RSAConference2016

Singapore | 20-22 July | Marina Bay Sands

Connect to Protect

SESSION ID: CMI1-R09

Eyes Everywhere: Monitoring Today's Borderless Landscape

Bill Shinn

Principal Security Architect
Amazon Web Services
@packet791

#RSAC

What we'll cover today

- Event & Finding Reference Architecture
- Generating Events and Findings – Old and New
- Design Patterns for Collection/Ingestion/Aggregation
- Approaches to Processing
- Analysis and Workflow
- Call to Action

Event & Finding Reference Architecture

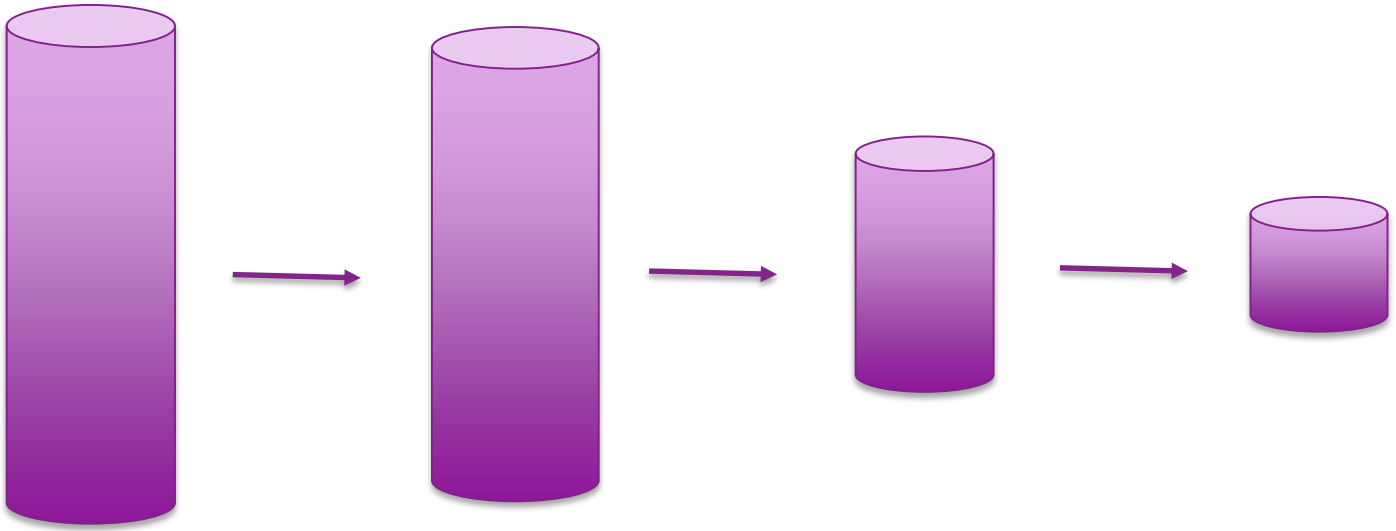
Simple Definitions - Events & Findings

- **Events** - one time record - or series of records that fire - where you can't change the state of what happened.
  - Examples: record of user activity, details of a network flow, parameters of an API call request/response

Simple Definitions - Events & Findings

- **Events** - one time record - or series of records that fire - where you can't change the state of what happened.
  - Examples: record of user activity, details of a network flow, parameters of an API call request/response
- **Finding** - longer-lived information which reflect the state of something that can be changed.
  - Examples: vulnerability scan result, patch state, software defect, build status, threat level, undesireable state of user entitlements

