CyberWar
OF THE WORLDS

Hacker Halted

Automated Detection and Response in AWS

October 19, 2020

Speakers

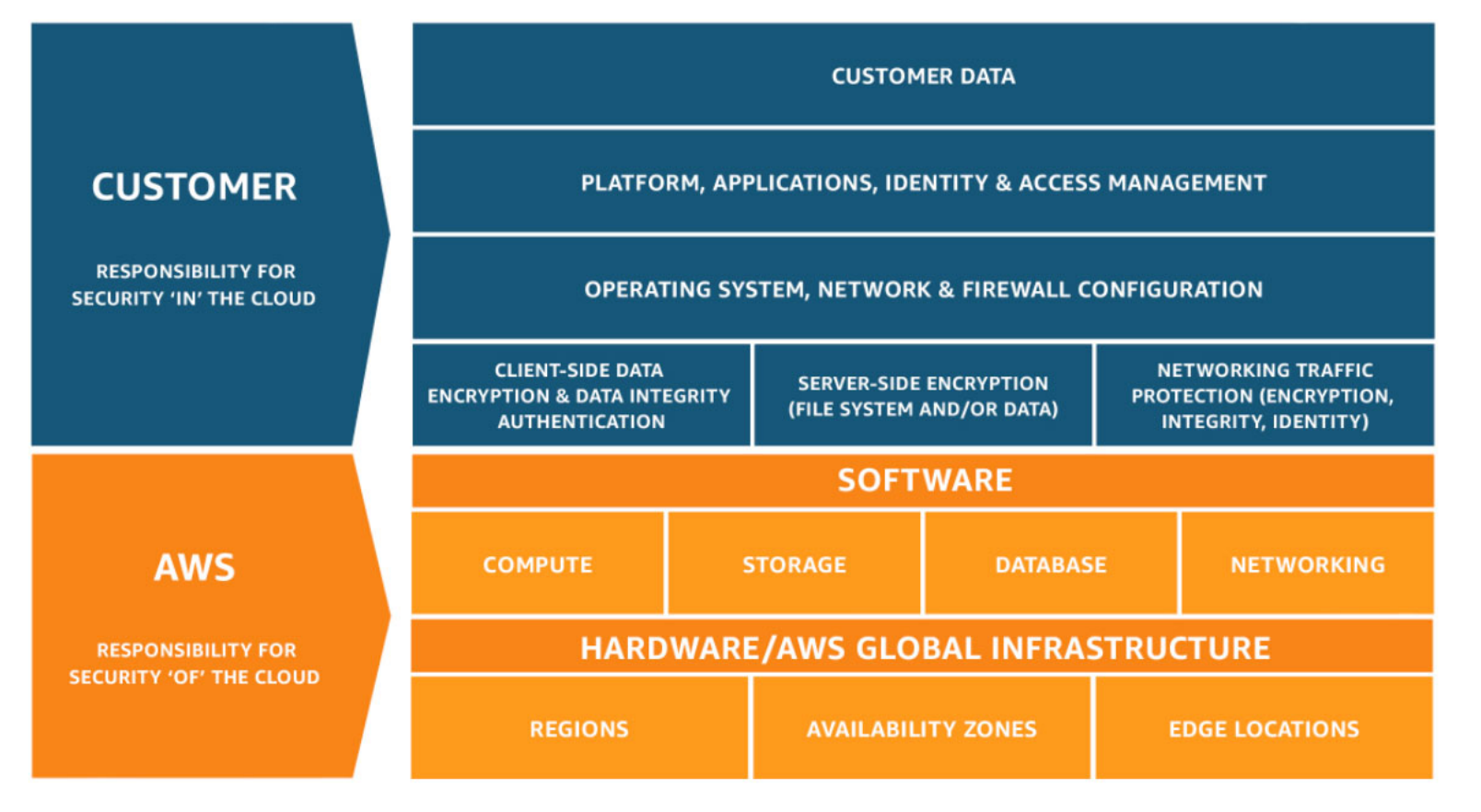
RYAN TICK

VAISHNAV MURTHY

