Continuous Cloud Security Monitoring

...will begin shortly

Take a moment to connect

linkedin.com/mwylie

twitter.com/TheMikeWylie
My Dream

# To have my own viral overly used cybersecurity acronym
# Continuous Cloud Security Monitoring (CCSM)
# One that people could drink to when used too much at RSA
# Something like AI, Cloud, Machine Learning, or Next Gen anything
# I thought CCSM was it
# Coca-Cola Signature Mixers (CCSM) stole my dream
About Me
Michael Wylie, MBA, CISSP

# Director, Cybersecurity Services @ RMTS
# Former DoD contractor & business owner
# Marine Corps Volunteer Cyber Auxiliary
# Cybersecurity professor
# Qualified TPN Assessor

Certifications

<table>
<thead>
<tr>
<th>CISSP</th>
<th>CCNA R&amp;S</th>
<th>CCNA CyberOps</th>
<th>GPEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMON</td>
<td>TPN</td>
<td>CEH</td>
<td>CEI</td>
</tr>
<tr>
<td>VCP-DCV</td>
<td>Pentest+</td>
<td>Security+</td>
<td>Project+</td>
</tr>
<tr>
<td>CHPA</td>
<td>Splunk User</td>
<td>CNVP</td>
<td>Sumo Logic Security Analytics</td>
</tr>
</tbody>
</table>

linkedin.com/mwylie
twitter.com/TheMikeWylie
CSM vs NSM
CSM vs. NSM

# Network Security Monitoring (NSM)
$ Collection and analysis of network data to detect and respond to intrusions
$ Data sources:
  > Full packet capture
  > NetFlow / sFlow
  > Alert data
  > Firewall logs
$ Focus on data in motion
$ Tools:
  > Security Onion
  > Bro
CSM vs. NSM

# Continuous Security Monitoring (CSM)
$ Ongoing automated detection and response to cyber threats
$ Continually reassess security posture
$ Keeping up with changing threat and vulnerability landscape
$ Increased visibility
$ Goal of timely incident detection
$ Focus on data at rest

“Prevention is ideal, detection is a must”
- Dr. Eric Cole
Introduction to Cloud Security
Newsworthy Cloud Breaches

# 2019 – Capital One (~100m records)
# 2019 – Lion Air (millions of records)
# 2019 – Facebook app providers (540m records)
# 2017 – Booz Allen Hamilton (admin credentials & battlefield imagery)
# 2017 – U.S. Voter Records (198m records)
# 2017 – Dow Jones (2.2m records)
# 2017 – Verizon (6m records)
# 2017 – Time Warner Cable (4m records)
# 2017 – National Credit Federation (47k records)

According to Sources: capitalone.com, bitdefender.com, darkreading.com, scmagazine.com
AWS Shared Responsibility Model

Customer Data

Platform, Applications, Identity & Access Management

Operating System, Network & Firewall Configuration

Client-side Data Encryption & Data Integrity Authentication

Server-side Encryption (File System and/or Data)

Network Traffic Protection (Encryption / Integrity / Identity)

Customer

AWS

Compute
Storage
Database
Networking

AWS Global Infrastructure

Regions
Availability Zones
Edge Locations

Source: amazon.com
Azure Shared Responsibility Model

Responsibility Zones

- **Always retained by customer**
- **Varies by Service Type**
- **Transfers to Cloud Provider**

Source: Microsoft.com
GCP Shared Responsibility Model
CloudTrail – API Calls
Config - Configuration State
CloudWatch – System Logs
Trusted Advisor – High-Level Security Events
Inspector – Vulnerability Assessment
GuardDuty – Threat Detection
CloudFormation – Gold Images
Macie – Data Breach Detection
WAF/Shield – Web App Firewall
Security Hub – Compliance Checks
Control Tower – Detect Policy Violations

Log Analytics – System Logs
Azure Security Center – Configuration State
API Console – Direct Activity

Multiple Tools – API Logging
Stackdriver - System Log
Cloud Security Scanner - Vulnerability Assessment
Dwell Time
(time from breach to detection)

### GLOBAL MEDIAN DWELL TIME

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>416</td>
<td>243</td>
<td>229</td>
<td>205</td>
<td>146</td>
<td>99</td>
<td>101</td>
<td>78</td>
</tr>
<tr>
<td>External</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>320</td>
<td>107</td>
<td>186</td>
<td>184</td>
</tr>
<tr>
<td>Internal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>56</td>
<td>80</td>
<td>57.5</td>
<td>50.5</td>
</tr>
</tbody>
</table>

Source: FireEye’s Mandiant M-Trends 2019 report
Alert Fatigue is Real

**Alarm fatigue** - Wikipedia

https://en.wikipedia.org › wiki › Alarm_fatigue

Alarm fatigue or alert fatigue occurs when one is exposed to a large number of frequent alarms (alerts) and consequently becomes desensitized to them. Desensitization can lead to longer response times or missing important alarms.