Event & Finding Reference Architecture

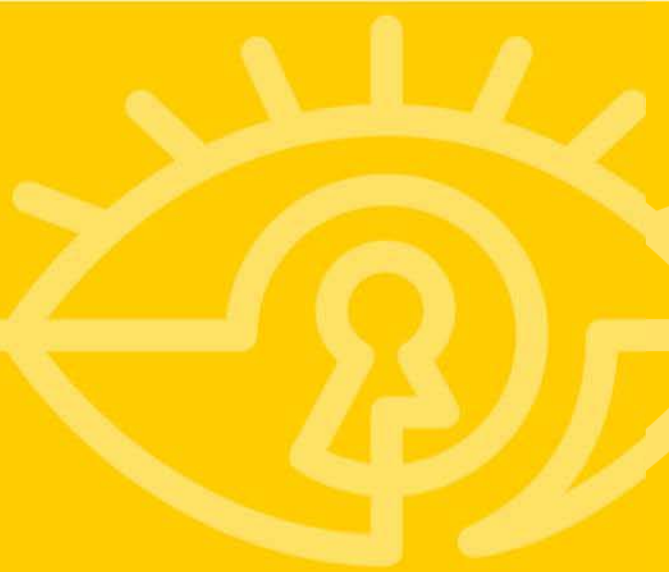


Event & Finding Generation

Event & Finding Collection/Ingestion Aggregation

Event & Finding Processing

Event & Finding Analysis & Workflow



Event & Finding Generation

Event Generation – Traditional Sources

- Source code you write (e.g `log.debug("fooDebug");`)
- Source code you configure (e.g. `log4j.properties`)
- Source code you configure with json, inf, xml files - DEBUG, WARN, INFO, etc.)

Event Generation – Traditional Sources

- Source code you write (e.g `log.debug("fooDebug");`)
- Source code you configure (e.g. `log4j.properties`)
- Source code you configure with json, inf, xml files - DEBUG, WARN, INFO, etc.)
- Operating System log configuration (*nix syslog.conf, Windows Event Log properties)
- Network devices (router/switch log configuration, firewall changes and drop/accept configuration, IDS/IPS signature set/severity configuration)
- Security services (application source code scanner job status, vulnerability scanner job status)

Finding Generation - Traditional Sources

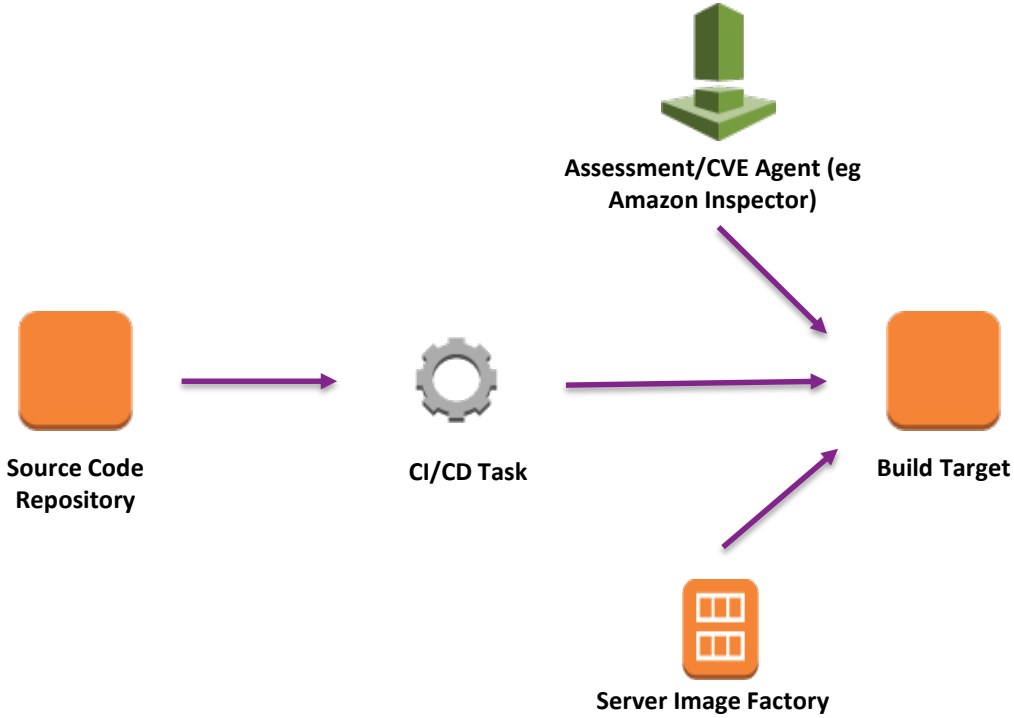
Traditional Sources

- Patch evaluation task results – missing patches
- Vulnerability scanner results – open CVEs
- Unit test results – failed classes/code base + error, failed build
- Application security assessments (software defects)
- Questionnaires - 1000's of vendor due diligence output
- Access/Entitlement review results (elevated privileges, affirm/reapprove)

