

[SOLUTION BRIEF]

Your cloud mapped like an attacker sees it.

Agents inventory every resource across AWS, Google Cloud, Azure and Oracle Cloud, trace real attack paths, and open the Terraform fix as a pull request.

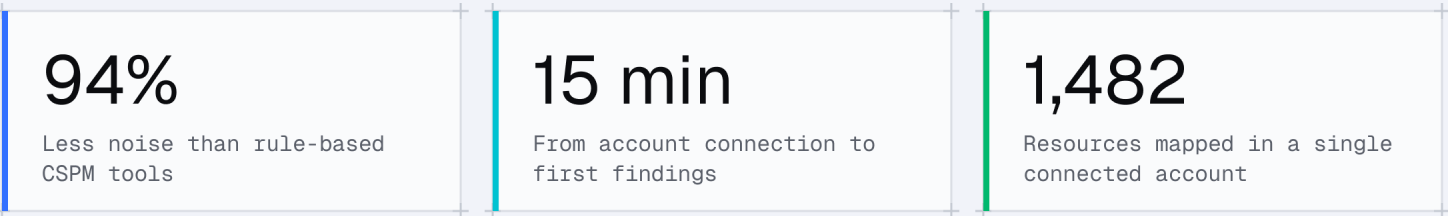
[THE PROBLEM]

Most cloud tools hand you everything and wish you luck. Rule-based CSPM flags every deviation from a benchmark, ranks it by CVSS, and leaves the team to work out which of thousands of findings an attacker could actually chain from the internet to data. The exploitable path is buried in the noise, and the fix arrives as a ticket rather than as code.

WHAT SECURITY TEAMS NEED

- 01 **Attack paths, not rule hits**
Reasoning about how misconfigurations chain together across identity, network and data.
- 02 **Ranked by exploitability**
Findings ordered by actual attacker reach so the top of the queue is what matters.
- 03 **Fix as code**
A Terraform patch opened against your IaC repository, not a screenshot in a ticket.

[BY THE NUMBERS]



BUILT BY THE TEAM THAT SECURED

Apple · Google · Microsoft · U.S. Department of Defense · T-Mobile · Henkel · Epic Games · AT&T · Yahoo

BACKED BY Y COMBINATOR

[HOW IT WORKS]

Inventory. Detect. Remediate as code.

01

Inventory

Connect a read-only role and agents map every resource, identity and trust relationship across your accounts. No agents to install, no performance hit.

02

Detect

Instead of matching rules, agents reason about how misconfigurations chain together, tracing real attack paths from the internet to your data.

03

Remediate

Every finding ships with its proven blast radius and a Terraform patch, opened as a pull request against your infrastructure-as-code.

[SOLUTION ARCHITECTURE]

