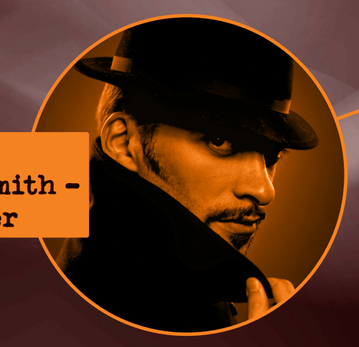
SANS
CLOUD SECURITY
PRESENTS

Finding Sherlock

Written by Shaun McCullough

Introduction

Sherlock Holmes, the world-renowned detective, has disappeared without a trace. Only his brother, Mycroft, knows of his whereabouts. Professor Moriarty has discovered that clues to Sherlock's location are hiding on Mycroft's highly secured private cloud infrastructure. With his gang of rogue hackers, Moriarty sets out to infiltrate Mycroft's infrastructure to uncover the truth and finally track down the elusive detective. Will Moriarty and his gang be successful in their mission, or will Mycroft's defenses prove to be too much for them to handle? The game is on in this thrilling tale of hacking and deception.



ATTACKER:
Culverton Smith - Serial Killer

Locating the Infrastructure

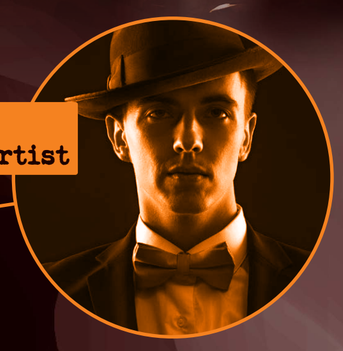
TACTIC: Initial Access | Lateral Movement

TECHNIQUES: T1213.003 | T1078.004

Culverton Smith, as vile as person as you can find, was assigned to uncover an entry point into Mycroft's system. Recognizing that initial access points often come in the form of **stolen credentials**, **hacked apps**, or **unsecured resources**, he set out to investigate Mycroft's online presence.

Utilizing his connections, he uncovered that Mycroft had a disposable email address. On a hunch, he searched for the address on other social networks and digital repositories. He stumbled upon a GitHub account belonging to Mycroft, dedicated to documenting and researching old and rare books.

As a bibliophile himself, Culverton meticulously examined the website and repository. Finding nothing of interest, he delved into the **commit history**. He made a groundbreaking discovery there: an overwritten commit that held long-term access credentials to a cloud environment. Grasping this opportunity, Culverton retrieved the credentials and tested them. To his delight, he successfully **gained access** to Mycroft's cloud infrastructure.



ATTACKER:
Henry Peter - Con Artist

Discovery

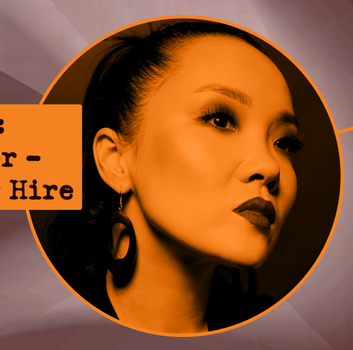
TACTIC: Discovery

TECHNIQUES: T1526 | T1580

Armed with the discovered credentials, Moriarty enlists the aid of Henry Peter, a seasoned con artist with a talent for selecting the ideal target.

Henry begins to scrutinize the **cloud's API**, utilizing the credentials acquired by Culverton. To his disappointment, he finds that the credentials are limited and do not grant access to storage services, containers, or any other credentials.

It seems that Mycroft's infrastructure operates using a **CI/CD process**, where changes to the infrastructure and website are made locally, then pushed to the CI/CD service. The discovered credentials were only used for this purpose and nothing more. The CI/CD service was found to be devoid of any valuable information, proving to be a dead end. Moriarty takes Henry's findings and passes them along to Irene Adler.



