Saransh Rana

+91-7905619056 | saransh.rana16@gmail.com | linkedin.com/in/saranshrana | github.com/groovyBugify | burrow.run

Profile

I run security at **Composio**, where AI agents write and deploy production code with no human in the loop. Most of my work is figuring out what that actually requires: sandboxing agent execution, scoping credentials so a prompt injection can't escalate, and moving the AWS estate onto a private network.

I also build **Burrow**, a runtime security platform for AI agents. It runs in production at Composio.

Before this, eight years in cloud and infrastructure security, most of it at **CRED** across 50+ AWS accounts under RBI and PCI DSS. I publish cloud research (GuardDuty runtime-detection bypass, ECS metadata credential exfiltration) and speak at BlackHat and DEFCON.

AWS Community Builder (Security) | CKA | BlackHat Arsenal | DEFCON CloudVillage Lab Instructor

Experience

Head of Security Apr 2026 – Present
Composio *Bangalore*

- **Zen — a production autonomous agent that writes, commits and deploys code with no human in the loop — reached production over public, unauthenticated sandbox and control tunnels.** Co-led its re-architecture onto **AWS Lambda MicroVM**, **ECS** and **NeonDB** behind the private network, with a self-hosted control plane, secrets moved to Secrets Manager + per-task IAM roles, and automated per-session TTL teardown. **Removed the public unauthenticated entry path entirely and contained every agent session to a single-tenant blast radius.**
- **Most code at Composio is now written in AI coding sessions, where instruction files steer agents but enforce nothing — and agents drift from written rules as sessions lengthen.** Designed a three-layer agent security baseline: a canonical `SECURITY.md` contract synced org-wide with `CLAUDE.md/AGENTS.md` pointer blocks; **PreToolUse hooks across Claude Code, Codex and Cursor** that intercept `git commit/push/gh pr create`, run Grype + gitleaks, and hand failures back to the agent to fix and retry; and a CI workflow as deterministic backstop. Shipped the hooks to the fleet as a self-healing **MDM** library item in Audit-and-Enforce mode, so enforcement is not opt-in. **Vulnerable code is stopped on the developer's workstation before it reaches CI, with each branch's contract hash-checked against the canonical repo so it cannot be weakened locally.**
- **Every CVE and cloud misconfiguration had to be read off a scanner report and patched by hand, so the backlog only ever grew.** Wrote autonomous **Claude Code routines** that close findings instead of reporting them — one queries **Socket Security**, patches dependency and image CVEs and opens the PR; another pulls **HIGH CSPM findings**, checks each one against the live cloud to see whether it is actually true, works out what would break if it were fixed, and opens an IaC PR. Nothing mutates without a rollback snapshot (enforced by hook) and one explicit human approval. Standardized on **Grype** org-wide after dropping Trivy when its release pipeline was compromised. **Remediation now runs every day instead of every quarter, and every applied change comes with a one-line revert.**
- **The GitHub org was the softest part of the supply chain: any third-party Action could run at a mutable tag, and repos could be made public by accident.** Required **SHA pinning for all Actions** org-wide, closing the tag-hijack path that had just been used against a major scanner vendor; put **CodeQL** and a **Socket Security** check on protected branches through a single org ruleset with branch-deletion protection; turned on **secret scanning and push protection across 100% of repos**; and made repositories **internal by default** so publishing one is a deliberate act. Extending the same ruleset to an **allow-listed Action registry**, **required review with CODEOWNERS**, and a **Grype** gate on every pull request as it moves from evaluate to enforcing. **A compromised upstream Action can no longer silently change under us, and secrets stop at the push.**
- **Inherited near-zero detection coverage and no controls on what production logs were emitting.** Built Composio's **Datadog Cloud SIEM** — CloudTrail, GitHub audit, Vercel audit and a centralized org-wide VPC Flow Logs pipeline — authored credential-recon detection rules, operated it AI-natively via MCP-driven auditing, and enabled **Sensitive Data Scanner**. **Caught and redacted live AWS presigned-URL credentials, OAuth access/refresh tokens, and JWTs leaking into production logs.**
- **Production AWS was reachable from the public internet, and SaaS admin planes had no fixed egress to allow-list.** Built `cmp-network`, a hub-and-spoke corporate network on Tailscale — **six subnet routers** (prod, zen, staging, integration, security, hub) reaching spoke VPCs over **VPC peering**, chosen over Transit Gateway for non-transitive spoke isolation, on a non-overlapping CIDR plan with each router in an HA autoscaling group and SSM-only access. Added private DNS (`corp.composio.io`) via a Route 53 Resolver inbound endpoint with Tailscale split-DNS, plus a fixed-EIP exit node so **Doppler** could allow-list a single egress address. **Engineers reach production infrastructure over the tailnet; nothing is publicly exposed.**
- **The AWS Org had no preventive guardrails, live root credentials, and a Vercel org that was almost entirely Owner-level.** Shipped a **10-policy SCP suite** across all **8 accounts** — IAM-mutation deny, Secrets Manager lockdown, security-service and SCP tamper protection, region pinning, public-S3 and public-compute-endpoint denial, EC2 shell-access and instance-mutation deny. Removed every root access key and moved the org to **centralized root access management**, restricting `sts:AssumeRoot` by SCP to four named Identity Center principals. Used an **EC2 declarative policy** to close whole vulnerability classes at the account default rather than by detection — IMDSv2 required at hop-limit 1, serial console disabled, public AMI and snapshot sharing blocked. Rebuilt access with Vercel