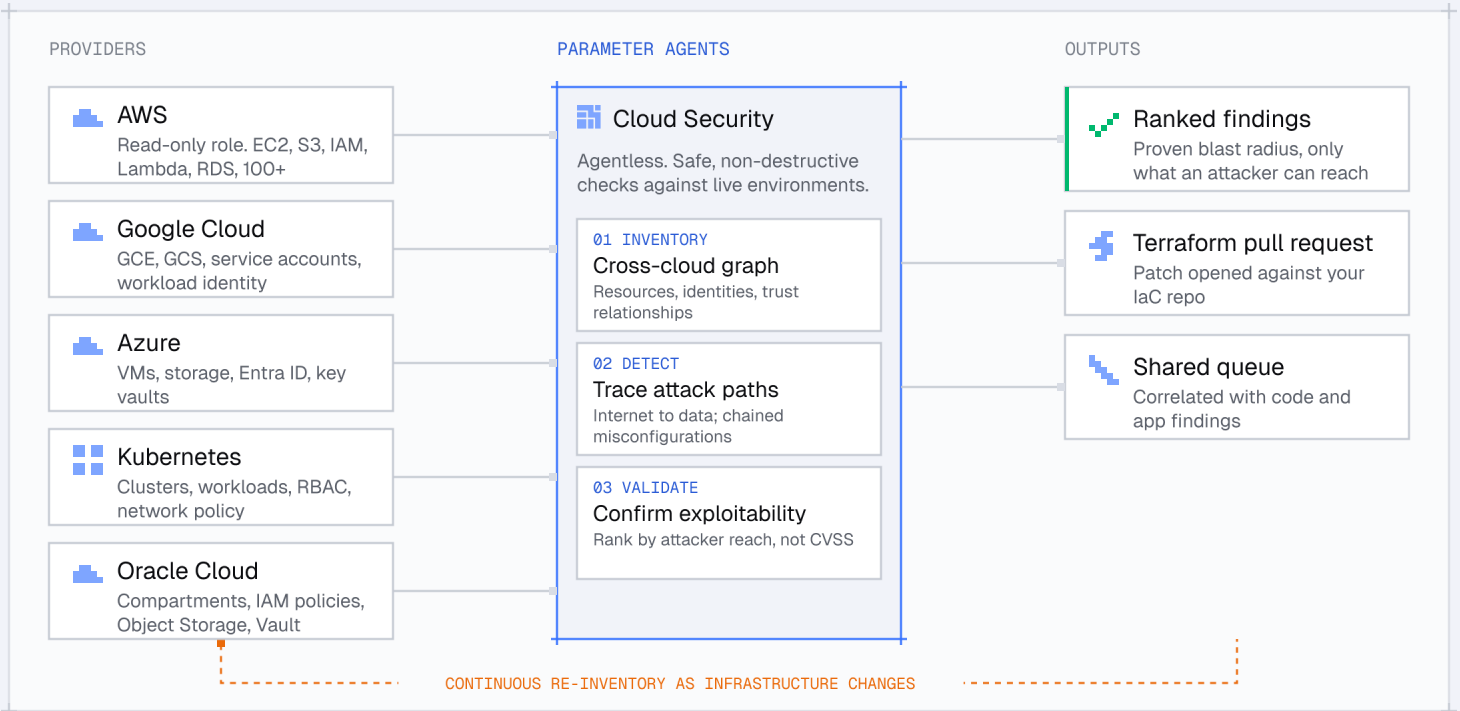


FIG. 01 Cloud Security architecture. Read-only API connections build a cross-cloud graph of resources, identities and trust; agents trace exploitable paths and emit ranked findings and Terraform pull requests.

Agentless in minutes

A secure read-only API connection is all Parameter needs. Checks are safe and non-destructive, validating findings without disrupting live systems.

Signal, not noise

Only what an attacker can reach. Every finding is validated before it reaches you; agents confirm an issue is exploitable rather than flagging theoretical risk.

One platform, one queue

Cloud findings sit alongside code, dependency and pentest findings, correlated to the resource, identity and repository behind them.

Every provider,
every trust relationship.

[PRODUCT]

FIG. 02 The cloud attack-path view: an internet-reachable service chained through an over-permissive role to a data store, with the Terraform fix ready to open as a PR.

[TECHNICAL SPECIFICATIONS]

PROVIDERS	AWS, Google Cloud, Azure, Oracle Cloud, Kubernetes
CONNECTION	Read-only role or API connection; agentless
DETECTION METHOD	Attack-path reasoning across resources, identities and trust relationships
PRIORITIZATION	Ranked by exploitability and attacker reach, not CVSS
VALIDATION	Safe, non-destructive checks confirm exploitability before a finding is raised
REMEDIATION	Terraform patch opened as a pull request against your IaC repository
TIME TO VALUE	First findings in about 15 minutes; most runs complete in hours
ROUTING	Jira, Linear, Slack; shared platform queue

WHAT THE AGENTS LOOK FOR

- **Exposed services**
Internet-reachable endpoints, management ports and public storage that an attacker can enumerate.
- **Misconfigurations that chain**
Individually low-risk settings that combine into a path from the internet to your data.
- **Weak or leaked credentials**
Access keys, tokens and default credentials present in resources, snapshots or metadata.
- **Takeover risk and insecure defaults**
Dangling DNS, orphaned resources and provider defaults that hand control to whoever claims them.

[COVERAGE BY PROVIDER]

AWS

EC2, S3, IAM, Lambda, RDS, 100+ services

GCP

GCE, GCS, service accounts, workload identity

Azure

VMs, storage accounts, Entra ID, key vaults

OCI

Compute, Object Storage, IAM policies, Vault

K8s

Clusters, workloads, RBAC, network policies

IAM

Roles, trust relationships, escalation paths

Data

Public buckets, snapshots, secrets, leaked keys

[WHY PARAMETER]

- ✔ **Attack paths over rules**

Agents reason about how findings chain, which is why the queue is 94% shorter than rule-based CSPM.
- ✔ **Validated before it reaches you**

Non-destructive checks confirm exploitability, so every entry in the queue is real.
- ✔ **Fix as a pull request**

Terraform patches land where infrastructure changes are already reviewed.
- ✔ **One queue with code and app findings**

Correlated with Pentesting Agent, Sentinel and Supply Chain findings on the same platform.

[PART OF THE PARAMETER PLATFORM]

Cloud Security pairs with the Pentesting Agent: an exposed service found in the cloud graph becomes a target the agents test from the outside, and a proven app finding is traced back to the resource that hosts it.

See first findings in 24 hours.

ABOUT PARAMETER

Parameter is an AI-powered security platform that continuously finds and validates vulnerabilities before attackers can exploit them. Autonomous agents pentest like a real adversary across your code, cloud and dependencies, with every finding proven. Formerly Hex Security, Parameter is backed by Y Combinator and built by a team that has secured Apple, Google, Microsoft, the U.S. Department of Defense, T-Mobile, Henkel, Epic Games, AT&T and Yahoo.

TRUST AND COMPLIANCE

SOC 2 Type I (Type II observation underway)
HIPAA BAA available
GDPR DPA with SCCs
Data processed in the US
parameter.ai · support@parameter.ai · app.parameter.ai