DEFENDING AGAINST CRYPTOJACKING IN

aws

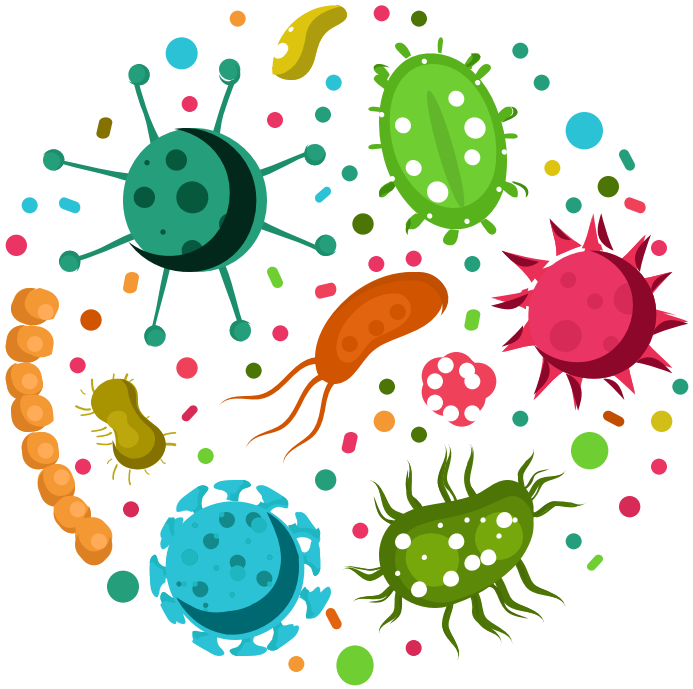
Azure

Google Cloud Platform

RedLock

TABLE OF CONTENTS

THE CRYPTOJACKING EPIDEMIC



Meet the internet's latest hacking menace - "cryptojacking". Hackers are using old tricks and new cryptocurrencies to turn stolen computing power into digital money. Cryptocurrency is "mined", or produced, by solving complex mathematical puzzles. Every millisecond matters, and the more computing power that's available to throw at the problems, the likelier the reward. Thus, having access to computing power is essential for anyone trying to make a lot of money from cryptocurrencies.

As the hype and soaring price of cryptocurrency has drawn in thousands of new players worldwide, generating a single bitcoin takes a lot more servers than it used to. It is becoming an arms race amongst miners for access to CPUs, GPUs and even electricity. As a result, we are starting to see a cryptojacking epidemic and hackers aren't sparing anyone; they are targeting everyone from consumers to large multinational organizations.

HACKERS AREN'T SPARING ANYONE

Some recent examples below highlight the expansion of cryptojacking attacks from consumers to business targets.

● business attacks ● consumer attacks

Oct 5, 2017:

Specifically, RedLock investigated cryptojacking attacks on organizations' public cloud environments and detected incidents at large multinational corporations such as Aviva and Gemalto.

Oct 30, 2017:

The Wannamine malware was detected which uses leaked NSA hacking tools to infiltrate computers at organizations and syphon processor power to crunch calculations

Jan 30, 2018:

RedLock uncovered a cryptojacking attack at Tesla resulting from a misconfiguration of its public cloud environment.

Jan 26, 2018:

Hackers had placed mining malware via Google DoubleClick ads on YouTube. These ads used visitors' compute power and electricity to mine cryptocurrencies for the hackers.

Feb 11, 2018:

4000 websites were infected with a compromised plugin called Browsealoud which used the compute power of unsuspecting website visitors to secretly generate cryptocurrency while racking up expensive electricity bills for the users.

PUBLIC CLOUD ENVIRONMENTS ARE PRIME TARGETS

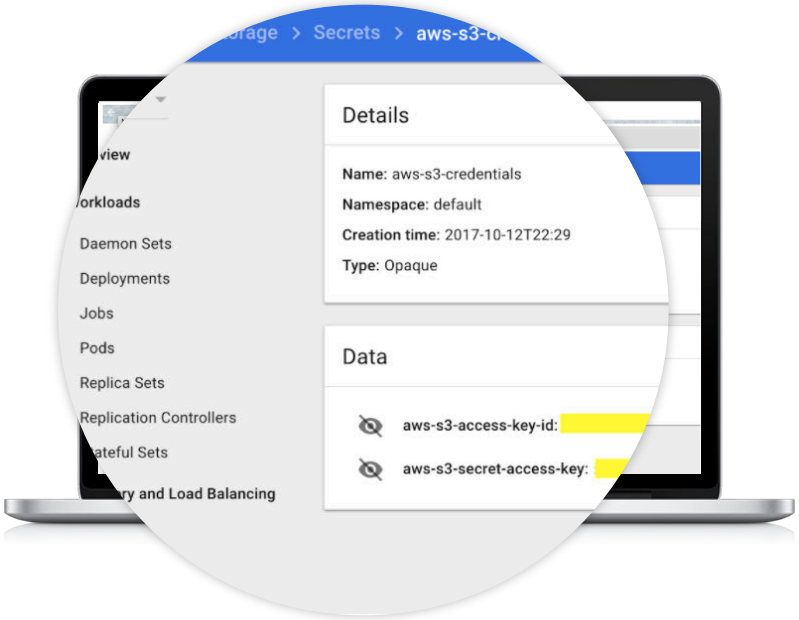
8% *of organizations had cryptojacking activity within their AWS, Azure, or Google Cloud environments."*