In healthcare - Unintended outcomes of ... - Solutions
28,005 VPC Flow Rejections
From Alert Fatigue to Detection & Response

Practical Advice for the Proactive SOC: How to Escape The Vicious Cycle of React

About this webinar
In this session, ExtrHop Deputy CISO Jeff Costlow will discuss how security operations teams can escape the cycle of reactivity characterized by constantly responding to a flood of alerts, and move toward a more proactive stance by using the right data sources and workflows, driven by network traffic analysis, to focus on developing proactive capabilities like continuous encryption auditing, policy auditing, and more advanced use cases like threat hunting.

Moderator: Michael Felker, Director of Information Security & Risk Management for Farmers Insurance

Speakers:
Jeff Costlow, CISO, ExtrHop
Michael Wylie, Director of Cybersecurity Services, Ritchey May Technology Solutions
Agenda
Cloud CSM (CCSM) Agenda

NOTE: Concepts are cloud platform neutral (AWS used for illustration)

# Monitoring a Non-Defensible Cloud
# Inventory Control
# Vulnerability Management
# Least Privilege
# Secure Configuration
# Monitoring & Logging
# Detecting High Fidelity Events
Monitoring a Non-Defensible Cloud
CHAPTER ONE

INTRODUCTION

Information security continuous monitoring (ISCM) is defined as maintaining ongoing awareness of information security, vulnerabilities, and threats to support organizational risk management decisions. This publication specifically addresses assessment and analysis of security control effectiveness and of organizational security status in accordance with organizational risk tolerance. Security control effectiveness is measured by correctness of implementation and by how adequately the implemented controls meet organizational needs in accordance with current risk tolerance (i.e., the control implemented in accordance with the security plan to address threats and is the security plan adequate). Organizational security status is determined using metrics established by the organization to best convey the security posture of an organization’s information and information systems, along with organizational resilience given known threat information. This necessitates:

- Maintaining situational awareness of all systems across the organization;
- Maintaining an understanding of threats and threat activities;
Monitoring a Non-Defensible Cloud

“If you know the enemy and know yourself, you need not fear the result of a hundred battles...

...If you know yourself but not the enemy, for every victory gained you will also suffer a defeat...

...If you know neither the enemy nor yourself, you will succumb in every battle.”

— Sun Tzu, The Art of War
Know Normal. Find Evil.

“In an intrusion case, spotting the difference between normal and evil is often the difference between success and failure.”

- SANS DFIR
Normal or Evil? Would you detect it?

# New instances are spun up at 3AM in Ireland region. Normal or Evil?

# 50 new instances don’t have names/tags. Normal or Evil?

# PSEexec is run on a server. Normal or Evil?
Normal or Evil? Would you detect it?

# Developers are only authorized to use Oregon and N. Virginia regions are used within AWS. New instances are spun up at 3AM in Ireland region. Normal or Evil?

# All prod services must be setup with Terraform including change management ticket# and owner in the tag. 50 new instances don’t have names or tags. Normal or Evil?

# AppLocker alerts to a new non-whitelisted PSEEXEC binary run on a server. Normal or Evil?
“know yourself”

How would you build a defensible fortress?
Defensible Agra Red Fort

# High thick walls with archers
# Draw bridges
# Water moat with crocodiles
# Dry moat with tigers
# Layered corridors for trapping intruders
# Sloped upwards towards the inner circle
# Narrow hallways with rolling boulders
# Hot/boiling oil poured into hallways
# Controlled ingress/egress points
Where would you keep your valuables?

Option A

Option B

Image Source: some sketchy site that my AV told me not to go to.

