Securosis

Blowing stuff up at scale with pitch perfect attack simulations

Rich Mogull
@rmogull

Will Bengtson
@__muscles

Topoff II

In 1998, Congress foresaw the value of multi-level, multi-agency, and multi-jurisdictional exercises to be used as training for top government officials in responding to terrorist attacks involving Weapons of Mass Destruction (WMD). These exercises became known as TOPOFF (Top Officials). The second of these exercises, TOPOFF 2, took place in Seattle, Washington and Chicago, Illinois in May 2003. Sponsored by the United State Department of Justice, this exercise brought together local, state, national, and international governments as well as non- governmental entities to test and ultimately to produce a more effective response to the threat of WMD terrorism.

TOPOFF 2 (T2) began on May 12, 2003, when two major U.S. cities, Seattle and Chicago, were the subjects of simulated terrorist attacks. T2 was the most comprehensive and "real" test of local, state, and national response capabilities ever conducted in our nation's history. The Chicago attack involved bioterrorism, while the simulated attack on Seattle was the detonation of a "dirty bomb" or radiological dispersal device (RDD).

- https://www.hsdl.org/?view&did=772553

Securosis

Securosis

Securosis

Securosis


NMRT
939


HURRACAN
KATRINA
RELIFE
FC07


American
Red Cross


NUMBER ENDS WITH
84 - 86
GO TO THE
4TH LEVEL
SECTION 457 ROW
Reliant Energy
Continental Airlines
NO FOOD
ON

Securosis

Securosis

The Practitioner

HASHICORP

NETFLIX

CapitalOne

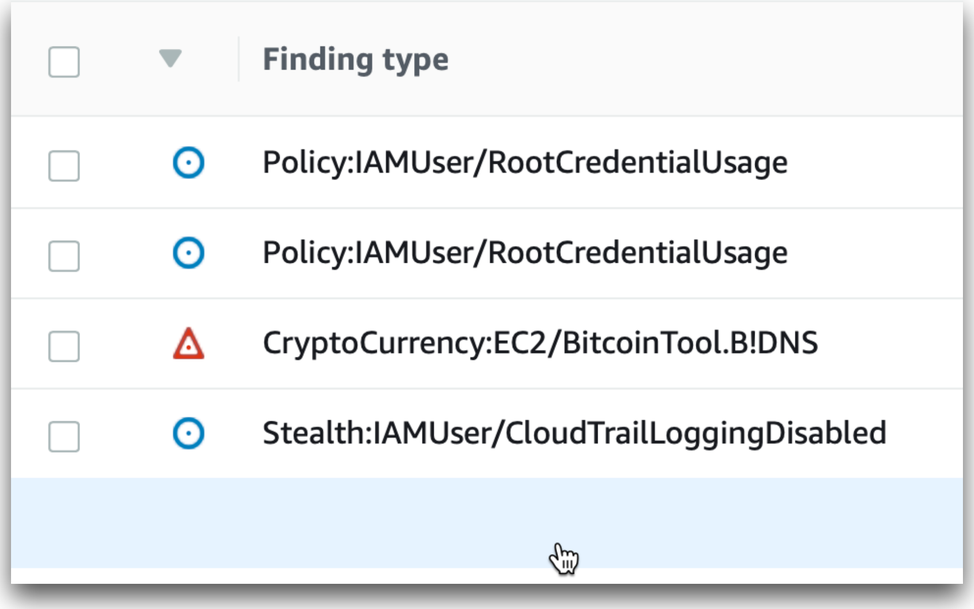
The Researcher

FIREMON

Securosis

{disrupt:Ops}

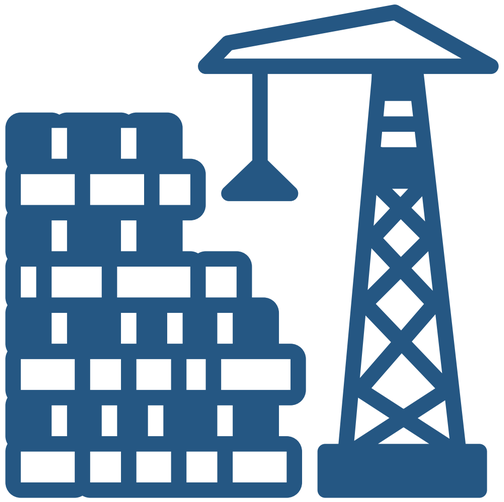
Securosis


Finding type
Policy:IAMUser/RootCredentialUsage
Policy:IAMUser/RootCredentialUsage
CryptoCurrency:EC2/BitcoinTool.B!DNS
Stealth:IAMUser/CloudTrailLoggingDisabled