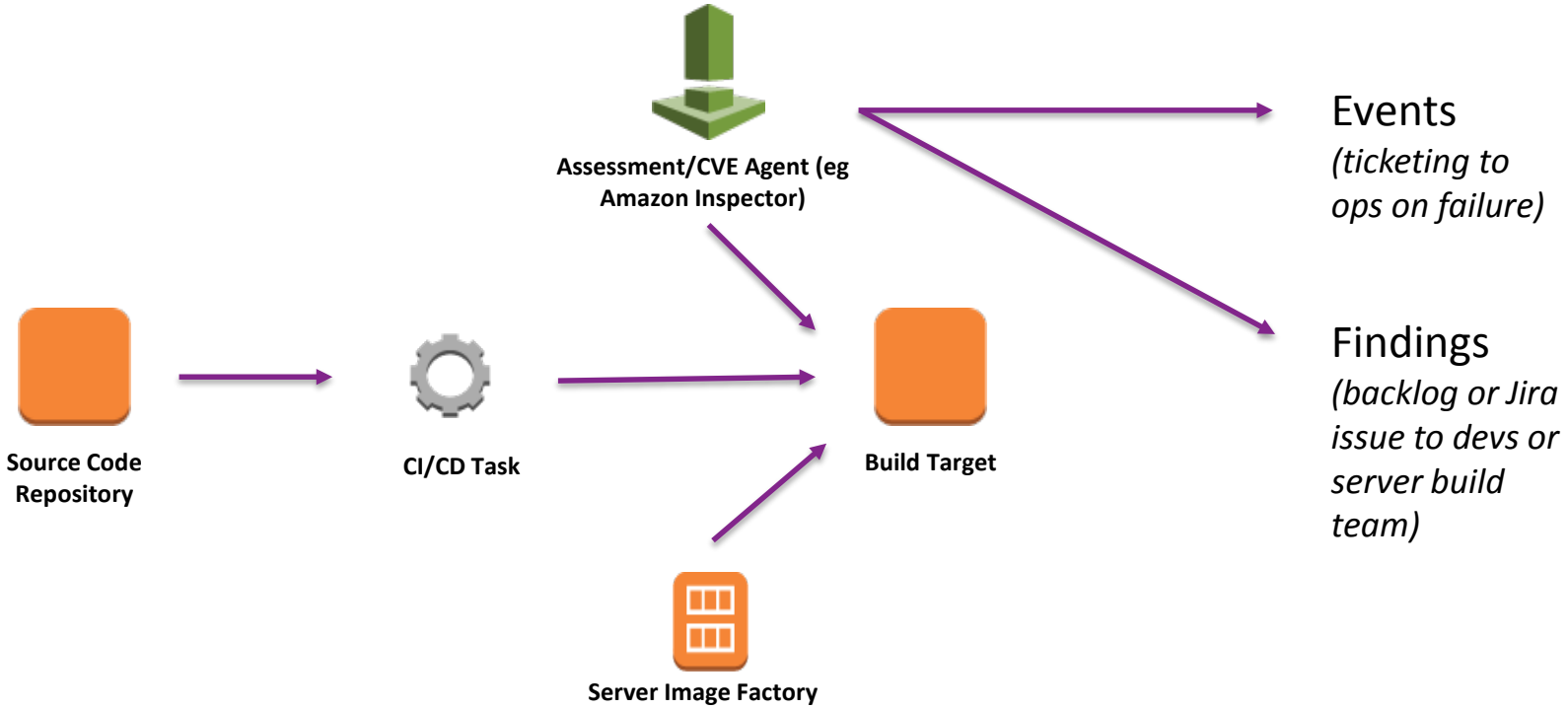
Demo #1

Events vs Findings from a CVE Assessment Tool

Streaming Reference Architecture



Streaming Reference Architecture



Event Generation – Modern Sources

- Cloud platform logs (entitlement management events, infrastructure events such as instance launches or network configuration changes)
- Server-less application logs/streams (cloud-based functions, object storage notification)
- PaaS & managed database services logs (not in a file, but accessible only via API or table)
- SaaS logs delivered via download or API feed