A Defensible Cloud

# Segmentation
- $ Accounts
- $ Networks

# Least privilege

# Change control

# Inventory of systems/services

# Continuous security monitoring

# Detective & preventative controls

Source: AWS - Multi-account architecture for restricting PCI DSS scope
M&E Secure Workflow
CIS Basic Best Practices
Center for Internet Security (CIS)

# Developed by SANS Institute
# Response to increased breaches and incidents
# Top 20 controls to make defense easier
# Benchmarks for secure configuration
# More than 100 contributors:
  $ US-CERT
  $ US DoD
  $ MITRE
  $ SANS
<table>
<thead>
<tr>
<th>Control</th>
<th>Control Title</th>
<th>Applicability of Service Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Inventory and Control of Hardware Assets</td>
<td>IaaS</td>
</tr>
<tr>
<td>2</td>
<td>Inventory and Control of Software Assets</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Continuous Vulnerability Management</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Controlled Use of Administrative Privileges</td>
<td>IaaS</td>
</tr>
<tr>
<td>5</td>
<td>Secure Configuration for Hardware and Software on Mobile Devices, Laptops, Workstations and Servers</td>
<td>IaaS</td>
</tr>
<tr>
<td>6</td>
<td>Maintenance, Monitoring and Analysis of Audit Logs</td>
<td>IaaS</td>
</tr>
<tr>
<td>7</td>
<td>Email and Web Browser Protections</td>
<td>IaaS</td>
</tr>
<tr>
<td>8</td>
<td>Malware Defenses</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Limitation and Control of Network Ports, Protocols, and Services</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Data Recovery Capabilities</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Secure Configuration for Network Devices, such as Firewalls, Routers and Switches</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Boundary Defense</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Data Protection</td>
<td></td>
</tr>
</tbody>
</table>

Mike Wylie
Inventory Control
My AWS account was hacked and I have a $50,000 bill, how can I ... https://www.quora.com/My-AWS-account-was-hacked-and-I-have-a-50-000-bill-how...

For years, my bill was never above $350/month on my single AWS instance. ... to find public keys for servers on GitHub and took control of the AWS account.

Spending 100K USD in 4,5 days on Amazon Web Services | OlinData https://www.olindata.com/en/blog/.../spending-100k-usd-45-days-amazon-web-servic...

Apr 25, 2017 - Spending 100K USD in 4,5 days on Amazon Web Services ... I created a separate AWS account for this and hooked it up to consolidated billing on our main ... Some more applicants took the assessment and on April 14th at ...

I was billed for 14k USD on Amazon Web Services - DEV Community ... https://dev.to/.../i-was-billed-for-14k-usd-on-amazon-web-services-17n

Jul 3, 2018 - I was billed for 14k USD on Amazon Web Services ... My monthly bill is about 1 USD. Yep, that's almost nothing, but ... Take a look at your email frequently and don't take a single email as a mistake ..... If your account was hacked, even the spending limit (which AWS does not have) system would not work.
The Cloud

# Before the “Cloud”:
$ Process: Purchase Order > Dell Quote > Purchase > Shipping > etc.
$ Accounting, server, network, and developers involved

# In the “Cloud”:
$ Process: Developers can start build public facing servers in seconds
  > EC2 Startup < 60 seconds & Lightsail < 30 seconds
  > DevOps involved

# VM Sprawl or account takeover
# Insider threats
Verizon 2019 DBIR
breaches by role

Figure 8. Select threat actors in breaches over time
Inventory Control

Inventory Summary

EC2
- Running Instances: 0
- Across 1 Regions
- EBS Volumes: 1
- Storage used: 25 GiB

Reserved Instances
- Service is not being used

S3
- Buckets: 16
- Objects: 161,049
- Storage used: 16.69 GiB

RDS
- Service is not being used

Auto Scaling
- Service is not being used

AWS Config
- Service is not being used

Certificate Manager
- Service is not being used

CloudFormation
- Stacks: 2
- Across 2 Regions

CloudFront
- Service is not being used

CloudHSM
- Service is not being used

CloudSearch
- Service is not being used
Terraform

# Helps with building, changing, & versioning infrastructure safely/efficiently

# Works with Alibaba Cloud, AWS, GCP, Microsoft Azure, OpenStack

# Plan/execute deployments

# Minimize human error

# Build/teardown quickly

# Overwrite unauthorized changes

```
resource "aws_s3_bucket" "bucket" {
  bucket = "mike-test-bucket"
  acl    = "private"
  tags = {
    Name        = "Mike's Test Bucket"
    Created_By = "Mike"
  }
}
```
Solution: PBNJ