ATTACKER:
Irene Adler - Hacker for Hire

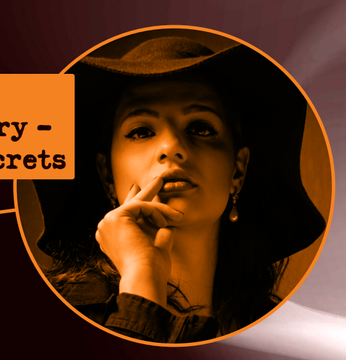
Escalation and Evasion

TACTIC: Privilege Escalation | Defense Evasion

TECHNIQUES: T1535 | T1098

Moriarty realized that he needed Adler's expertise to escalate their limited privileges and gain further access to the environment. Irene carefully analyzed the environment, searching for opportunities to **escalate privileges** through **misconfigured accesses**. Focusing on the CI/CD service, she wondered how far she could manipulate it. Upon examining the previous code, she discovered that the stolen credentials were created through this service. She realized she could escalate her own privileges. With a strategic update, she upgraded her limited privileges to full administrative access. Success!

With elevated privileges, she set about creating **backdoors** to ensure Moriarty's continued access in case of discovery. She established a **virtual machine** in an unused region with full admin rights and a backdoor, and returned the stolen credentials to their original limited state to avoid detection. With these enhanced credentials, she conducted a brief reconnaissance and discovered access to new storage services, including one labeled "CLASSIFIED."



ATTACKER:
Violet Norbury - Leaker of Secrets

Collection

TACTIC: Collection

TECHNIQUES: T1074 | T1530

Violet Norbury, a British government employee with a history of selling classified information, carefully evaluates each piece of data she acquires.

With the backdoor established by Irene, Violet accessed the "CLASSIFIED" files stored in the cloud. As they were not protected by customer-generated encryption, she was able to view their contents. To avoid suspicion, she **transferred** the files to the backdoored virtual machine and then to a **publicly accessible storage service** acting as a website. From this public website, she safely downloaded the information for review.

Although Moriarty was solely seeking information about Sherlock Holmes, Violet kept all the files, anticipating their potential use in the future. Among the files, one stood out, tagged as "MINDPALACE" and encrypted. She knew this was a crucial find.

ATTACKER:
Professor Moriarty - Mastermind

Finale

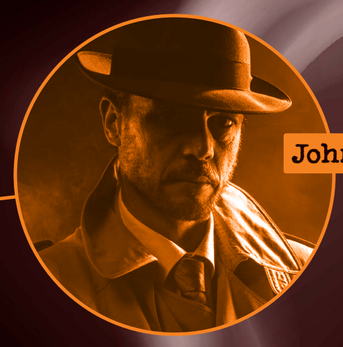
TACTIC: Impact

TECHNIQUES: T1485 | T1491

Moriarty received the encrypted MINDPALACE file from Violet, eager to discover its contents. He set up a decryption program, using various cipher methods and a dictionary of potential keys. After hours of processing, the program finally succeeded in cracking the code, revealing a message from none other than Sherlock himself:

> "Moriarty, it's always amusing to see you chase after me like a wild goose. But, I'm afraid your latest pursuit is nothing more than a red herring, a mere diversion that's led you astray. Better luck next time, old boy!"

Infuriated by the message, Moriarty used the **backdoor access** to gain full control over the network. In a fit of rage, he vandalized the website and **destroyed all the classified data** stored in the account.