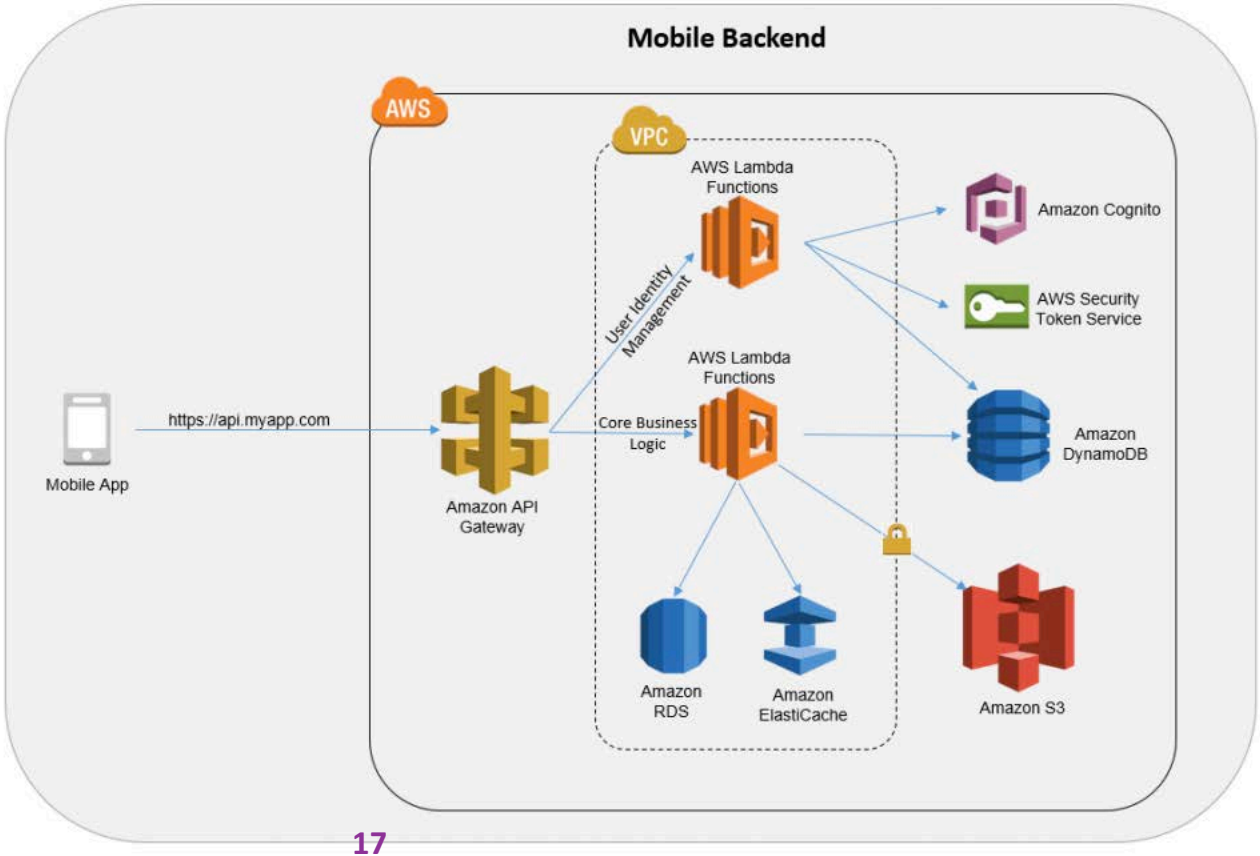
Demo #2

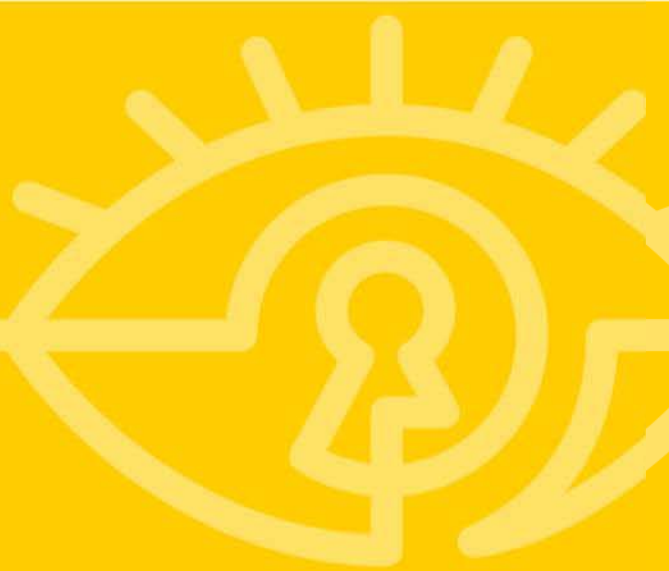
Cloud Service Provider Platform Logging

“Demo” #3

Server-less Architecture Logging

“Demo” #3 (reference architecture)





Event & Finding Collection/Ingestion/Aggregation

Event Collection – Traditional Forms

