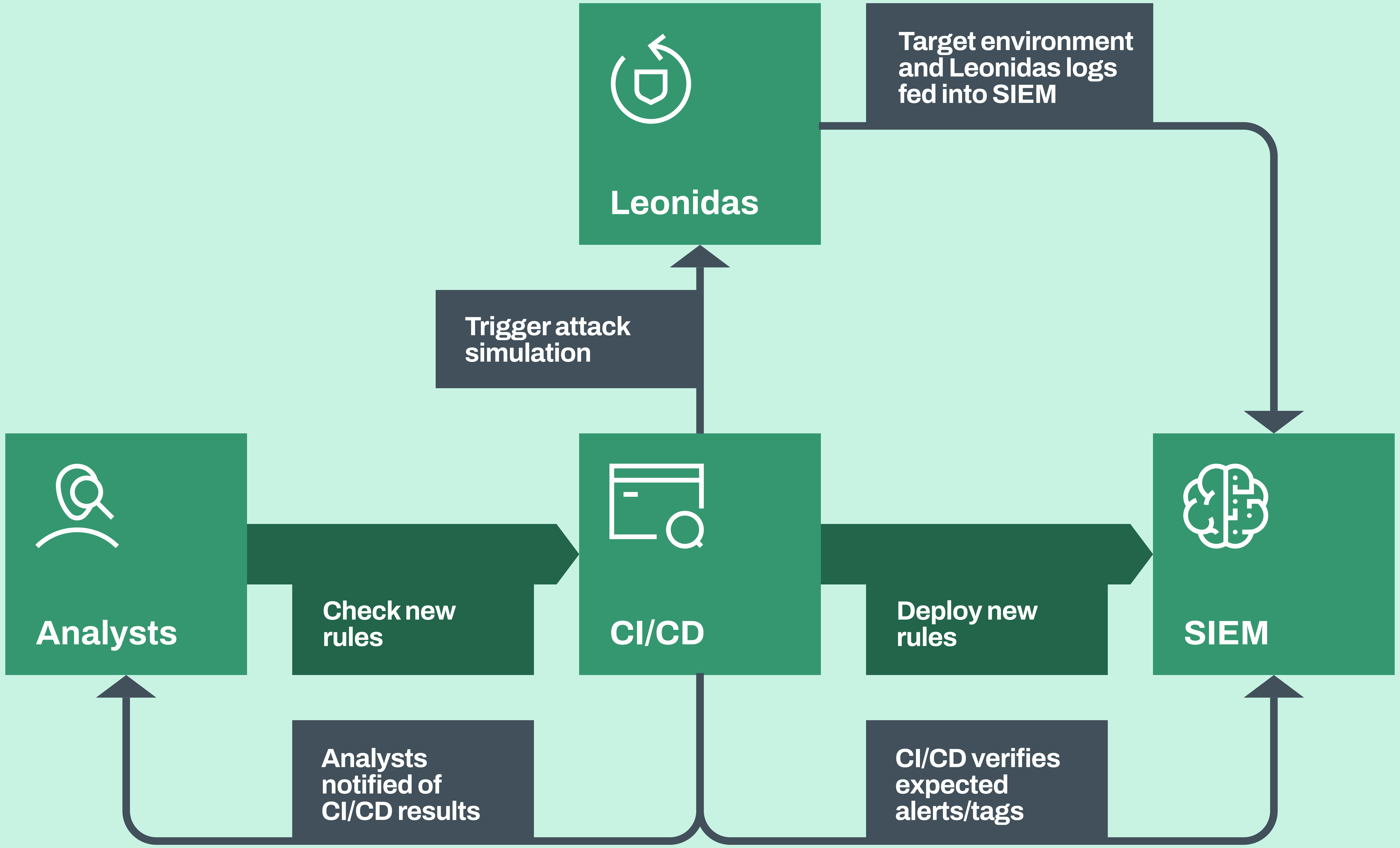
Once you understand how the threat landscape in the cloud is relevant to your organization and to your cloud workloads (attack paths and the TTPs that attackers will use to navigate them), as well as the things you can collect telemetry for, the next stage is attack simulation. Here, you can safely simulate the attacker actions devised and technically validate the degree to which you're able to detect malicious activity within a given cloud environment.