Agenda

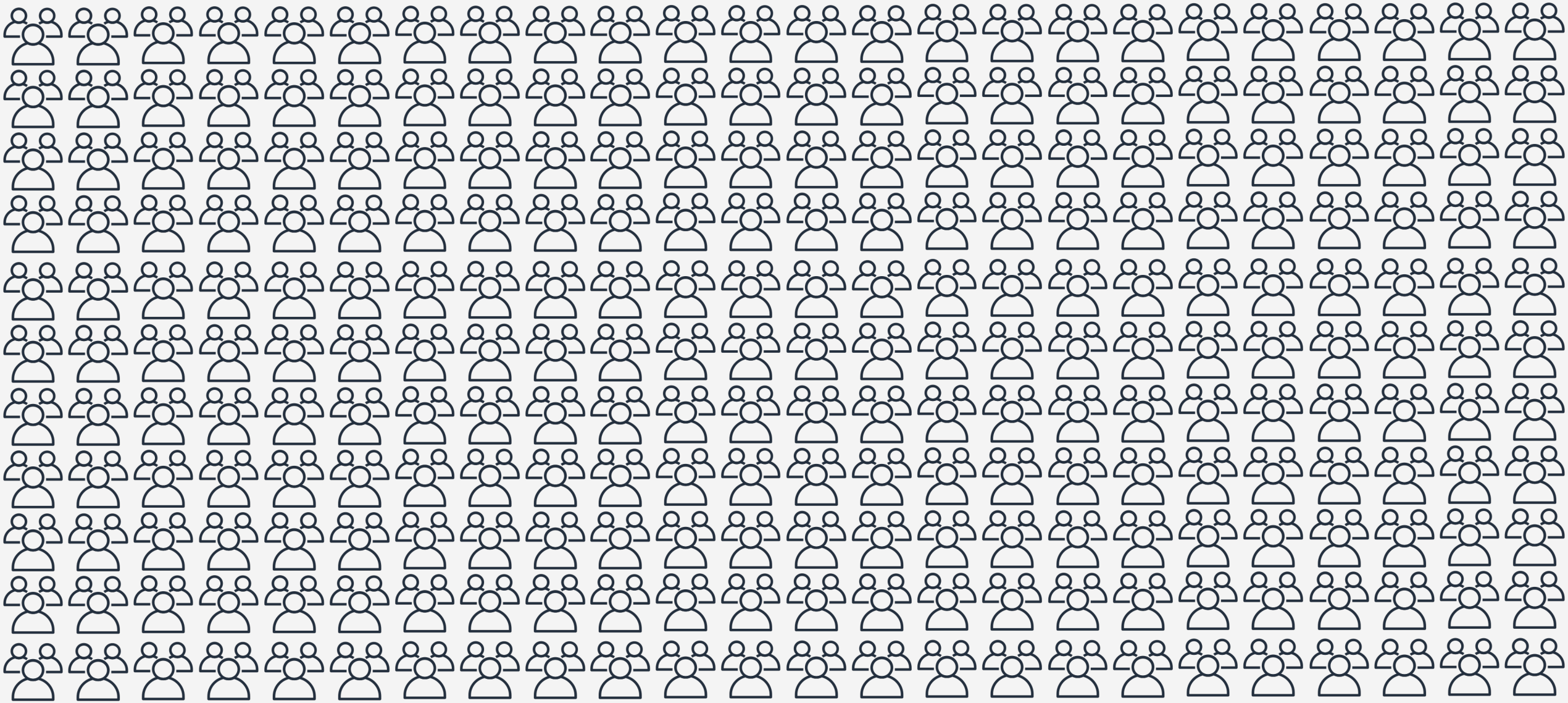
1. Problem Statement
2. Automated Evidence Collection Solution
3. Challenges Faced and Next Steps
4. Getting Started Resources
5. Q&A

