SAQL: A Stream-based Query System for Real-Time Abnormal System Behavior Detection

Peng Gao, Xusheng Xiao, Ding Li, Zhichun Li, Kangkook Jee, Zhenyu Wu, Chung Hwan Kim, Sanjeev R. Kulkarni, Prateek Mittal

Impact of Advanced Persistent Threat (APT) Attacks

APT attacks have plagued many well-protected businesses with significant financial losses.

- Advanced: sophisticated techniques exploiting multiple vulnerabilities
- Persistent: adversaries are continuously observing and stealing data from the target
- Threat: strong economical or political motives

Effectively Detecting APT Attacks in Real-Time

Ubiquitous system monitoring

- Recording system behaviors from kernel as system events (subject, operation, object), e.g., proc p read file f
- Unified structure of audit logs (not bound to applications), presenting a global view of system behaviors

Approach: timely anomaly detection via querying the real-time stream of system monitoring data

Challenges:
- Finding various damaging needles in very large haystacks
- How to incorporate expert knowledge effectively? => expressive and concise domain-specific language
- How to analyze “big data” (~50GB for 100 hosts per day) efficiently? => efficient query execution engine with optimizations tailored to the domain characteristics of the data and the semantics of the query

SAQL Architecture

SAQL: a novel stream-based query system (50K LOC) for real-time abnormal system behavior detection (paper accepted in USENIX Security ’18)

- Agents (built on top of audid, ETW, DTrace) deployed across enterprise hosts to collect critical attributes

Real-World Deployment and Evaluation

We deployed SAQL on 150 hosts of NEC Labs (generating 3750 events/s). We performed realistic attacks in the deployed environment and collected 1.1TB of real system monitoring data for evaluation (containing 3.3 billion system events).

- Evaluation 1: Case study on major attacks (17 SAQL queries): low detection latency (< 2s)
- Evaluation 2: Pressure test on the deployed server (12 cores, 128GB of RAM): high system throughput (~110,000 events/s; supporting ~4000 hosts)
- Evaluation 3: Performance evaluation of the concurrent query scheduler (64 microbenchmark queries): 30% average memory savings than existing systems