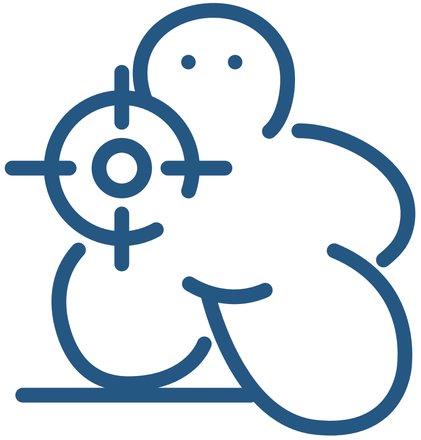
Principles of Scenarios

- Progressive
  - Skills stations
  - Constrained scenarios
  - “Megacode” full spectrum simulations
- Guided -> Unguided
- Includes processes
- Includes teamwork
- Realistic setups

Scaling Simulations



Setup



Attack

Cleanup

Securosis

Setup Constraints

- Single account in an Organization
  - Multi-account per student is too complex to effectively manage
  - SCPs essential to maintain account control
    - Protect OrgAccountAccessRole, root account, external tooling
- Should look and feel like a real account
- Any attack pathways should blend in with the environment
- Not so vulnerable it is exploited before the student gets hands-on
- Should be clean with no vestiges of prior attacks/config

DISCLAIMER

The remainder of this presentation contains spoilers!

Environment Components

- 3 VPCs
  - Shared Services, Dev, Prod
- CloudTrial, GuardDuty, SecurityHub, Config
- 100 IAM users
- 50 IAM roles
- 50 custom policies
- 15 instances
  - 10 in dev/prod autoscale groups
  - JumpBox, SecOps, Logger, 2 skills stations
- Dynamo, Lambda, and some other services

5 CloudFormation Templates
Attack resource generator
Student login generator

Securosis

Adding Realism

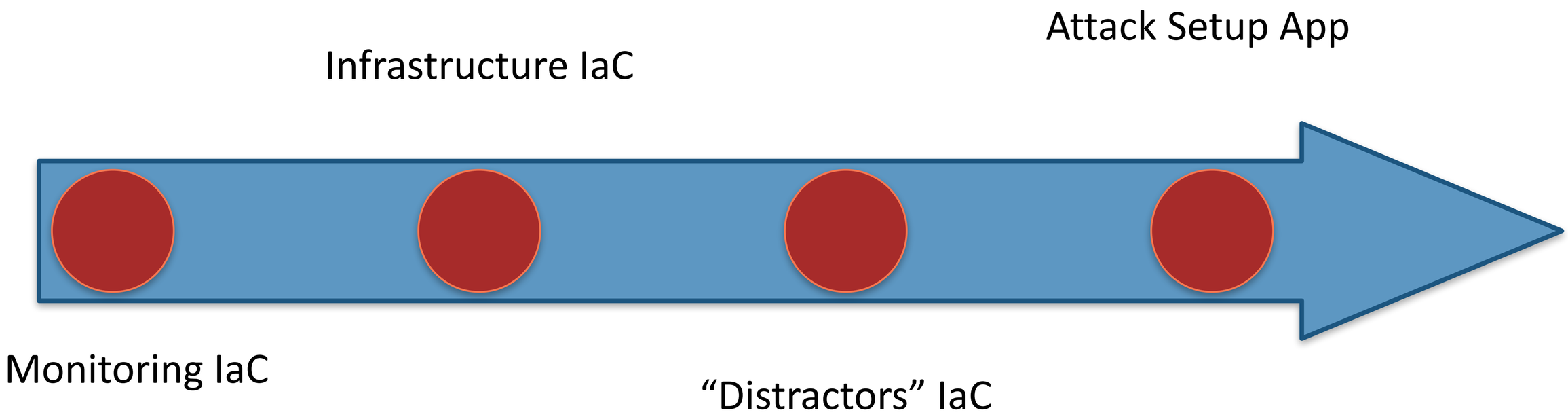
- Procedurally generate
  - Used real name lists
  - Generated roles with combinations of common roots
  - Create customer managed IAM policies by duplicating and renaming AWS managed
- Watch out for S3 naming
- Only need to run once to generate template
  - Use same patterns for attack resources
  - For a game day use deterministic names that match environment

```
    environmentlist = ['Dev', 'Prod', 'Stage', 'Test', 'QA', 'Website', 'Shared', 'Ops', 'App1', 'Public', 'App2']
    rolelist = ['developer', 'admin', 'qa', 'operations', 'manager', 'backup', 'ops', 'reader', 'security']
    servicelist = ['ec2', 'lambda', 'rds', 's3', 'SSM', 'IAM']
    terms = ['customer', 'appinfo', 'data', 'backup', 'shared', 'internal', 'temp', 'test']
    cfn_start = """AWSTemplateFormatVersion: "2010-09-09"
Description: "IR Training Monitoring Setup"
Resources:
"""
```