Shared Responsibility Model



Problem Statement

Operating at Scale



Multiple Evidence Sources

Amazon CloudWatch

AWS CloudTrail

AWS Config

AWS IAM
Access Analyzer

AWS Security Hub

Amazon VPC
Flow logs

Amazon GuardDuty

Amazon Inspector

Amazon Macie

Antivirus

Anti-malware

IDS

IPS

EDR

Timely and Expensive

MTTD > 6 Months

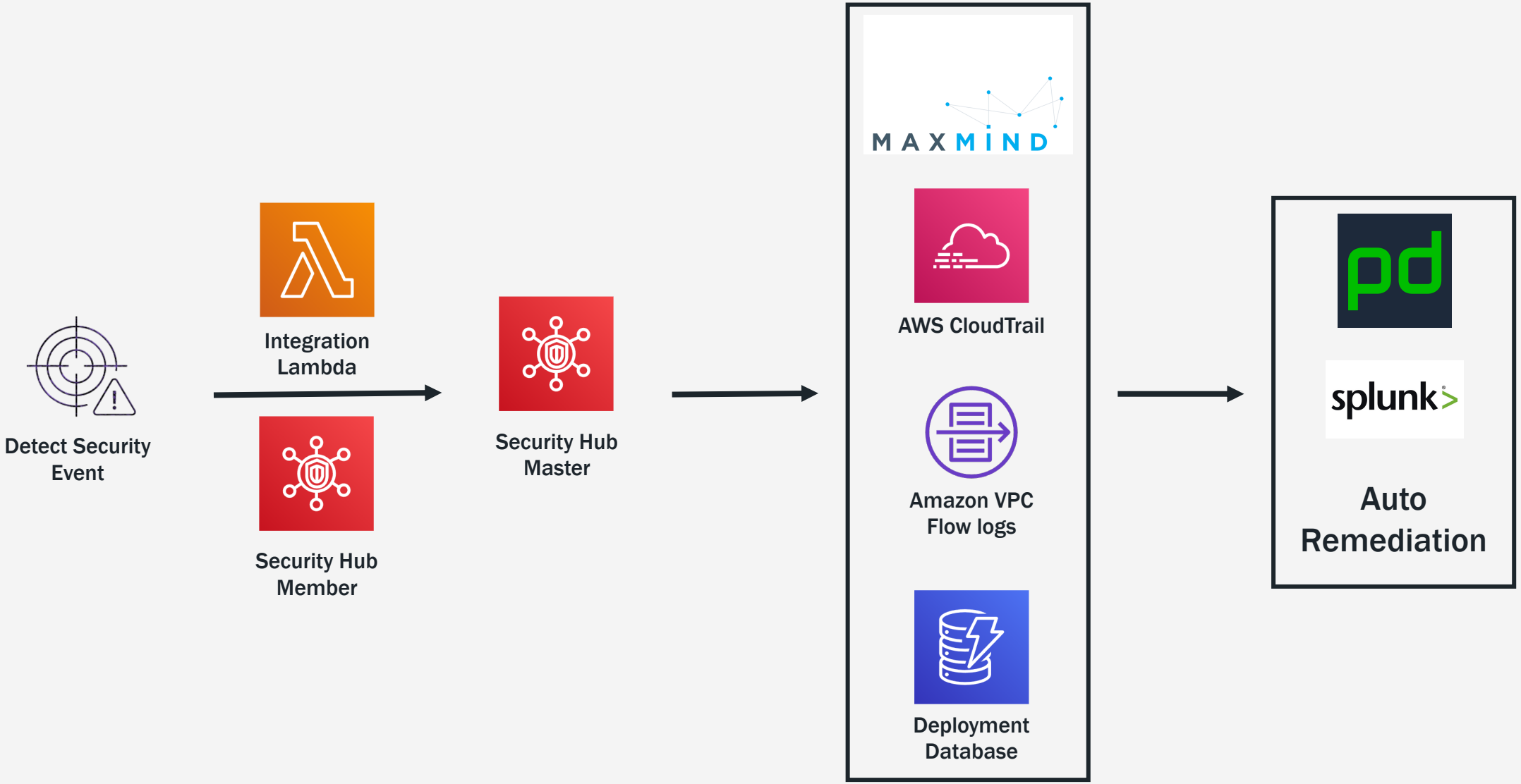
Yearly Spend > $10M

Sky Watch

What is Sky Watch?

- Detect: Identify potentially malicious events and inserts them as a finding into Security Hub
- Enrich: Once data is inserted into Security Hub, enrich it with contextual data (registry data, CloudTrail, VPC flow, etc.)
- Auto-remediate: If we can confidently say a change was bad/malicious, then auto-remediate in real-time.
  - This is where we can trigger the evidence collection workflow if necessary.

Architecture



Example Attack Scenario

MITRE ATT&CK – Cloud Matrix

Discover	Collection	Exfiltration	Impact
• What information on the instance did the attacker gather before installing the Bitcoin miner?	• Were there any large zip files / archives created on the EC2 instance after the Bitcoin miner was installed?	• Were there any network connections established to a potential C2 / traffic redirector?	• How much computing power did the crytominer consume (memory, CPU, GPU, burst credits, etc.)?
• Were there any attempts to reach out to other EC2 instances?	• Were any privileged / sensitive files opened?	• Were there any network connections established to an anonymized file sharing service?	• Did the resource consumption crash / negatively impact other legitimate processes?
• Did the attacker try to list other cloud resources of interest (S3 buckets, EFS shares etc.)?			• Were other EC2 instances spun up to mine cryptocurrency?