# Runs Nmap scans and stores results in a database
# Stores IP, OS, Hostname, and other data in MySQL
# Can be run on a $3.50 AWS Lightsail instance

# Usage:
   $ scanpbnj [options] [target]
   > $ sudo scanpbnj scanme.nmap.org
   $ outputpbnj [options]
   > $ sudo outputpbnj --query latestinfo
Billing Alarms

# Billing alarms

$ Switch region to US East (N. Virginia)
$ Billing > Preferences > Receive Billing Alerts
$ Alarms > Billing > Create Alarm

When my total AWS charges for the month exceed: 10.00 USD
send a notification to: [Input Field]
Billing Alarms

You are receiving this email because your estimated charges are greater than the limit you set for the alarm "BillingAlarm" in AWS Account 123456789011.

The alarm limit you set was $ 200.00 USD. Your total estimated charges accrued for this billing period are currently $ 200.91 USD as of Wednesday 15 May, 2019 04:06:55 UTC. The actual charges you will be billed in this statement period may differ from the charges shown on this notification. For more information, view your estimated bill at: https://url.emailprotection.link/?
Vulnerability Assessment
Dealing with Results

# Finding vulnerabilities is easy
# Remediation is can be challenging
# Educate developers to avoid vulnerabilities
# Break vulnerability findings into manageable weekly ‘To Do’ tasks
# Understand ramifications and agree on an action plan
# Get outside help if needed
# Vulnerability scans are not penetration tests
# Automated tools don’t work well with custom anything
Least Privilege
IAM Basics

# Delete your root access keys
# Activate MFA on your root account
# Create individual IAM users
# Uses groups to assign permissions
# Apply an IAM password policy
# Alert on root account use
Least Privilege

#Granular IAM Policies (least privileged)

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "s3:*",
            "Resource": "*"
        }
    ]
}
```

VS.

```json
"Effect": "Allow",
"Action": [
    "s3:AbortMultipartUpload",
    "s3:DeleteObject",
    "s3:DeleteObjectVersion",
    "s3:GetObject",
    "s3:GetObjectAcl",
    "s3:GetObjectVersion",
    "s3:GetObjectVersionAcl",
    "s3:PutObject",
    "s3:PutObjectAcl",
    "s3:PutObjectVersionAcl"
]
```

AmazonS3FullAccess

Custom S3 Policy
Credential Report

- IAM > Credential Report > Download Report
- Find stale accounts
- Find unauthorized users
- Audit service accounts
- Audit MFA

<table>
<thead>
<tr>
<th>arn</th>
<th>user_creation_time</th>
<th>mfa_active</th>
<th>access_key_1_active</th>
<th>access_key_1_last_rotated</th>
<th>access_key_1_last_used_date</th>
</tr>
</thead>
<tbody>
<tr>
<td>arn/awsiam:sa-123456</td>
<td>2017-09-14T01:17:33+00:00</td>
<td>TRUE</td>
<td>FALSE</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>arn/awsiam:sa-654321</td>
<td>2019-10-20T03:07:17+00:00</td>
<td>FALSE</td>
<td>TRUE</td>
<td>2019-10-20T03:07:17+00:00</td>
<td>2019-10-21T03:02:00+00:00</td>
</tr>
<tr>
<td>arn/awsiam:sa-789012</td>
<td>2019-05-15T00:03:10+00:00</td>
<td>FALSE</td>
<td>TRUE</td>
<td>2019-05-15T00:03:11+00:00</td>
<td>2019-11-03T16:22:00+00:00</td>
</tr>
<tr>
<td>arn/awsiam:sa-321098</td>
<td>2019-05-17T05:53:51+00:00</td>
<td>FALSE</td>
<td>TRUE</td>
<td>2019-05-17T05:53:51+00:00</td>
<td>2019-05-20T12:08:00+00:00</td>
</tr>
</tbody>
</table>
Cloud Accounts

# Goal is principal of least privilege
# Often a battle between business need and least privilege
# AWS provides over 2,500 permissions
# Granular controls means complexity
# Complexity means prevalent human error
One Glove Doesn’t Fit All

Developers don’t need access to the entire cloud platform
Developers don’t need access to log buckets
All developers don’t need the same access
AWS Access Advisor shows if/when an IAM has accessed a service
Netflix expanded on Access Advisor with Aardvark & Repokid
  $ Deploy default role based permissions
  $ Review access to services and sub-services over time (e.g. 6 mo.)
  $ Revoke unneeded services
Secure Configuration
CIS Benchmarks

# Not all CIS cloud benchmarks are verifiable programmatically

# Third-party tools to help with 85% of the leg work:

- Cloudcheckr
- CloudHealth
- Cloudaware
- AWS Config
<table>
<thead>
<tr>
<th>Control</th>
<th>Scoring</th>
<th>Set Correctly</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity and Access Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Control 1.1 - Avoid the use of the &quot;root&quot; account</td>
<td>Not Scored</td>
<td>Yes</td>
</tr>
<tr>
<td>Control 1.2 - Ensure multi-factor authentication (MFA) is enabled for all IAM users that have a console password</td>
<td>Not Scored</td>
<td>No</td>
</tr>
<tr>
<td>Control 1.3 - Ensure credentials unused for 90 days or greater are disabled</td>
<td>Not Scored</td>
<td>No</td>
</tr>
<tr>
<td>Control 1.4 - Ensure access keys are rotated every 90 days or less</td>
<td>Not Scored</td>
<td>No</td>
</tr>
<tr>
<td>Control 1.5 - Ensure IAM password policy requires at least one uppercase letter</td>
<td>Not Scored</td>
<td>Yes</td>
</tr>
<tr>
<td>Control 1.6 - Ensure IAM password policy require at least one lowercase letter</td>
<td>Not Scored</td>
<td>Yes</td>
</tr>
</tbody>
</table>
1.1 Avoid the use of the "root" account (Scored)

Profile Applicability:

- Level 1

Description:

The "root" account has unrestricted access to all resources in the AWS account. It is highly recommended that the use of this account be avoided.

Rationale:

The "root" account is the most privileged AWS account. Minimizing the use of this account and adopting the principle of least privilege for access management will reduce the risk of accidental changes and unintended disclosure of highly privileged credentials.

Audit:

Implement the following two log metric filter and alarm exist for usage of "root" account recommendation in the Monitoring section of this benchmark to receive notifications of root account usage. Additionally, executing the following commands will provide ad-hoc means for determining the last time the root account was used:

```
aws iam generate-credential-report

aws iam get-credential-report --query 'Content' --output text | base64 -d | cut -d -f 1,5,11,16 | grep -A 1 'root_account>
```

Note: there are a few conditions under which the use of the root account is required, such as requesting a penetration test or creating a CloudFront private key.

Remediation:

Follow the remediation instructions of the following recommendation

Ensure IAM policies are attached only to groups or roles

References:

2. CIS CSC v6.0 #5.1

Mike Wylie
17.3.2 (L1) Ensure 'Audit Process Creation' is set to include 'Success' (Scored)

Profile Applicability:
- Level 1 (L1) - Corporate/Enterprise Environment (general use)

Description:
This subcategory reports the creation of a process and the name of the program or user that created it. Events for this subcategory include:

- 4688: A new process has been created.
- 4696: A primary token was assigned to process.

Refer to Microsoft Knowledge Base article 947226, "Windows Vista and in Windows Server 2008" for this setting.

The recommended state for this setting is to include.

Rationale:
Auditing these events may be useful when investigating policy violations.

Audit:
Navigate to the UI Path articulated in the Remediation and make the configuration prescribed.

Remediation:
To establish the recommended configuration via GP, set the following UI path to include:

Success:

```
Computer Configuration\Policies\Windows Settings\Security Settings\Advanced Audit Policy Configuration\Audit Policies\Detailed Tracking\Audit Process Creation
```
Monitoring & Logging
# AWS Logs You Can’t Live Without

<table>
<thead>
<tr>
<th>Service</th>
<th>What</th>
<th>Delay</th>
<th>Storage</th>
</tr>
</thead>
<tbody>
<tr>
<td>CloudTrail</td>
<td>API calls</td>
<td>15 minutes</td>
<td>S3</td>
</tr>
<tr>
<td>CloudWatch Events</td>
<td>Specific API calls</td>
<td>Real-time</td>
<td>CloudWatch Events (optional -&gt; SES)</td>
</tr>
<tr>
<td>VPC</td>
<td>Network flow</td>
<td>15 minutes</td>
<td>CloudWatch Logs</td>
</tr>
<tr>
<td>S3</td>
<td>Bucket access</td>
<td>Hourly</td>
<td>S3</td>
</tr>
<tr>
<td>ELB</td>
<td>Web requests</td>
<td>5 minutes</td>
<td>S3</td>
</tr>
<tr>
<td>CloudFront</td>
<td>Cache requests</td>
<td>Up to 24 hours</td>
<td>S3</td>
</tr>
</tbody>
</table>

Source: Jonathon Poling
CloudTrail

# CloudTrail enabled by default for 90 day retention
# External report of breaches is still > 100 days (2018)
# Each region’s logs are kept in that region’s buckets

$ A.K.A. un-centralized logging
CloudWatch Tips

# Data can be overwhelming
# Default alerts are every 15 min
# Detailed Monitoring is every 1 min
# S3 is already storing all API calls
# Don’t create unnecessary noise/alerts
# Alert on high fidelity items only
  
  $\quad$ E.g. Alert on 0.0.0.0/0 Security Group change
  
  $\quad$ E.g. CloudWatch for CloudTrail being disabled
S3 Logs