Securosis

Setup Process

One Time Setup: External Tool Integration



Day Before Class: Student Account Assignment App

Securosis

Attack Setup App (Current)

AccountID,
username, secrets

Create users, credentials,
least-privilege (*almost*)
policies

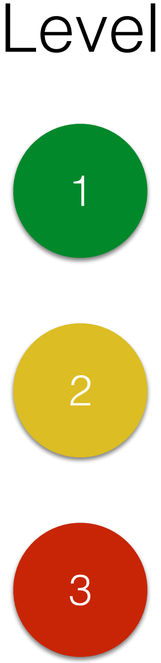
… 60+ accounts

Securosis

Demo and Tour

Skills Stations and Scenarios

- Skills stations
  - Focused on a specific skill
  - Constrained and directed
- Simple Scenario
  - Focus on a process or a subset of skills
  - Contrived, and is more directed. Sometimes tabletop
- “Megacode” full scenarios
  - As realistic as possible
  - Test the full IR process from start to finish
- Game Days



Securosis

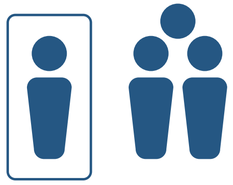
“Skills Stations”

- Focus on specific skills
- Highly constrained
- Should still look real
  - Analysis
    - Trace a misconfiguration
    - Trace an attack
  - Containment
    - Isolate an IAM role (5+ techniques in AWS!)
    - Isolate network connection

Full Scenarios

Detect

Analyze



Contain/
Eradicate

Recover

- Progressive complexity
- MUST replicate real incidents as completely as posible
- Should focus on team and process, not just individual skills
- Partial to start… full spectrum by the end