Sky Collect

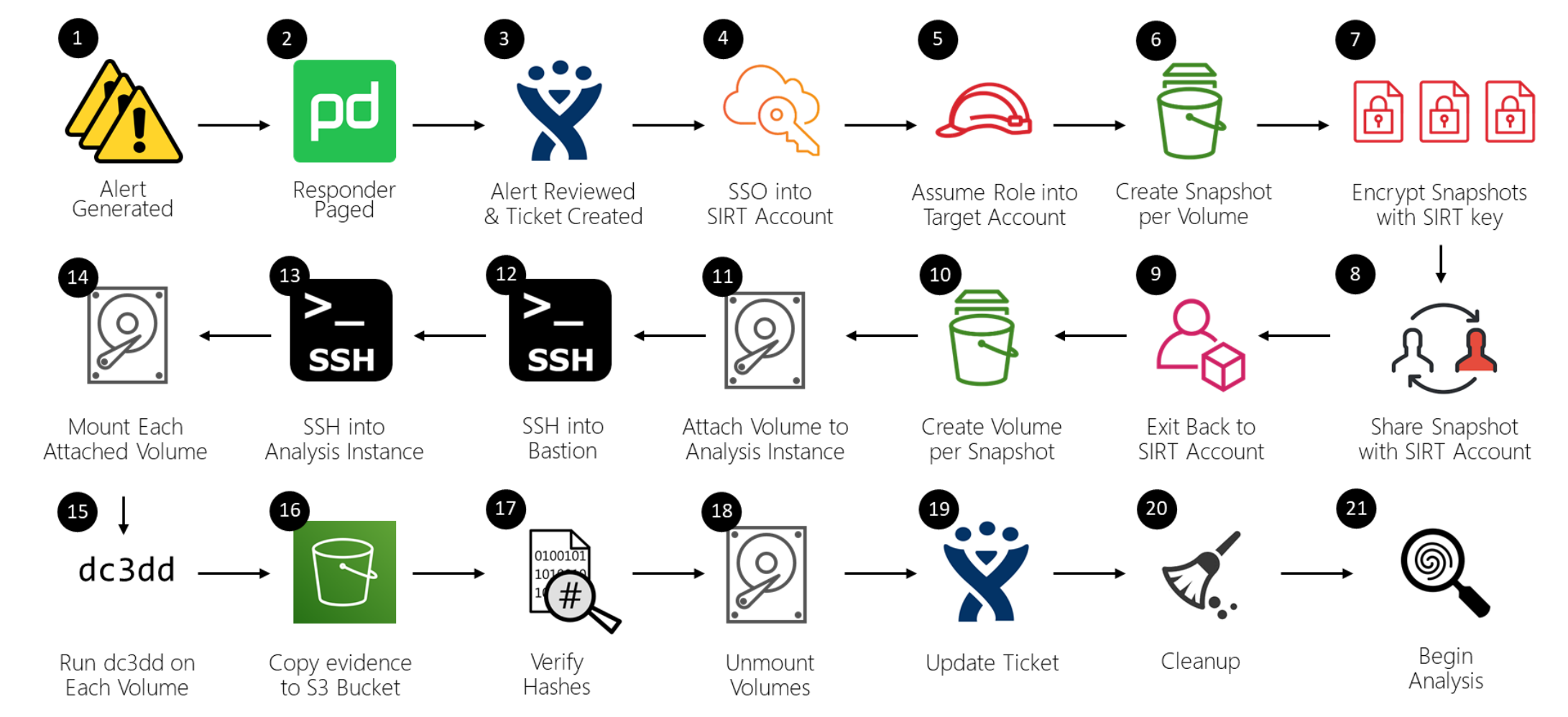
What is Sky Collect?

- Disk collection workflow
  - Raw dd image per volume
  - dc3dd log files
  - Custom collection log files
  - AWS CloudWatch log group
- Memory collection workflow
  - Full memory capture
  - If Linux, custom memory profile
  - Collection log files