source: RedLock Cloud Security Trends - February 2018

Public cloud environments are elastic by nature and compute power can be increased on-demand. Moreover, organizations with large environments typically have remnant compute power and are unlikely to notice the unauthorized use of their compute power. This makes public cloud environments a very lucrative target for cryptojacking.

In RedLock's recent Cloud Security Trends - February 2018 report, it was revealed that **8% of organizations had cryptojacking** activity within their Amazon Web Services (AWS), Microsoft Azure, or Google Cloud environments. While this may not seem like a huge percentage today, it is anticipated this will rapidly increase as this technique gains popularity amongst the hacker community.

COMMON CLOUD SECURITY MISSTEPS



Tesla AWS Access Credentials Exposed

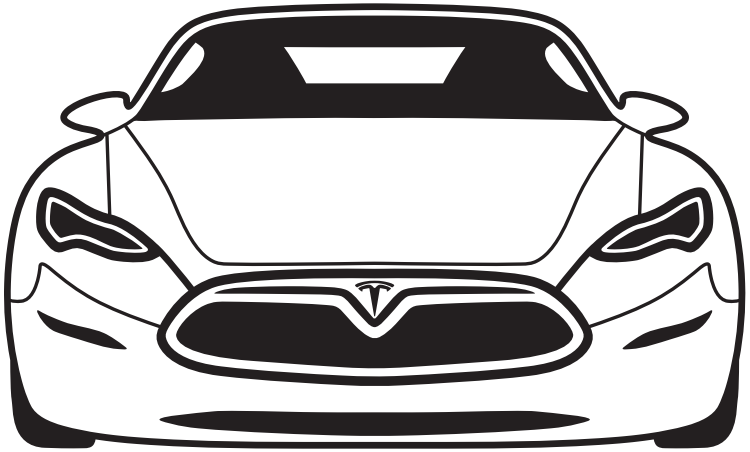
The RedLock CSI team identified some common traits amongst cryptojacking attacks on public cloud environments that were of particular interest:

- The incidents involved some kind of misconfiguration issue. More specifically, Tesla, Aviva, and Gemalto were using Kubernetes consoles that were not password protected, which provided an entry point to the environment. This is not totally surprising since in the cloud, many non-security users have elevated privileges and can easily make such mistakes.

- Upon further investigation, the team discovered that access credentials to the cloud environments were leaked within the misconfigured Kubernetes consoles which created additional exposures.

- In all cases, the nefarious network traffic from the mining operations went completely undetected. This suggests that there is inadequate network monitoring within public cloud environments.

LESSONS FROM THE CRYPTOJACKING ATTACK AT TESLA



In January 2018, the RedLock CSI team uncovered a cryptojacking attack at Tesla. While the attack was similar to the ones at Aviva and Gemalto, the team noted some sophisticated evasion measures that were employed in this attack:

- Unlike other cryptojacking incidents, the hackers did not use a well known public "mining pool" in this attack. Instead, they installed mining pool software and configured the malicious script to connect to an "unlisted" or semi-public endpoint. This makes it difficult for standard IP/domain-based threat intelligence feeds to detect the malicious activity.

- The hackers also hid the true IP address of the mining pool server behind CloudFlare, a free content delivery network (CDN) service. The hackers can use a new IP address on-demand by registering for free CDN services. This makes IP address-based detection of crypto mining activity even more challenging.

- Moreover, the mining software was configured to listen on a non-standard port which makes it hard to detect the malicious activity based on port traffic.

- Lastly, the team also observed on Tesla's Kubernetes dashboard that CPU usage was not very high. The hackers had most likely configured the mining software to keep the usage low to evade detection.

AREN'T AWS, AZURE AND GOOGLE CLOUD SECURE?