- Files & remote syslog (ultimately another file, but with some filtering on the way)
- Database tables
- Windows Event Log

Finding Collection – Traditional Forms

- Static reports (console, csv/excel, pdf)

- Findings entered into a database table

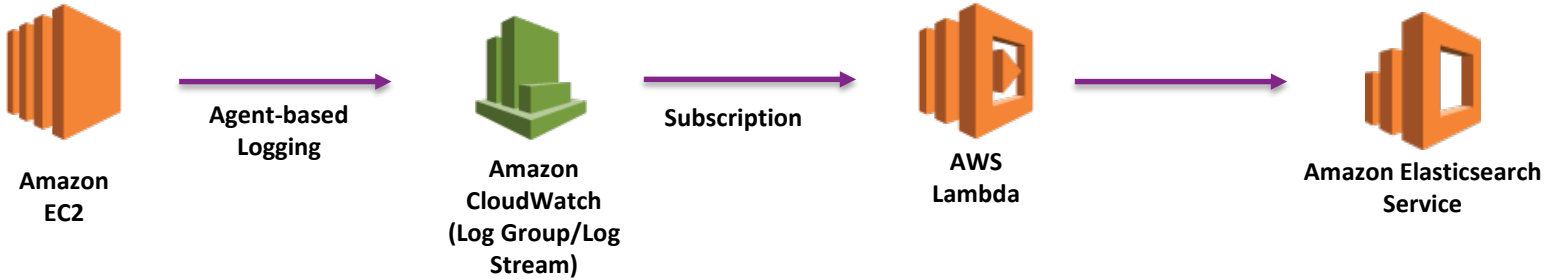
Event & Finding Collection – Modern Forms