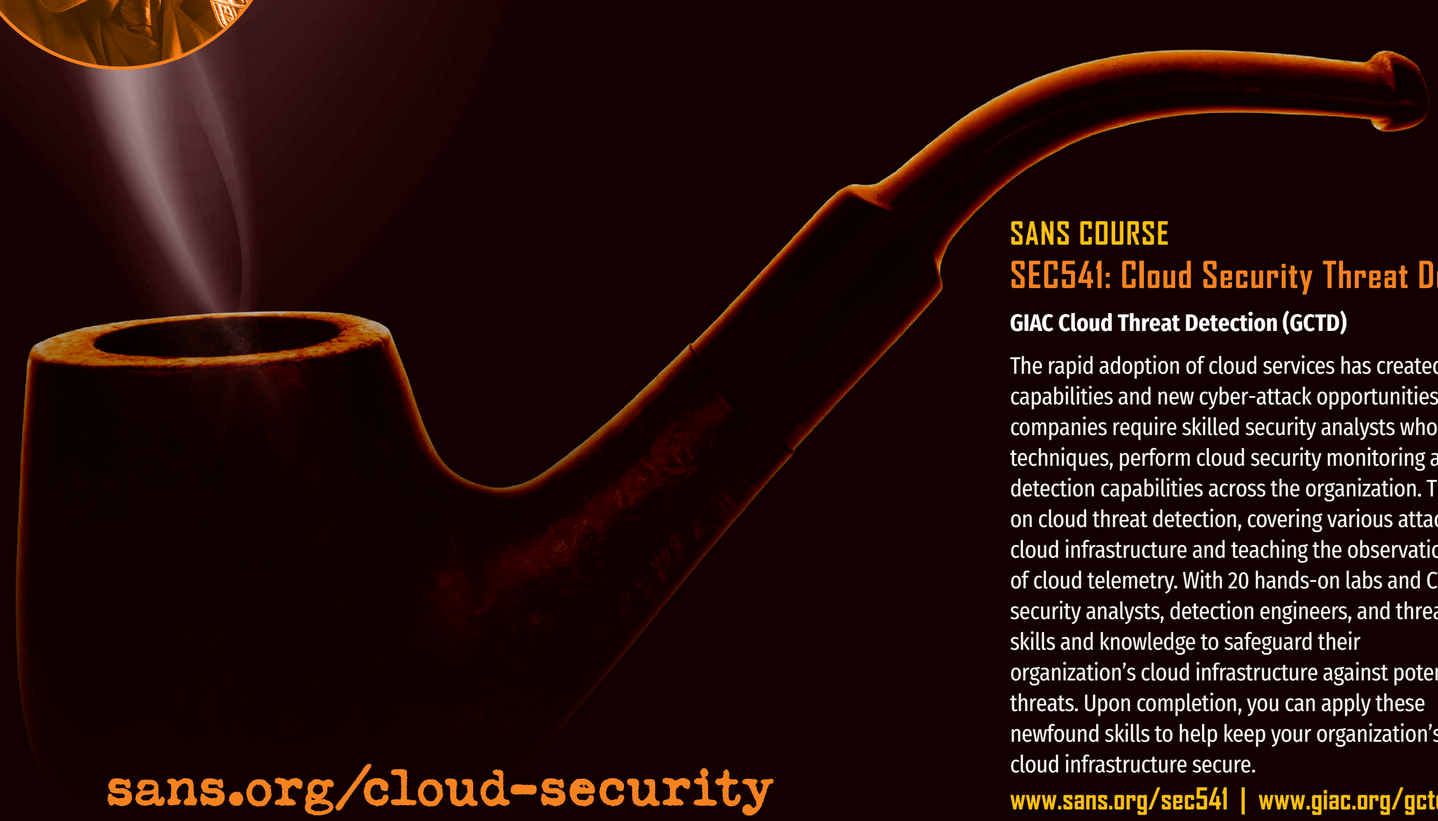
John H. Watson, M.D.

Epilogue

Sherlock and Mycroft fabricated this network of deceptions for the sole purpose of fooling the brilliant Moriarty. How did I know what the attackers did? Mycroft was watching everything with logs collected from the **cloud environment**. Using this **Deception** Network, we learned about the gang's modus operandi.