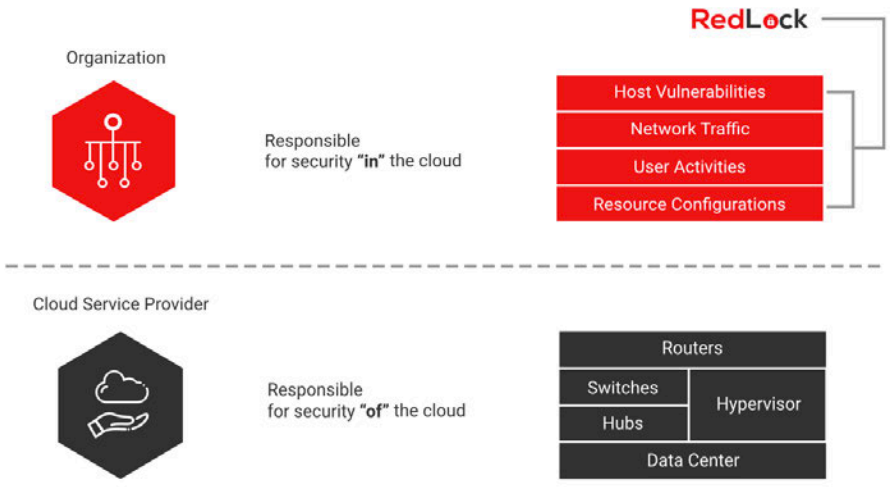


Figure 1: The Shared Responsibility Model

Worldwide spending on public cloud computing will increase from **$67B to $162B from 2015 through 2020** - growing at more than 6 times the rate of IT spending. So it's not surprising cybersecurity defenses haven't nearly kept pace with public cloud adoption, resulting in glaring security holes that lead to issues such as cryptojacking.

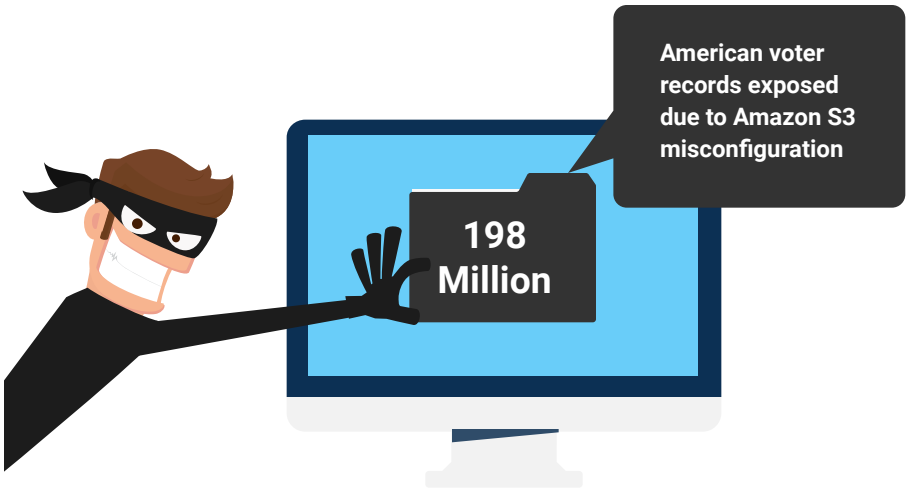
Cloud security is a shared responsibility between the cloud service provider (CSP) and the organization. CSPs are responsible for the managing the security of the cloud. Amazon, Microsoft and Google are doing their part, and none of the major breaches reported were caused by their negligence.

Organizations are responsible for monitoring their infrastructures for risky configurations, anomalous user activities, suspicious network traffic, and host vulnerabilities. Without that, anything the providers do will never be enough.

Organization Responsibility

#1

RISKY CONFIGURATION MONITORING



All of the cryptojacking incidents in public cloud environments involved some kind of misconfiguration issue. While the cloud enables agility by allowing users to create, modify, and scale storage, network and compute resources on-demand, this often occurs without any IT or security oversight. As a result, misconfigurations are likely and continuous configuration monitoring is essential. Manual monitoring and auditing of configurations is not really practical in public cloud environments where change is constant.

Open source and cloud service provider configuration monitoring tools weren't built for scale. They can be deployed to monitor individual cloud accounts which creates a siloed view of the cloud environment. Alternately, point configuration monitoring solutions can detect configuration drift but lack context on the environment, making it difficult to ascertain the severity of issues. For example, an open security group is a risky practice but not necessarily an indicator of compromise.

Effective configuration monitoring requires configuration data to be correlated with other security data sets such as user activities, network traffic, host vulnerabilities, host activities, and threat intelligence to provide context on the severity of an issue.

Organization
Responsibility

#2

SUSPICIOUS USER ACTIVITY DETECTION

Access credentials leakage was a common issue in all of the public cloud cryptojacking incidents, which created exposures. The damaging consequences of access credential leakage was recently illustrated by the breach at Uber where the personal information of **57 million** users was compromised. In that incident, a developer exposed credentials in GitHub to Uber's public cloud environment, and these were subsequently used to infiltrate the environment and access sensitive data.