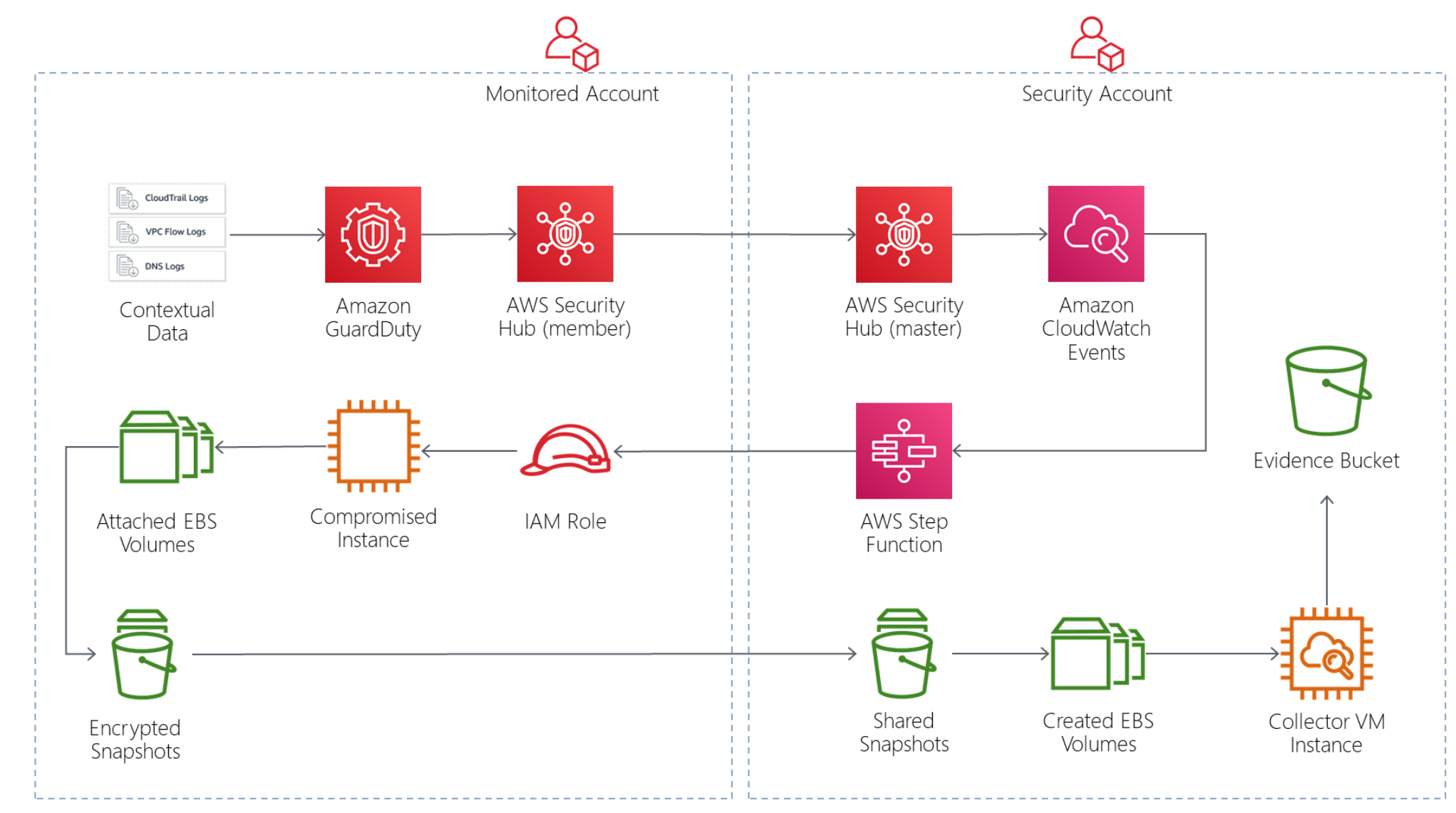
Sky Collect

DISK WORKFLOW

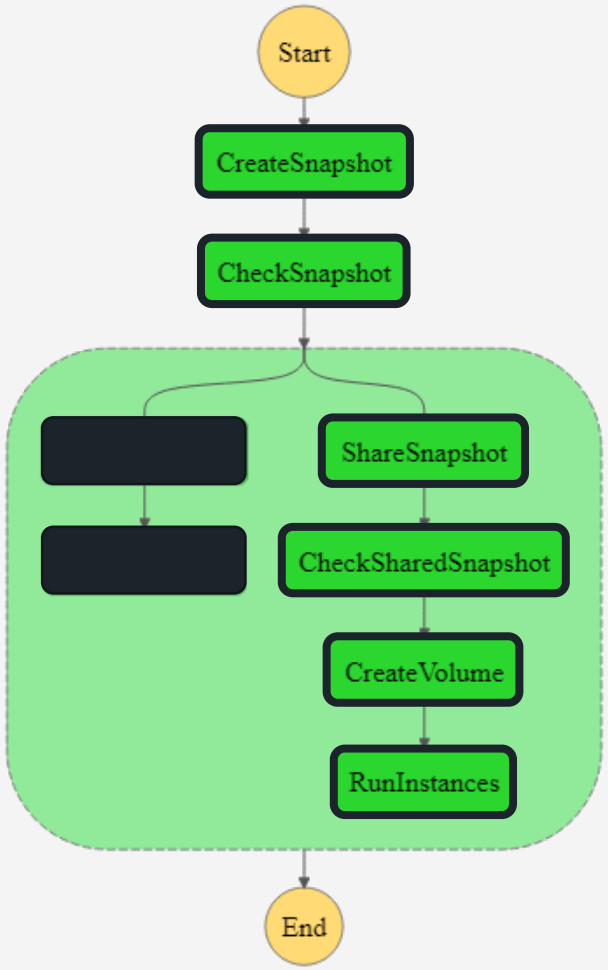
Before Automation



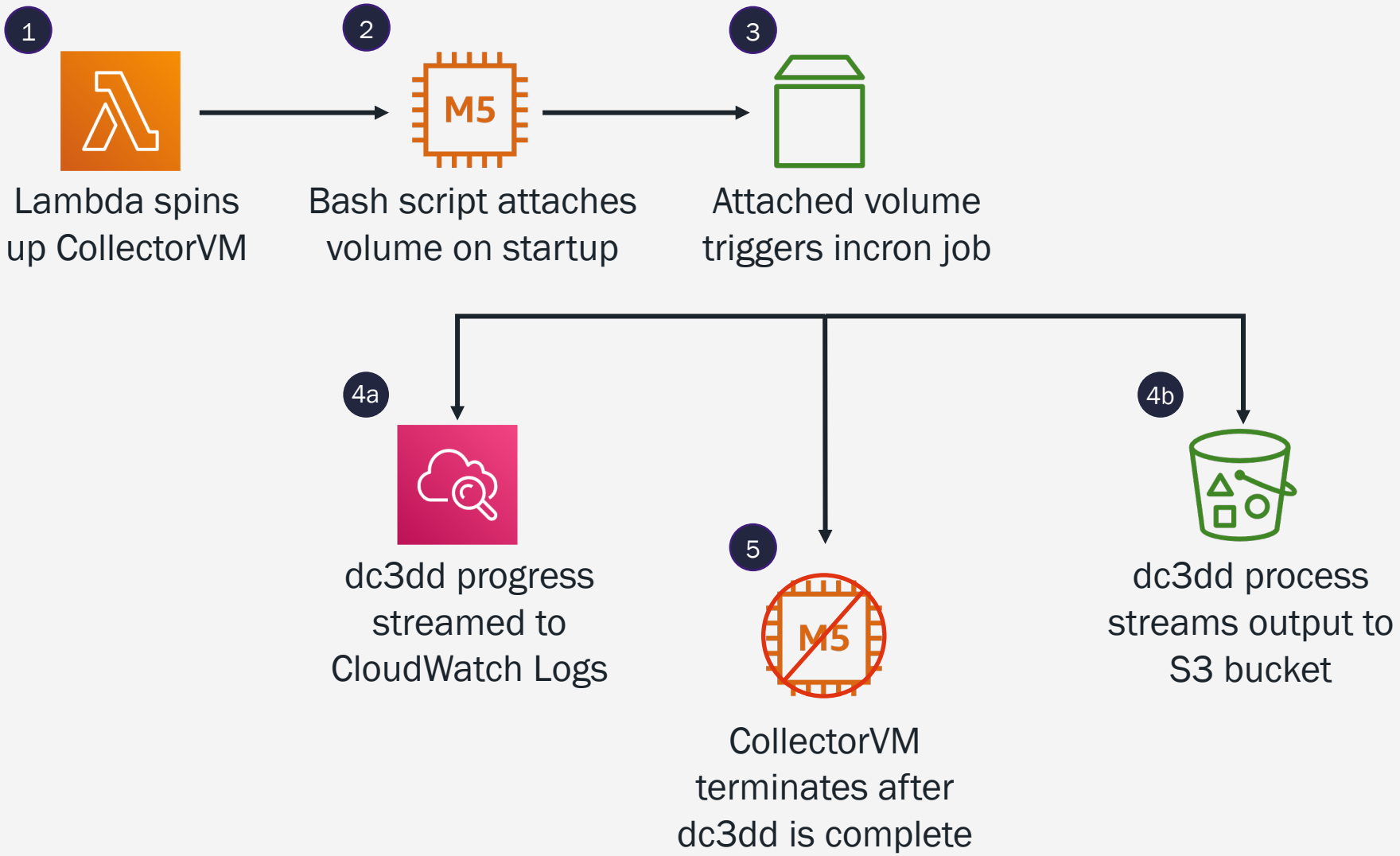
After Automation



Step Function – Disk Workflow



Collector VM AMI



CloudWatch Log Group

CloudWatch > Log Groups > CollectorVMLogs > i-0188100932061bd51

Expand all ● Row ○ Text

Filter events | all | 2019-11-26 (19:32:55)