- Event Streams
- Findings available via an API

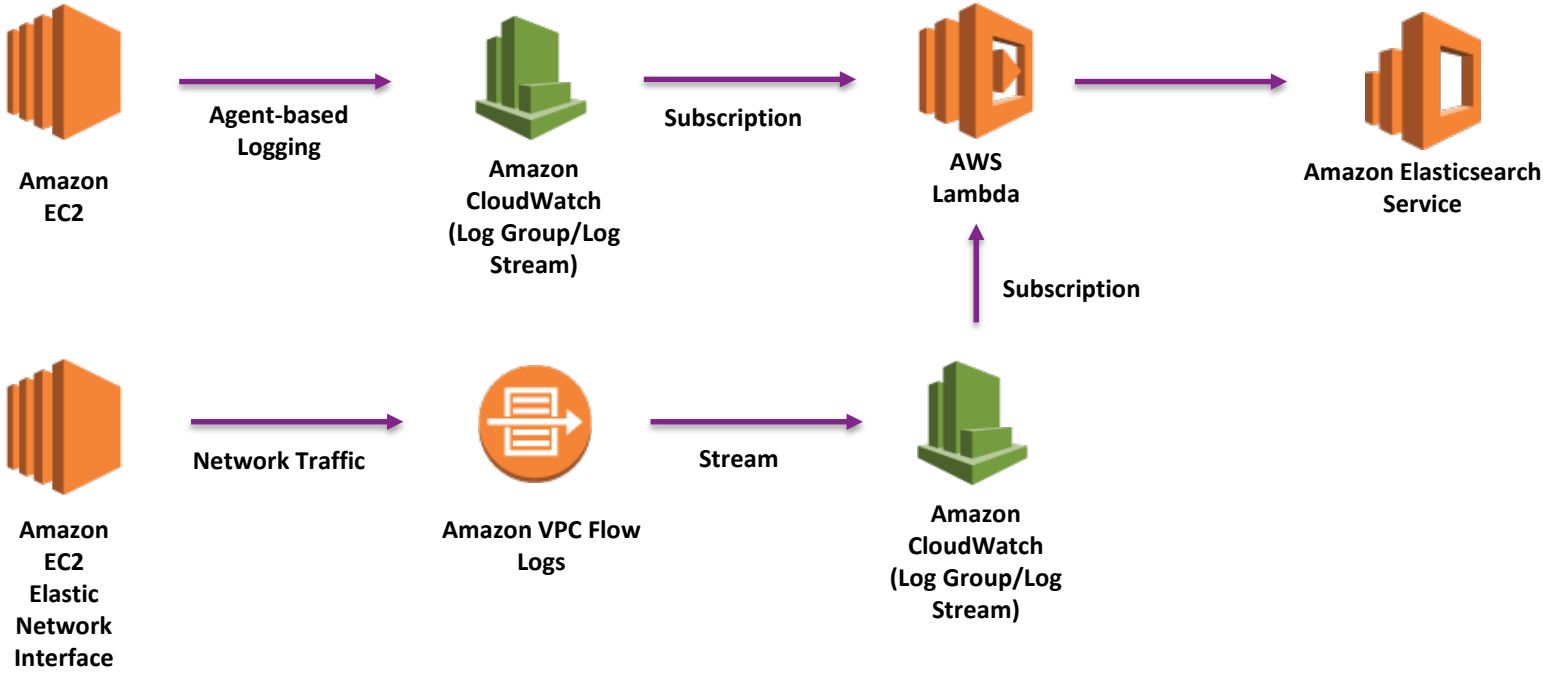
Demo #4

Event Streams

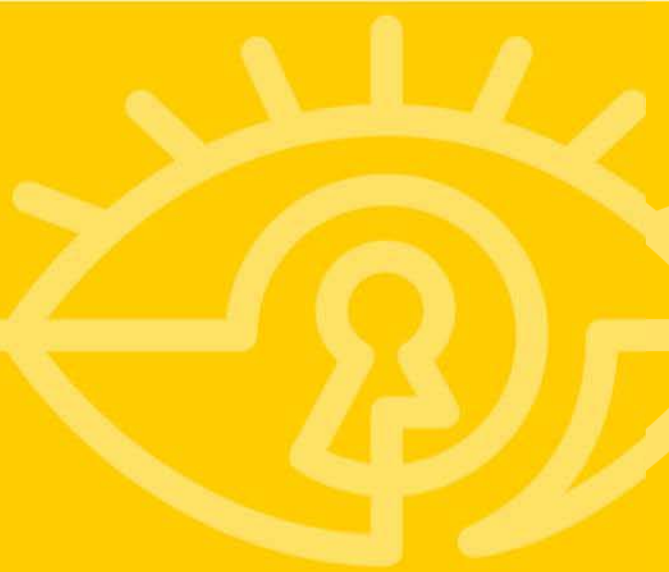
Streaming Reference Architecture



Streaming Reference Architecture



flow logs



Event & Finding Processing

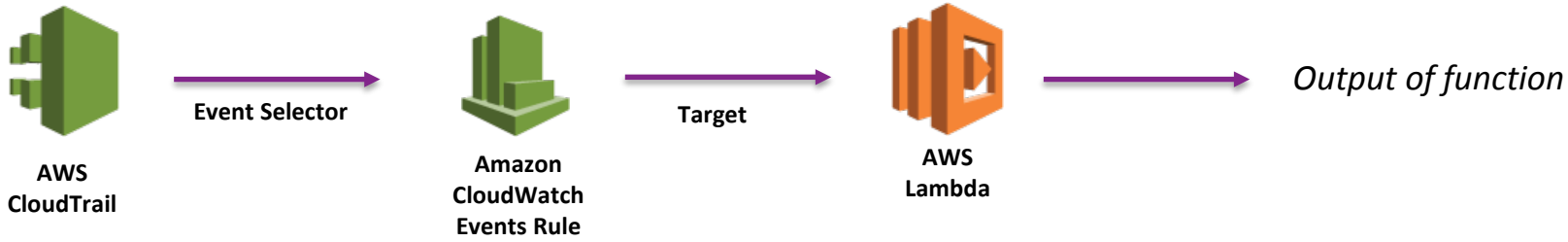
Event & Finding Processing

- Rules-engine application
- MapReduce tasks
- Elasticsearch cluster

Demo #5

Cloud-based rules engine

Rules Engine Reference Architecture



Event & Finding Analysis and Workflow

Event & Finding Analysis

- Commercial log/event search tools and product consoles
- SIEM tools
- Kibana (part of ELK stack)

Analyzing Ephemeral Events & Findings

- What happens when you log something from a server that no longer exists?
- Even more … with serverless architectures, what happens when you log something from a function call that didn't exist as a running set of code loaded into a servlet container or container until it was invoked?

How?

- Track the configuration, not the configuration item
- Correlate events from no-longer-existing assets with platform events

Event & Finding Enrichment

- Meta data on objects
  - Resolver groups
  - Data classification