Enterprise RBAC + Access Groups, IAM Identity Center and GitHub-org RBAC. **Root use, privilege escalation and security-control tampering are denied by policy org-wide rather than caught in review.**

- **A growing Mac fleet had no endpoint detection and no automated response path — covered by one security engineer.** Rolled out **CrowdStrike Falcon** EDR through **Kandji MDM** in Audit-and-Enforce mode with tenant-wide uninstall protection, reconciling MDM and Falcon inventories as the coverage metric, and authored a detection-as-code IOA catalog. Stood up **Falcon Fusion SOAR** with response workflows authored as YAML in a Foundry app; irreversible actions stay human-gated by design. **Detections route and enrich automatically, and the response path is reviewable code rather than console clicks.**

Founder Apr 2026 – Present
Burrow (burrow.run) — runtime security for AI agents *Remote*

- **AI agents execute with inherited human credentials and no runtime policy between intent and action.** Built the platform that closes that gap — **Lattice** (FastAPI control plane, ML detection pipeline, React dashboard), **Warden** (Go PDP/PEP policy engine, deny-wins across tenant→connection→cluster→agent), **Lookout** (forensic investigator), and a **Go CLI** that enforces inline. **Blocks an agent action before it executes — now running in production at Composio.**
- **Every agent framework has its own callback model, so there was no common identity or policy layer.** Shipped Python and TypeScript SDKs with ∼**20 framework adapters** (LangChain, CrewAI, OpenAI Agents, Claude Agent SDK, Google ADK, Pydantic AI, AutoGen, Vercel AI SDK, LiteLLM, LlamaIndex) plus a Go CLI hooking **10 AI coding tools**. **Per-agent identity and policy enforcement across the ecosystem without changing a line of agent code.**
- **Agent threats had no shared taxonomy, so findings could not be triaged like security events.** Built a detection engine over user-defined **CEL rules, chain/sequence detection and per-agent anomaly z-scores** with pre-built rulepacks; engineered multi-tenancy with HMAC-signed inter-service auth, **OAuth 2.1** (Ed25519 JWT + JWKS), request-scoped tenant isolation and pgvector cross-session correlation. **Every finding maps to a MITRE ATLAS technique and OWASP LLM category.**

Head of Infrastructure Security Dec 2025 – Apr 2026
JioStar (Hotstar) *Bengaluru, Karnataka*