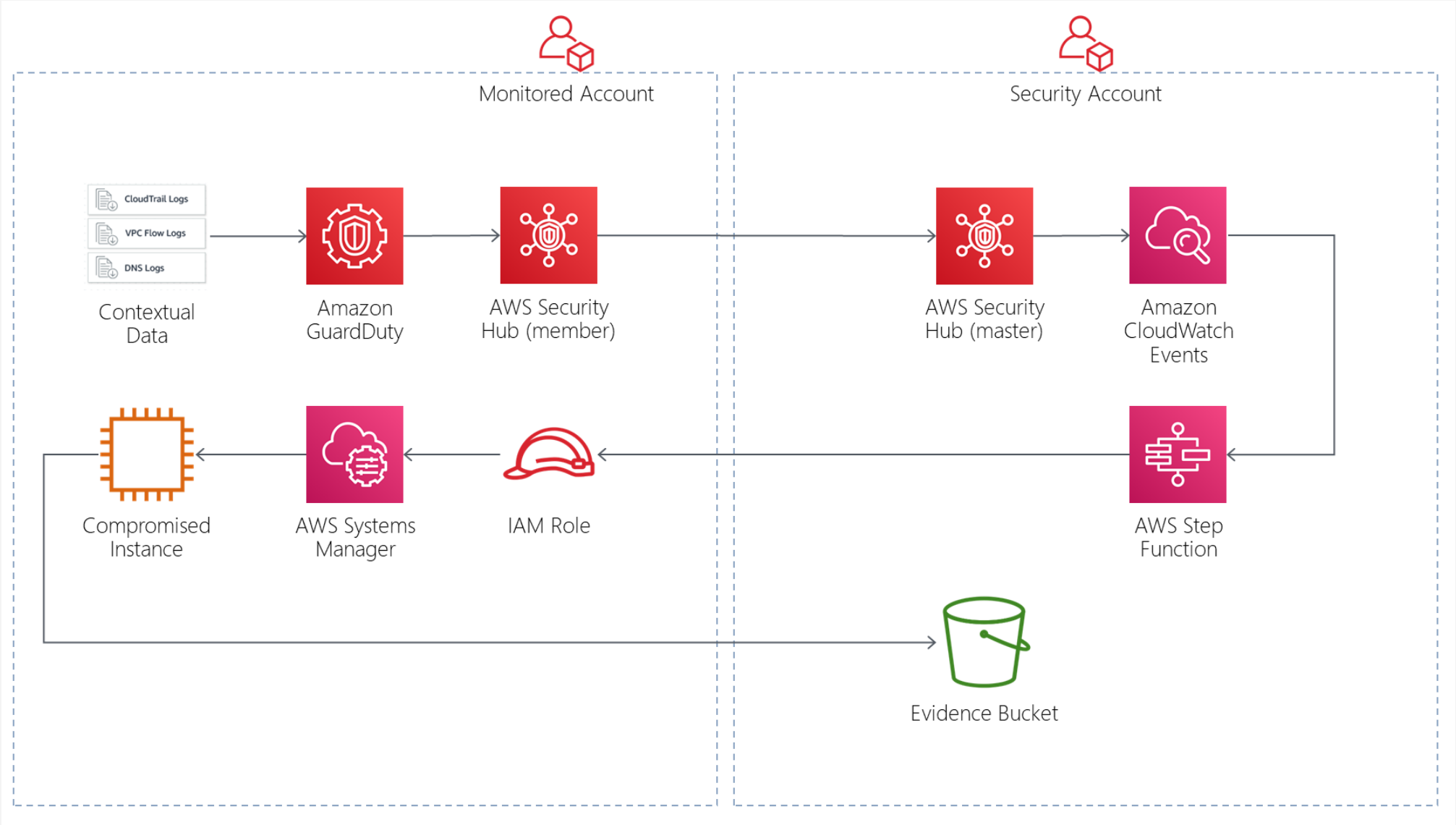
Time (UTC +00:00)	Message
2019-11-27	
	No older events found at the moment. Retry.
19:32:55	2019-11-27T19:32:55+0000 sector size: 512 bytes (probed)
19:33:10	2019-11-27T19:33:10+0000 1195376640 bytes (1.1 G) copied (14%) 15.1414 s 75 M/s
19:33:26	2019-11-27T19:33:26+0000 2705326080 bytes (2.5 G) copied (31%) 30.2665 s 85 M/s
19:33:41	2019-11-27T19:33:41+0000 3177185280 bytes (3 G) copied (37%) 45.4943 s 67 M/s
19:33:56	2019-11-27T19:33:56+0000 3617587200 bytes (3.4 G) copied (42%) 60.625 s 57 M/s
19:34:11	2019-11-27T19:34:11+0000 3995074560 bytes (3.7 G) copied (47%) 75.8563 s 50 M/s
19:34:26	2019-11-27T19:34:26+0000 4435476480 bytes (4.1 G) copied (52%) 90.9839 s 46 M/s
19:34:42	2019-11-27T19:34:42+0000 4812963840 bytes (4.5 G) copied (56%) 106.208 s 43 M/s
19:34:57	2019-11-27T19:34:57+0000 5253365760 bytes (4.9 G) copied (61%) 121.335 s 41 M/s
19:35:12	2019-11-27T19:35:12+0000 5662310400 bytes (5.3 G) copied (66%) 136.461 s 40 M/s
19:35:27	2019-11-27T19:35:27+0000 6039797760 bytes (5.6 G) copied (70%) 151.686 s 38 M/s
19:35:42	2019-11-27T19:35:42+0000 6448742400 bytes (6 G) copied (75%) 166.916 s 37 M/s
19:35:57	2019-11-27T19:35:57+0000 6889144320 bytes (6.4 G) copied (80%) 182.047 s 36 M/s