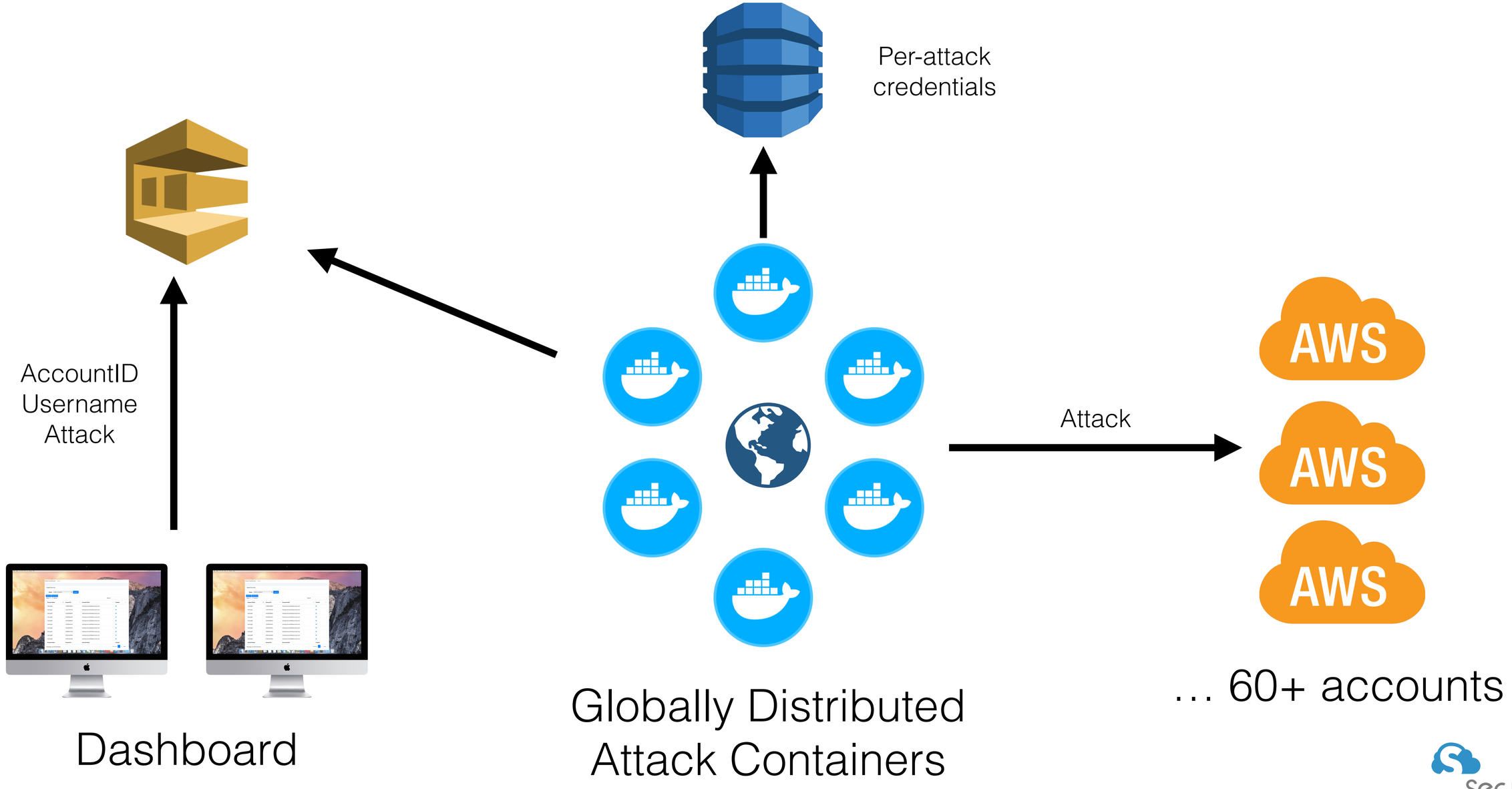
Securosis

Keeping it Real

- Attack origin (Tor? VPN?)
- Limited IAM policies (NOT all full admin)
- Recon/enumeration
- Trigger
  - Sometimes harder than you'd like
- Attack sequence
- Containment constraints