- Correlation or “joins” on other sources
  - Extract fields and key off critical key/value pairs to perform lookups on related data such as last change, related objects

Event & Finding Workflow

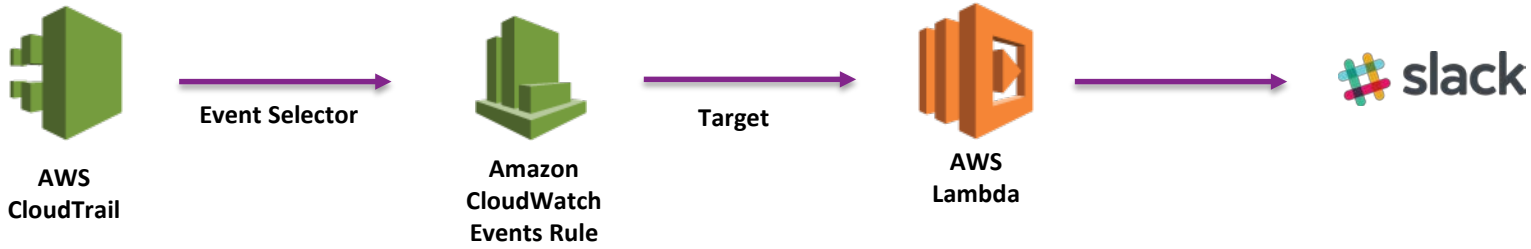
- Ticketing or case management system

- DevOps collaboration platforms (Slack, HipChat, etc.)

Demo #6

Cloud-based event workflow

Cloud-based Workflow Reference Architecture



Applied Architecture – Call to action - 1-2 weeks

- Document event & finding flow for 2 critical applications
  - Keep it to super simple documentation like this and check into revision control:
    - Java properties -> flat file -> file monitor agent -> aggregation microservice -> object storage -> Elasticsearch)
    - Java properties -> flat file -> file monitor agent -> aggregation microservice -> stream processing rules engine-> ticketing API)
- Pair up security analysts with developers to understand log generation and events of interest in 1-2 critical applications

Applied Architecture – Call to action - 3 months

- Stand up a centralized log ingestion platform
- Create micro-services to integrate event/finding processing & analysis tiers with actionable workflow systems