Sky Collect

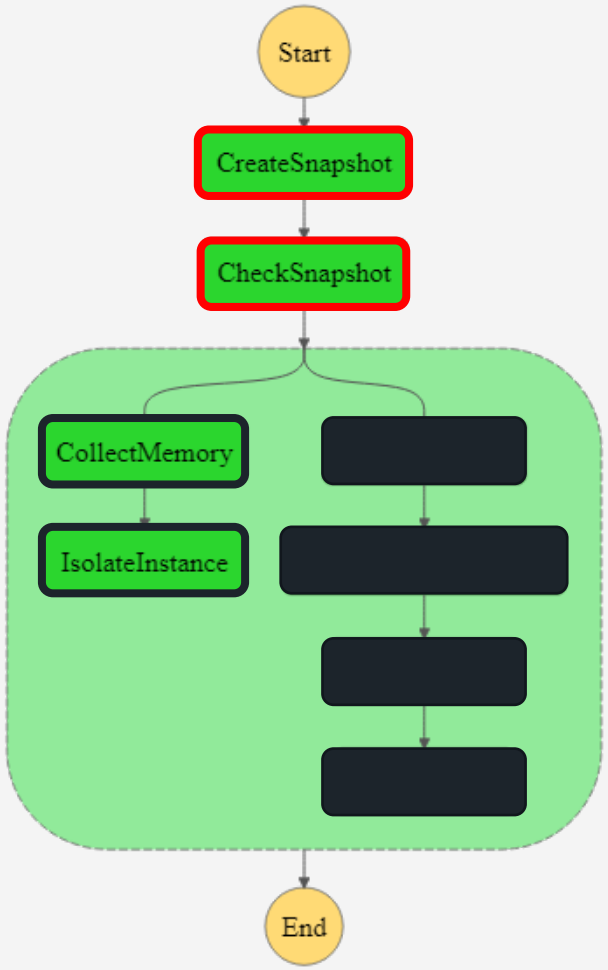
MEMORY WORKFLOW

Before Automation

After Automation



Step Function – Memory Workflow



Memory Profile (Linux)

- Create the LiME kernel module used to collect memory upon reboot if kernel changes
- Create the dwarf file containing the Linux kernel data structures
- Zip up dwarf file and System map file for the active kernel version

Closing Thoughts

Next Steps

- Collect instance store data
- Further leverage FIM to restrict access to necessary tools

- Automate analysis (T1/T2) of evidence using ECS/Fargate
  - Currently Fargate does not support Windows platform versions

- Assign confidence score based off analysis
- Generate example reports

Appendices

Appendix A: Getting Started with Disk

- Leverage and compare multi-threading and concurrent execution with Lambda
- Determine the necessity for a full disk collection vs. a triage collection
- Stream dc3dd output directly to S3
- Leverage VPC endpoints wherever possible
- Consider the tradeoff between cost and collection speed
- Remember that instances store data is not collected during an EC2 snapshot
- Restore EBS snapshots to HDDs (as opposed to SSDs) for hash verification

Appendix B: Getting Started with Memory

- Stream directly to S3 and include the expected size flag
- Consider the potential business impact of collecting memory
- Research the complexities of gateway VPC endpoints
- Leverage a file integrity monitoring (FIM) solution where possible
- Verify memory evidence before stopping or terminating the original host
- Understand the difference between tracked and untracked connections

Appendix C: Further Reading

- AWS Security Incident Response Guide: https://d1.awsstatic.com/whitepapers/aws_security_incident_response.pdf
- Best Practices for Security, Identify, & Compliance: https://aws.amazon.com/architecture/security-identity-compliance
- Connection Tracking: https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ec2-security-groups.html#security-group-connection-tracking
- Shared Responsibility Model: https://aws.amazon.com/compliance/shared-responsibility-model

Thank You

Automated Detection and Response in AWS

Ryan Tick
Cloud Security Architect
Goldman Sachs

Vaishnav Murthy
Cloud Security Architect
Goldman Sachs