# Enable access logging
# By default, CloudTrail logging bucket can be deleted by anyone with AmazonS3FullAccess policy
# Enable MFA to delete
# ~1 hour delay on logs
VPC Flow Logs

# Similar to sFlow/NetFlow
# Needs to be enabled on all VPCs
# Great for threat hunting and spotting evil
# Create baseline of traffic and look for anomalous traffic
  $ SSH/RDP brute force
  $ Access from non-US IPs
  $ DNS/NTP going to non-authorized destinations
AWS Flow Logs
# Problem

**May 6-12 2019**  
**RDP Connection Attempts**

<table>
<thead>
<tr>
<th>Initator IP</th>
<th>Initiator Host</th>
<th>Initiator MAC</th>
<th>User</th>
<th>Connections</th>
<th>Transferred</th>
</tr>
</thead>
<tbody>
<tr>
<td>193.188.22.33</td>
<td>e8.ed:<a href="mailto:05@vultr.com">05@vultr.com</a></td>
<td></td>
<td></td>
<td>16</td>
<td>442.9 MB</td>
</tr>
<tr>
<td>193.188.22.17</td>
<td>e8.ed:<a href="mailto:05@vultr.com">05@vultr.com</a></td>
<td></td>
<td></td>
<td>1254</td>
<td>86.47 MB</td>
</tr>
<tr>
<td>193.188.23.7</td>
<td>e8.ed:<a href="mailto:05@vultr.com">05@vultr.com</a></td>
<td></td>
<td></td>
<td>334</td>
<td>120.41 KB</td>
</tr>
</tbody>
</table>

*route: 193.188.22.0/24*
*descr: r7servers.ru*
*origin: AS57043*
*mnt-by: HOSTKEY-MNT*
*created: 2019-01-09T13:10:*
*last-modified: 2019-01-09T13:10:*
*source: RIPE*
Detecting High Fidelity Events
Detection

# Each organization will need to tune their detection differently
# Alerts need to be actionable
# Goal is < 10 alerts/day/analyst
# Sample high fidelity events
  $ Alert on activity on regions you’re not using
  $ Create and alert on access to HoneyBuckets
  $ Alert on attempts to disable CloudTrail
  $ Alert on attempts to delete logging buckets
  $ Alert on access from foreign non whitelisted IPs
  $ Alert on machine generated names (use freq.py)
HoneyPot

# The first public honeypot was Fred Cohen's Deception ToolKit in 1998
   $ "intended to make it appear to attackers as if the system running DTK [had] a large number of widely known vulnerabilities"

# Decreased number of false positives
# Requires less data collection
# Great way to create actionable high fiddley alerts
# Canned Example: Open Canary
# DIY Example: HoneyBucket
HoneyBucket

# Create a bucket (e.g. richey-may-db-backup)
# Allow public access & enable detailed logging
# Lambda function or use SNS to alert on actions
HoneyToken

# Fictitious database entries
  $ Names
  $ Credit Card Numbers
  $ PII

# Setup DLP to alert on use

# Setup ModSecurity on Apache to alert on HoneyToken Breach
HoneyCreds

# Create AWS IAM Credential without any permissions
# Setup CloudTrail & CloudWatch to alert on use
# Place credentials in locations attackers typically enumerate
  $ E.g. ~/aws/credentials
  $ In a file called credentials on a developer’s desktop
  $ In an environment variable
# Automate playbooks (optional)
Thank You

linkedin.com/mwylie

twitter.com/TheMikeWylie

www.TheMikeWylie.com