- **Owned security architecture for a multi-cloud estate with no unified posture view** — 850+ microservices, 24 EKS clusters across 4 AWS regions, 54 GCP projects. Audited every cluster for RBAC, namespace-isolation and workload-identity gaps. **Discovered a GKE default service account enabling universal impersonation across 72 production nodes**, and designed the NetworkPolicy and Kyverno enforce-mode rollout that closed it.
- Ran an evidence-based IaC threat-modeling audit across 15+ repos — **317 unauthenticated routes, 14 fail-open gateway clusters, 120 security groups open to 0.0.0.0/0, 6+ hardcoded secrets** — converted into a 12-objective remediation roadmap with named owners.
- Built a **GCP Security MCP Server** for autonomous cloud auditing and ran a 6-agent audit with it. **Found 57 findings and 3 compound CRITICAL cross-cloud attack chains — including AWS EKS to full GCP compromise in 8 steps.**

Staff Security Engineer (prev. Senior Security Engineer) Mar 2021 – Dec 2025
CRED *Bengaluru, Karnataka*

- **A regulated fintech scaling past 50 AWS accounts had no repeatable security baseline.** Designed and enforced architecture across all of them via guardrails, SOPs and onboarding playbooks, and drove **55+ infrastructure and compliance audits** (PCI DSS 4.0, RBI, ISO 27001). **Zero critical findings across every audit; raised org security score 72%→99% — highest in APAC — and AWS Well-Architected security 85%→96%.**
- Led threat modeling for **15+ critical platforms** — UPI stack, Lending, Payment Aggregator, PPI Wallets — and designed **110+ preventive cloud security controls**, an EKS Security Baseline, a Common Control Framework and a Day-0 Cloud Security Baseline. **Made the secure configuration the default one for every new account and cluster.**
- **Security-owned M&A for 4 acquisitions** (Happay, Kuvera, CreditVidya) — diligence, integration architecture and **150+ controls** under RBI and PCI DSS v4.0. **Cleared every regulatory gate without blocking deal timelines.**
- **Misconfigurations across 50+ accounts were caught by periodic scans, so a public bucket or an open security group could sit exposed for days.** Built **DIAL**, a detection framework running entirely on API Gateway, EventBridge and Lambda that reacts to CloudTrail events as they land, and **Xenon**, which spots EC2 instance credentials being used from outside the instance they were issued to — the signature of a stolen role. Both alert into Slack with the offending resource and owner. **Dropped mean time to detect from days to under five seconds, and flagged stolen instance credentials faster than GuardDuty's own detection for it.**
- **Turning on IMDSv2 across the estate by hand would have taken months, and until it was done every SSRF bug was a route to live AWS credentials.** Wrote **IMDShift** to find every affected instance across accounts and switch it in place. **Migrated 500+ instances in under five minutes.** Also shipped **TOM** (Tenable-driven patching over Ansible), **Ripple** (blast-radius analysis for a compromised ECS/EKS workload: which task roles it holds, which IAM policies those carry, and what an attacker actually reaches) and **ARKVP** for org-wide resource mapping.
- **Auditing AWS meant knowing which of a hundred API calls to make before you could ask a simple question.** Built **aws-security-mcp**, an MCP server that lets an AI assistant query and inspect AWS security services directly, and drove LLM-based triage into vulnerability management. **Turned cloud security review into**

something you could ask for in a sentence — now the most-used of my open source projects.

- **Container workloads had no runtime visibility.** Led ECS/EKS image-scanning pipelines and runtime protection (Upwind Enterprise), and partnered with the Red Team on adversarial simulations — GitHub PAT abuse, VPN pivoting, metadata-service exploitation, CI/CD privilege escalation. **Findings drove org-wide hardening across GitHub, EDR and cloud.**
- **Published critical cloud research out of that production work — AWS GuardDuty container-runtime detection bypass**, **ECS task-metadata credential exfiltration**, AWS AppStream CloudTrail bypass, AWS CloudShell container breakout, GCP Cloud Workstations privilege escalation. **Featured in AWS Security Digest and TL;DR Sec; presented at BlackHat USA 2023, Ekoparty 2022 and NoNameCon 2021.**