Your's truly, John H. Watson, M.D.

ps. Turn this poster over to learn more about the logs in AWS and Azure and what secrets they may reveal.



sans.org/cloud-security

SANS
CLOUD SECURITY

sans.org/cloud-security
linkedin.com/showcase/sanscloudsec
@SANSCloudSec
sansurl.com/cloud-discord

CS_SEC541_v1.4_0624

SANS COURSE

SEC541: Cloud Security Threat Detection

GIAC Cloud Threat Detection (GCTD)

The rapid adoption of cloud services has created exciting new business capabilities and new cyber-attack opportunities. To detect these threats, companies require skilled security analysts who understand attack techniques, perform cloud security monitoring and investigations, and detection capabilities across the organization. The SEC541 course focuses on cloud threat detection, covering various attack techniques used against cloud infrastructure and teaching the observation, detection, and analysis of cloud telemetry. With 20 hands-on labs and CTF, this course equips security analysts, detection engineers, and threat hunters with practical skills and knowledge to safeguard their organization's cloud infrastructure against potential threats. Upon completion, you can apply these newfound skills to help keep your organization's cloud infrastructure secure.

www.sans.org/sec541 | www.giac.org/gctd

Detect

Resources and the Logs They Create

Azure & AWS

Management API Logs

CloudTrail (AWS) and **Azure Activity Logs** (Azure) are the most important logs at your disposal. They detail user and resource authentication, interactions with the cloud's management API, tracking creations, deletions, and changes to cloud-managed resources. CloudTrail also tracks read activity to those resources, which Azure Activity Logs do not capture read events.

Detect

Any attack that engages with the cloud API could be detected in these logs whether evading defenses by spinning up resources in unused regions, creating new accounts, or destroying cloud resources. AWS's CloudTrail will also allow the detection of Discovery (TA0007) attacks. Irene Adler manipulated her access policy to escalate privileges. Any changes to privileges should be monitored and evaluated.

Databases

Azure and AWS provide a number of database services as part of their cloud offering. Some databases are cloud unique, such as **DynamoDB** (AWS) and **CosmosDB** (Azure). Others manage the execution of well-known databases, such as **MySQL** and **Postgress**. Each database type has its own logging options, but we typically look at **access** and **query** logs. Access logs can tell us how a connection was made and the amount of data transferred, typically with the source IP address, user agent string, and request/response sizes. Query logs will store the full query that is made in each connection.

Detect

OWASP A03_2021-Injections, T1190, T1005*
Just as cloud storage logs are needed to tell what files were accessed, database logs can tell us if an attacker accessed database content. Query logs can be used to determine if SQL injection manipulated web application calls to a database, or access logs to see if a user gained direct access to the database. Query logs tend to have a lot of unnecessary or redundant logs, but are helpful in an investigation.

*This MITRE ATT&CK technique does not exactly match the attack we are describing, so we picked the closest.