In 2020, our team developed an open-source attack simulation tool, Leonidas, to automate many of these attacker actions. The framework allows purple teamers to increase the number of attacker actions that can be executed within the test timeframe. It also significantly reduces the cost of repeating simulations while developing new detections, in support of a continuous approach to detection improvement. Leonidas was created by our cyber defense specialists with the objective of allowing users to execute attacker actions in the cloud via a serverless web API. This API is deployed by AWSnative continuous integration and delivery tools, allowing rapid development and deployment of new test cases. The API logs and results returned provide the data necessary to easily validate detection telemetry and events against the actions executed. Fig. 6. shows a workflow of the Leonidas tooling.

Phase 5: Implementing a continuous approach to detection validation assessment

For cloud workloads, the cyclical process of detection improvement (fig. 7) is even more pertinent. This set of activities should not be seen as a one-off. As development teams update workloads, new cloud services may be used while others are retired, some log sources may become vital while others are made redundant. In addition, attacker techniques and approaches evolve over time and it is important to stay abreast of developments in attacker tradecraft.

Repeating this process ensures detection capability is maintained and stays relevant to your workloads, and that the complexity of your cloud attack detection end-to-end doesn't lead to unseen regression that only surfaces when you have been compromised. Given that test cases can become so numerous in the cloud, it's often helpful to think of an attack detection assessment as an ongoing process, acknowledging that context-based TTPs are harder to define than on-premises, where you can rely to some extent on detailed open-source threat intelligence about attacker TTPs, which affect different organizations in much the same way. Here's a workflow of what this should look like as a continuous improvement process (fig. 7.):



Leonidas
Target environment and Leonidas logs fed into SIEM
Trigger attack simulation
Analysts
Check new rules
CI/CD
Deploy new rules
SIEM
Analysts notified of CI/CD results
CI/CD verifies expected alerts/tags

Fig. 7. Detailed workflow highlighting Leonidas's application for continuous validation.

In order to support this approach, it is critical that there is minimal cost and friction to simulate attacker TTPs and validate detection use cases. The software development world has embraced automated unit and integration testing as part of CI/CD in order to achieve many of the same goals: drive down the cost of testing and catch bugs sooner. This same strategy can enable a detection engineering team to rapidly develop and validate new detections while ensuring that there are no regressions introduced into previous coverage. It can also be used to provide useful metrics as part of reporting to upper management, in order to demonstrate continual improvement and justify further investment.