Securosis

Scalable Attack Simulation Architecture



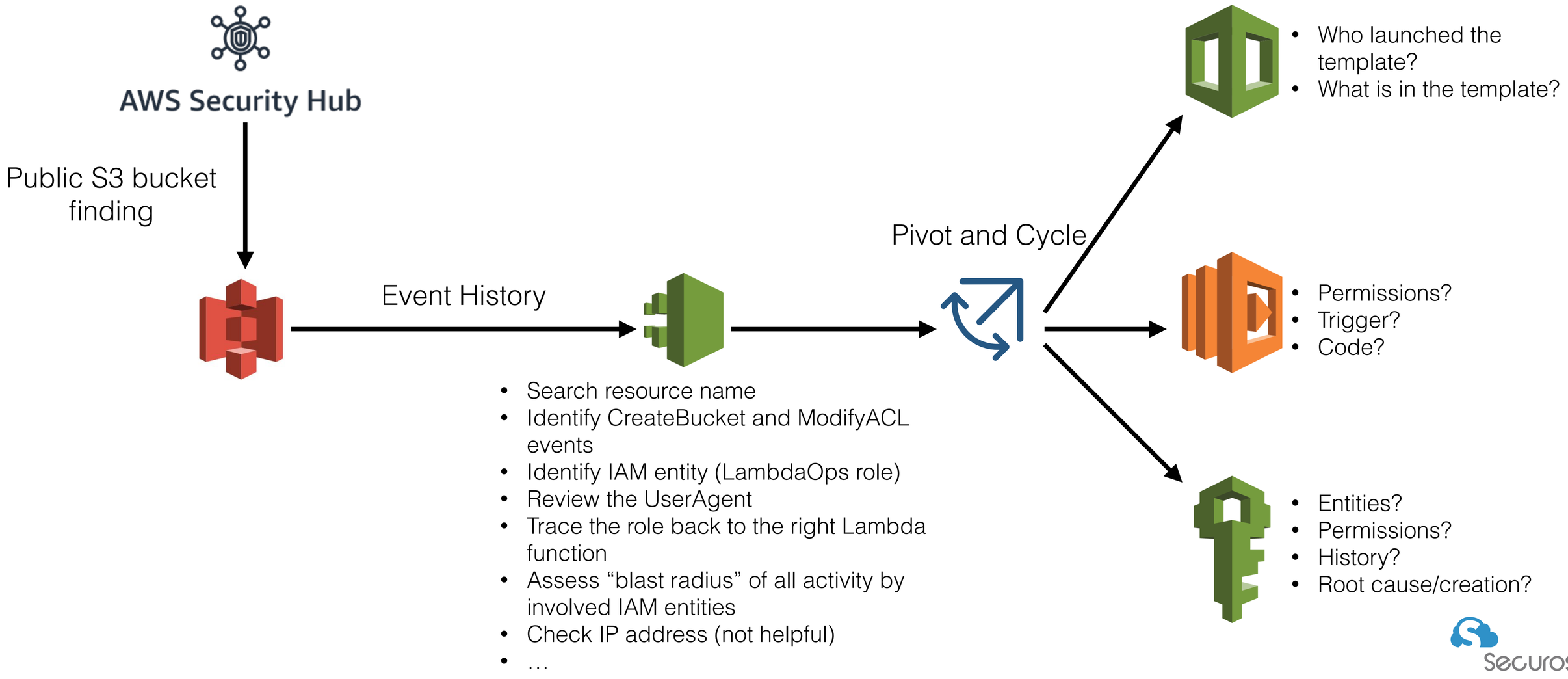
Securosis

Attack Portal

Architecture Tour

Skills Station Walkthrough

Public S3 Bucket via CloudFormation Misconfiguration



Securosis

Skills Station Demo

Skills

- Where to start when a misconfiguration is identified
- Prioritization of analysis
- Sick/Not-Sick and “stop the bleed” containment decisions
- What to look for in CF
  - We start in the event history so students don’t get hung up on the details of writing Athena queries
- How to avoid distractors and tunnel vision
- Fundamentals of tracing and pivoting analysis
  - Building the timeline and story

Full Scenario Walkthrough

Privilege Escalation to S3 Exfiltration via AMI

- Recon
  - Identify target S3
  - Identify InstanceProfile Role
- RunInstance + PutRole
  - Embed bash script into user-data field to copy S3 bucket
- Make AMI public

Public AMI

slack

Amazon EC2

- Identify AMI
- Identify base instance
  - user-data

- Search based on:
  - AMI
  - Instance
  - IAM user
  - IP address

- Find the recon
- Determine blast radius based on IAM permissions
- Identify S3 data
- Track other activity from IP

- Contain IAM

- Contain S3

- Contain AMI

Securosis

Scenario Demo

Securosis

Gamedays!

Cleanup

- Fully reset the account
- THE HARDEST PART
  - I blame S3
- Initially tried aws-nuke
  - Not designed for multi-account
  - Wrote a custom python wrapper
  - Nuke is slow, largely due to S3 objects created by StreamAlert and Cloudtrail… plus having to check everything
- Wrote a python app customized to how we use the accounts, that can also take advantage of CLI efficiencies for S3
  - Still use nuke as a secondary cleaner

Tech Details

- Nuke wrapper
  - Python using pexpect
  - Links into the account list to provide the account alias when required
- Custom cleaner
  - Leverage AWS CLI for object deletion
  - Bridge cross-account if certain dependencies detected (e.g. transit gateway/attachment)
  - Delete resources ***in proper order*** that breaks CloudFormation/etc.

Tech Lessons Learned

- Setup and cleanup should be distributed
  - Moving to a container/SQS architecture
  - Execution times are too long for lambda
- Stacksets are NOT meant for this kind of usage
  - Can’t handle S3 and often get order of operations wrong on deletion
  - Not actually parallel, a fleet of containers will be faster
- Error handling important for attacks
  - One bad message in the queue can take down a fleet of containers

Lessons Learned

- Know your attack timing
  - The time from click to IoC alert is critical to maintaining a good flow
  - e.g. GuardDuty takes time to detect
- Have backup accounts
- Attack origin matters
  - Some attacks should come from AWS. Others from outside AWS.
- *Simulate collaboration, communication, and process*