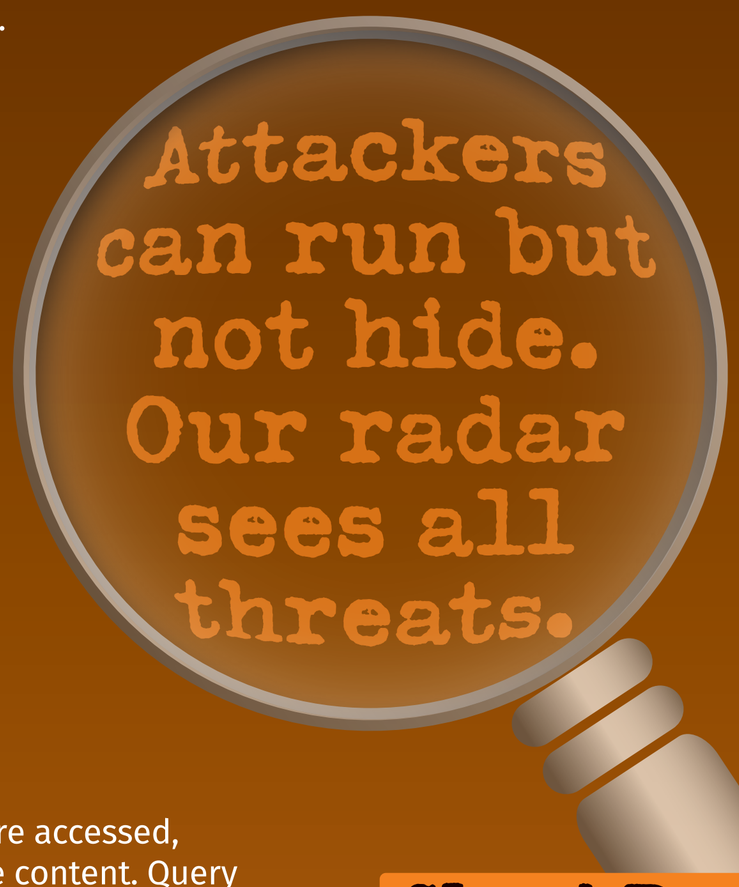
Network

Network traffic data through **VPC Flow Logs** (AWS) and **NSG Flow Logs** (Azure) provides metadata about the network conversation with data such as timestamp, destination and source port and IP address, amount of data exchanged, and the network interface involved. Fun fact: AWS has a default version (2) of the flow logs, but you should customize the log format with additional data in versions 3-5. Azure's flow logs are cryptic, but sending the data to the Log Analytic Workspace with traffic analytics turned on will translate the data into something more queryable.

VPC Network Mirroring (AWS) and **Network Packet Capture** (Azure) gives full capture of network traffic (PCAP), but at a cost.

Detect

T1595.001, T1110, T1021
Although network flow data could detect brute force attacks, Azure and AWS security tooling such as GuardDuty (AWS) and Sentinel (Azure) can do that as well. Flow logs can be a great data source when you know what to look for, especially when you can pivot off an IP in an investigation. You can find open ports, uncover traffic patterns, or understand which part of your network was interacted with by a suspicious endpoint. Mycroft saw the initial intrusion by Culverton, then used flow logs to see what other systems he may have probed.



Container Orchestration

ECS/EKS (AWS) and **AKS/Container** (Azure) are orchestration services provided by the cloud service to make it easier to run containers and Kubernetes. The cloud has options to collect management API logs describing cluster activities, container deployments, and access configurations.

Detect

T1610, T1204.003, T1609, T1053.007, T1552.007, T1613, T1496
We commonly see attackers gaining access to a cloud environment and spinning up virtual machines or containers, to be backdoors or mine cryptocurrency for example. These logs could tell us when a suspicious container was spun up, what secrets may have been accessed, and if operations were manipulated.

Cloud Storage

S3 Logs through CloudTrail (AWS) and **Storage Access Logs** (Azure) record the Create/Read/Write/Deletes actions with data stored inside the storage container. It may be the only detection telemetry available to show what an attacker could have viewed or manipulated in these cloud-hosted storage services. However, they can generate a significant amount of data, so consider the tradeoff for how long to keep them. **S3 Access Logs** (AWS) will record web requests made to a bucket in a manner similar to web access logs.

Detect

T1619, T1530, T1537
Cloud Storage logs tracked Violet Norbury's collection of sensitive data. Mycroft knew which data was accessed and in the hands of Violet.

Cloud Proxy

Azure and AWS provide a number of **load balancers**, **content delivery networks**, and **multi-regional proxies** that help deliver network traffic in highly elastic network environments. The log content varies depending on where in the TCP/IP stack the services are proxying. AWS Network Load balancer captures access logs ONLY if a TLS listener is used, but AWS Application Load Balancer will capture access logs with a number of bytes, source and destination information, and user agent string.

Detect

T1595.001, T1110
Flow logs will show a brute force attack, but the source IP address is that of the proxy. The proxy logs tell the rest of the story, showing the originating IP address, and the real source of the attack.

Host Logs

Although we are focusing on cloud-specific logs, we should talk about logs generated by our hosts. **CloudWatch Agent** (AWS) and **Azure Monitor Agent** (Azure) provide detailed host metrics and forwards logs into the cloud's log management solution. **CloudWatch Insights** (AWS), **Microsoft Sentinel** (Azure), and **Log Analytic Workspace** (Azure) provides search and query tooling to detect and investigate.

Detect

Any detections leveraging host logs apply. This includes application-generated logs, such as web server logs showing a server-side request forgery attack.