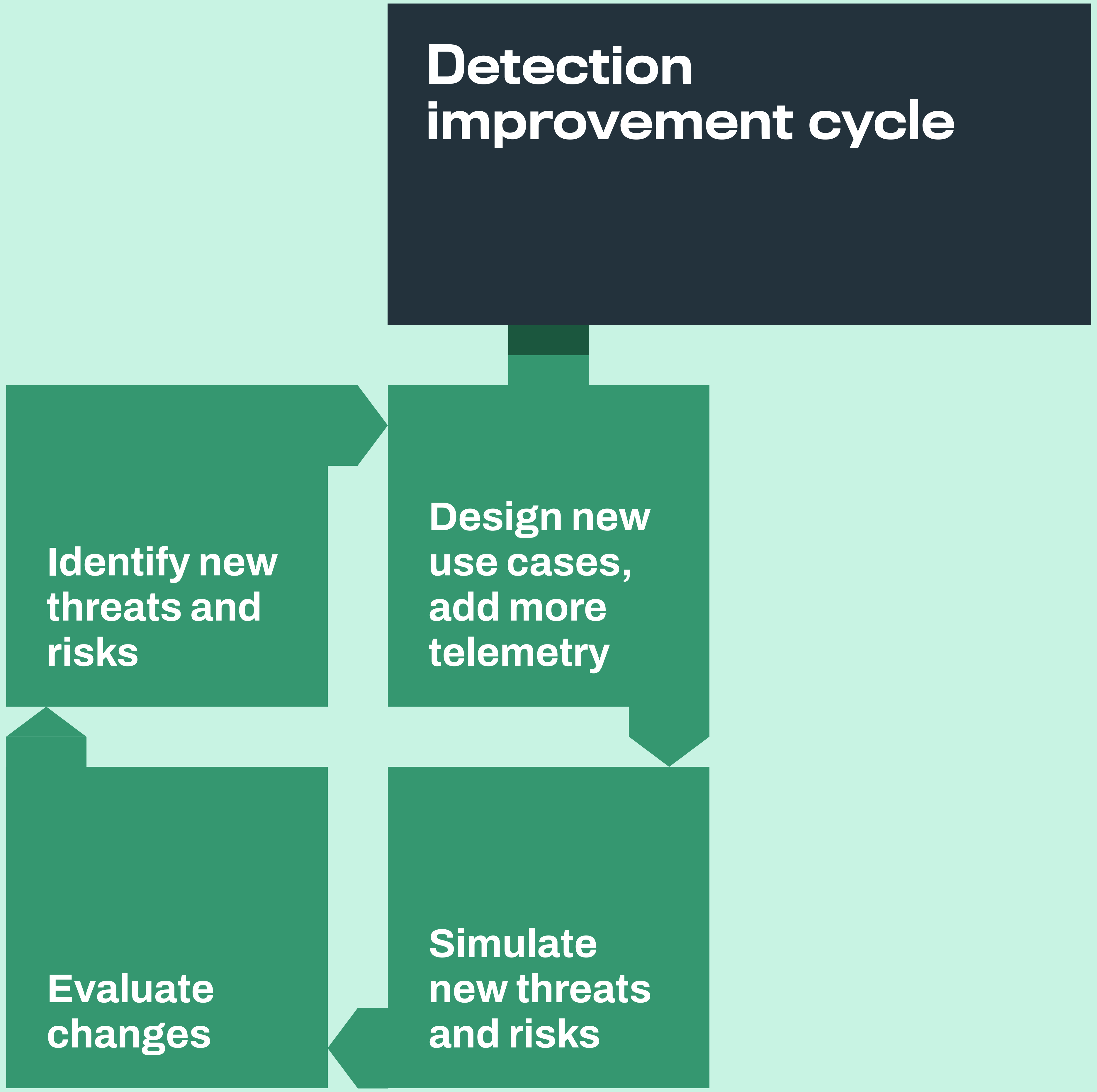

Detection improvement cycle
Identify new threats and risks
Design new use cases, add more telemetry
Simulate new threats and risks
Evaluate changes

Fig. 8. The cyclical process of building detection capability.

Conclusion

The development of cloud attack detection is not a one-time deal, nor as simple as flicking a couple of switches in a portal. For all but the simplest of workloads, the environments you are defending are evolving over time as development teams improve and adapt them to take advantage of the latest technologies and services.

An iterative process of threat modelling and attack simulation can enable detection capability to remain relevant (we don’t need a suite of alerts for Lambda functions that we deprecated in the last release, do we?), as well as ensuring that the telemetry and alerts you rely on remain operational.

Arguably, the greatest benefit of the process illustrated in this eBook is that analysts are given invaluable exposure to what an attack on an unfamiliar technology would look like, as well as highlighting any areas where processes and technologies could be improved to better enable timely investigation and response.

Who We Are

WithSecure™ is cyber security's reliable partner. IT service providers, MSSPs and businesses along with the largest financial institutions, manufacturers, and thousands of the world's most advanced communications and technology providers trust us for outcome-based cyber security that protects and enables their operations. Our AI-driven protection secures endpoints and cloud collaboration, and our intelligent detection & response is powered by experts who identify business risks by proactively hunting for threats and confronting live attacks. Our consultants partner with enterprises and tech challengers to build resilience through evidence-based security advice. With more than 30 years of experience in building technology that meets business objectives, we've built our portfolio to grow with our partners through flexible commercial models.

WithSecure™ is part of F-Secure Corporation, founded in 1988, and listed on the NASDAQ OMX Helsinki Ltd.

W/TH™
secure