TRAINING SUIT
#1 ANDY


1-800-BY

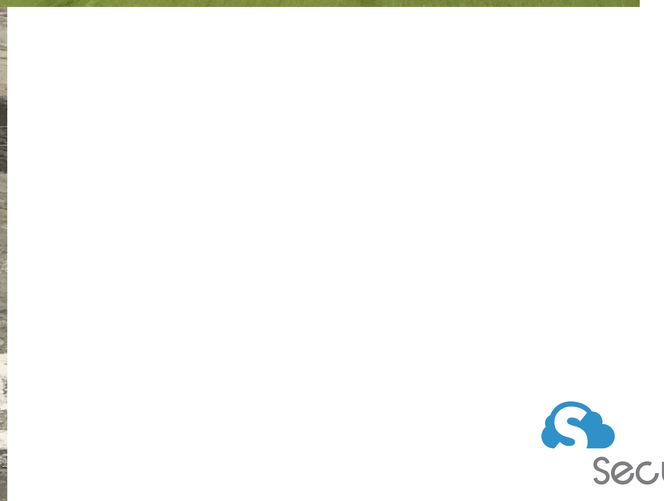

KAYENTA
HEALTH CENTER
Emergency

Securosis

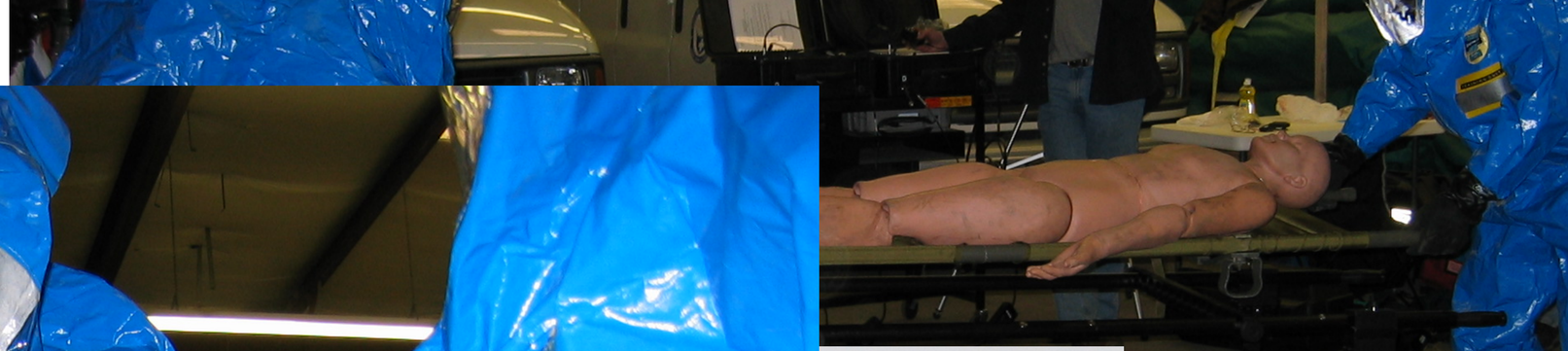
Blowing stuff up at scale with pitch perfect attack simulations

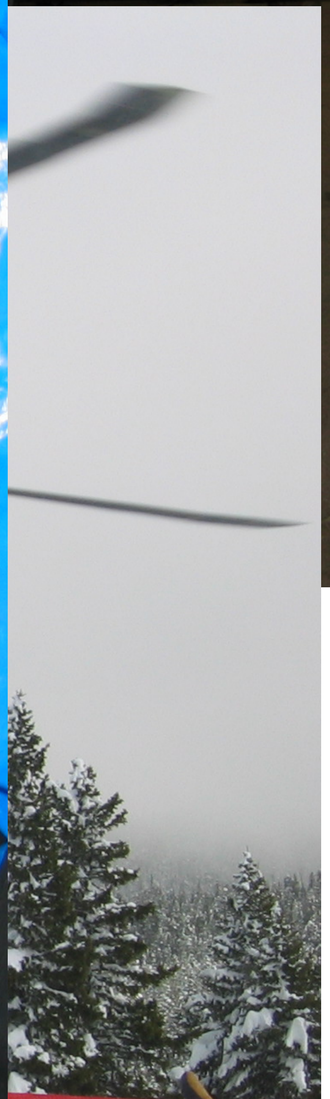
Rich Mogull
@rmogull

Will Bengtson
@__muscles



Securosis





Securosis