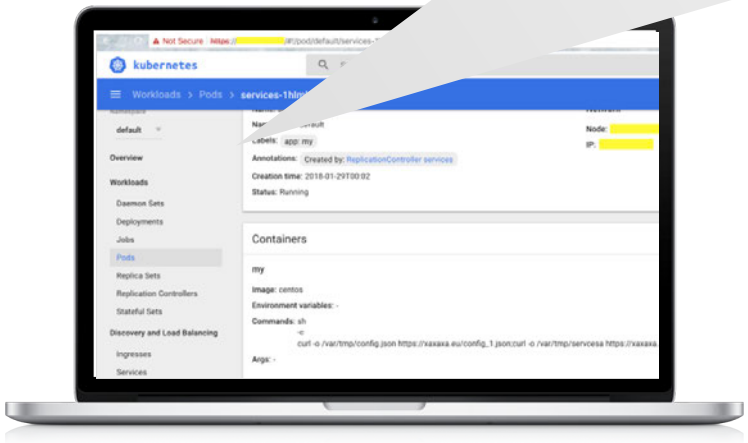
Since non-security users have elevated privileges in public cloud environments, organizations must operate under the assumption that accidental access credential exposures will occur and ultimately lead to account compromises. As a result, it is imperative to monitor environments for suspicious user activities.

Organizations may elect to build a custom user monitoring solution using SIEMs where they collect activity data across the environment and search for malicious patterns. This is a rather rudimentary approach and ineffective for detecting sophisticated insider threats or account compromises. A better approach requires an understanding of normal user activities and an automated way to detect anomalous behavior that goes beyond just identifying geo-location or time-based anomalies, but also event-based anomalies.

Organization Responsibility

#3

NETWORK TRAFFIC MONITORING



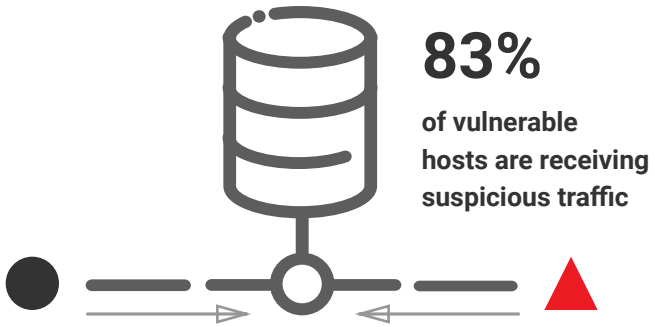
Crypto mining script running in Tesla's environment

The cryptojacking incidents illustrate the need for organizations to vigilantly monitor network traffic and detect suspicious activity. However, traditional network monitoring tools create security blind spots since they cannot be deployed for monitoring traffic to API-driven services in the cloud. A more effective approach involves collecting network traffic data from public cloud environments via APIs and correlating it with third party threat intelligence sources to derive context on risks and pinpoint nefarious activities.

Organization
Responsibility

#4

HOST VULNERABILITY MANAGEMENT



Unpatched hosts in cloud computing environments are just as vulnerable to attack as those in on-premise environments and the impact can be devastating as we saw in several recent high profile breaches. Vulnerable hosts can be used to penetrate environments and exfiltrate sensitive data or perform unauthorized activities such as crypto mining.

While standalone vulnerability management tools can be used in on-premise environments which are relatively static, they are ineffective in dynamic cloud environments. These tools perform periodic scans of an environment to identify hosts with missing patches based on IP address. However, cloud environments are constantly changing and IP addresses are elastic, which makes the results unreliable.

The key to effective vulnerability management in public cloud environments is having context. Security data from public cloud environments must be correlated with vulnerability data from third party vulnerability management tools.

TIPS FOR CLOUD THREAT DEFENSE

“The RedLock Cloud 360 platform provides Veeva with the tools it needs to continuously monitor our environment, quickly identify and respond to issues, and provide improved visibility on its compliance posture.”

David Tsao
Global Information Security Officer (CISO) at Veeva

Organizations are spending millions of dollars with cloud service providers like Amazon Web Services (AWS), Microsoft Azure, and Google Cloud. With decentralized adoption across organizations, dynamic nature of workloads, and limited monitoring tools, it can be extremely challenging to detect security and compliance risks. Given the immaturity of cloud threat defense programs today, it is anticipated cybercrime such as cryptojacking will increase in scale and velocity.

Download the Cloud Security Buyer’s Guide

Download the Cloud Security Buyer’s Guide to get 20+ tips based on the NIST Cybersecurity Framework and manage risks across your public cloud environment.

Download:
https://info.redlock.io/lp-nist-csf-cloud-security

Get a Demo

Get a demo of cloud threat defense using the RedLock Cloud 360™ Platform. You will learn how to address the following issues:

- Are there any resources with risky configurations?
- Have any accounts been compromised
- Is there any network intrusions or nefarious traffic such as cryptojacking?
- Are there unpatched hosts in your environment that are network exploitable?

Request Demo:
https://info.redlock.io/demo-request