Product Security Engineer — Oct 2019 – Feb 2021
OYO Rooms — *Gurugram, Haryana*

- **Product security across the consumer platform** — ran API and web application penetration testing on auth, inventory, pricing and the consumer portals, and reviewed designs before they shipped rather than after. **Security review became part of the design stage, not a gate at the end.**
- **Owned the bug bounty program end to end** — triage, reproduction, severity calls, researcher response and fix tracking, alongside the Akamai WAF. **Turned a noisy inbox into a triaged pipeline with SLAs the engineering teams actually worked to.**
- **Nobody knew how much of OYO was exposed to the internet.** Built centralized scanners for public endpoints, servers and subdomains (misconfigurations and CVEs), secret-leak detection across GitHub and Bitbucket, a central security-reporting dashboard, and Slack bots for subdomain discovery, scan results and SLA tracking. **Gave the org its first continuous view of its own attack surface.**

Security Engineer I — Jan 2019 – Oct 2019
Signzy — *Bengaluru, Karnataka*

- **Early security hire at a fintech handling KYC and identity data, covering product and compliance at once.** Ran API and web application security testing across the product and built an automated Web/API scanner plus an ML-based S3 upload scanner to catch sensitive documents landing in the wrong place. **Supported the company's first ISO 27001 and PCI DSS audits.**

Open Source

aws-security-mcp | *Python, AWS Boto3, Model Context Protocol — 84⋆* — 2024
MCP server letting AI assistants query AWS security services in natural language. Companion **aws-inspector-mcp** adds LLM-driven vulnerability validation over AWS Inspector.

IMDShift | *Python 3, AWS Boto3* — 2023
Bulk migration of AWS compute to IMDSv2 for SSRF protection — 500+ instances in under five minutes.

DIAL — Did I Alert Lambda? | *Python, API Gateway, EventBridge, Lambda* — 2022
Misconfiguration detection on AWS managed services only, MTTD under five seconds. Paired with **Xenon** for access-key compromise detection.

Research & Publications

- Bypassing AWS GuardDuty Runtime Detection
- Exfiltrating ECS Task Metadata Credentials
- AWS CloudShell Container Breakout
- Bypass AWS CloudTrail Detection via AppStream
- Persistence via Shared Sessions on GCP Cloud Workstations
- Horizontal Privilege Escalation via AWS CLI Aliases
- Model Context Protocol Security
- Tracking Outbound Traffic from AWS NAT Gateways

Featured in *AWS Security Digest* and *TL;DR Sec.*

Speaking & Recognition

- **DEFCON 33** — CloudVillage, Lab Instructor
- **BlackHat USA 2023** — Briefings
- **BlackHat Arsenal USA** — tool demo
- **Ekoparty 2022** **NoNameCon 2021**
- **AWS Summit India** **GotRoot KPMG**
- **CTFs:** AWS Security Gameday, DEFCON CloudVillage, Big IAM Challenge, EKS Cluster Games

Bug bounty acknowledgements: Apple, AWS, GCP, IBM, eBay, Palo Alto Networks, AT&T, Ford, Rockstar Games, OYO.

Technical Skills

- **AI & Agent Security:** agent runtime security, MCP security, prompt-injection / jailbreak detection, Claude Code hooks & plugins, agent routines, Claude Agent SDK, LangChain/LangGraph, Bedrock
- **Cloud & Platform:** AWS (IAM, SCP/RCP, Declarative Policies, IMDSv2, GuardDuty, CloudTrail, EKS, ECS, Lambda MicroVM, IAM Identity Center), GCP (GKE, Secret Manager, SCC), Kubernetes (NetworkPolicy, Kyverno, Workload Identity), Pulumi, Tailscale, VPC peering, Transit Gateway, PrivateLink
- **Detection, Endpoint & Supply Chain:** Datadog Cloud SIEM, CSPM/CIEM, Sensitive Data Scanner, VPC Flow Logs, detection-as-code, CrowdStrike Falcon (IOA/RTR, Fusion SOAR), Kandji MDM, eBPF, Falco, Socket.dev, GHAS (secret scanning, push protection, CodeQL), Grype, gitleaks, Cosign, E2B/Firecracker
- **Identity, Compliance & Languages:** Vault, OAuth 2.1/OIDC/SAML/SCIM, SPIFFE/SPIRE, KMS; PCI DSS v4.0, ISO 27001/27701, SOC 2, GDPR, RBI; Python, Go, TypeScript, Bash. **Education:** B.Tech ECE, Manipal University Jaipur