ANT GROUP SECURES THEIR PLATFORM WITH KATA CONTAINERS AND eBPF FOR FINE GRAINED CONTROL

OVERVIEW

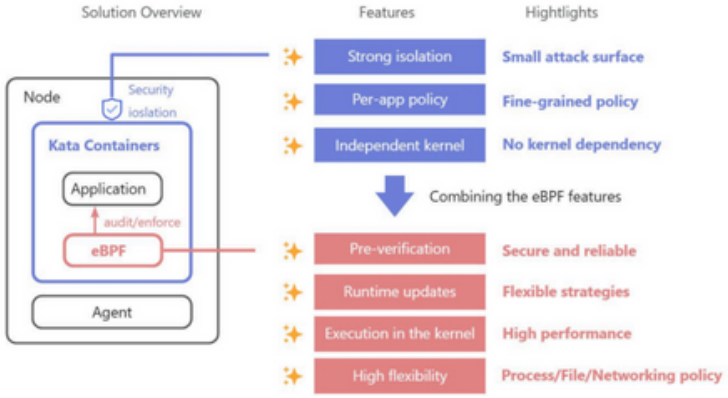
Ant Group needed to strengthen security for its large scale containerized workloads, which power their critical online services and AI applications. Traditional container workload protection solutions left gaps around container escape, inconsistent kernel support, and policy management complexity. To address these challenges, Ant Group developed the Cloud Workload Protection Platform (AntCWPP), combining Kata containers with eBPF. This solution provides strong workload isolation, fine-grained security auditing, and real-time behavioral enforcement. AntCWPP has successfully eliminated the risk of container escape, reduced the blast radius of attacks, and delivered flexible policy enforcement in production environments.

CHALLENGES

As a global technology leader, Ant Group operates diverse containerized workloads across massive production clusters. These workloads faced a growing set of security challenges:

- Container escape and blast radius risks: In traditional runtimes (e.g., runc), containers share the host kernel. This architecture makes it difficult to fully prevent attackers from breaking isolation, and a single compromise can affect multiple workloads.
- Diverse security needs: Internet-facing services, internal applications, and AI agents require different security policies.
- Inconsistent kernel versions: Large-scale production environments contain many kernel versions, making it complex to deploy and maintain consistent security policies across all workloads.

Traditional CWPP solutions built on kernel modules or host-level eBPF introduced operational challenges in these environments, underscoring the need for an approach that could combine strong isolation with flexible, fine-grained enforcement.



SOLUTION

Ant Group built AntCWPP, a next-generation workload protection platform, by combining Kata Containers and eBPF:

- **Kata Containers for Isolation**
- Each Kata container runs inside a lightweight VM with its own dedicated kernel, preventing workloads from impacting each other or the host. This design eliminates container escape and reduces the blast radius of attacks.
- **eBPF for Auditing and Enforcement**
- AntCWPP attaches eBPF programs to Linux Security Module (LSM) hooks and network control points inside the Kata VM kernel. This enables:
  - Process execution monitoring and drift prevention
  - Network activity auditing and fine-grained enforcement
  - File integrity monitoring (FIM) and path-based access control
  - System call monitoring and real-time attack interception
- **Security Agent for Policy Management**
- A node-level security agent delivers policies to Kata Pods, loads eBPF programs into VM kernels, and collects logs through eBPF channels. This architecture decouples security enforcement from the host kernel, enabling flexible and application-specific policies without cluster-wide disruption.

RESULTS

By deploying AntCWPP across high-risk workloads, Ant Group achieved major improvements in security and stability:

- Reduced container escape: Kata's independent kernel significantly reduced escape risks.
- Application-level security control: eBPF-based policies allow precise monitoring and enforcement at the workload level.
- Comprehensive policy coverage: AntCWPP now enforces policies across processes, files, networks, and system calls.
- Production-proven: The solution is running stably in Ant Group's large-scale production environment, securing critical online services and AI agent workloads.

FUTURE PLANS

Ant Group plans to expand its use of eBPF and Kata-based security:

- Broader security policy development: Exploring additional auditing and enforcement capabilities, including advanced protections against emerging threats.
- Extending eBPF to runc containers: Evaluating how eBPF-based policies can be applied in non-Kata environments to provide consistent protection across all container runtimes.
- Industry collaboration: Sharing lessons learned with the open source community to accelerate the adoption of eBPF in secure container environments.

WHY eBPF?

eBPF was chosen because it:

- Provides a safe and stable way to extend the kernel, avoiding risks from kernel modules.
- Offers rich instrumentation points for fine-grained monitoring and control.
- Supports dynamic, runtime updates without system downtime.
- Ensures compatibility and flexibility, even in heterogeneous production clusters when combined with Kata's independent kernel design.