Exploiting Containers (version 1.0)

**Cloud Service Label: PaaS, SaaS, IaaS**

Docker containers utilized in the cloud can be exploited by an adversary. Azure manages containers via Azure Container Services (ACS) and Amazon manages containers via Amazon Elastic Container Service (Amazon ECS). Depending on how the infrastructure is provisioned, exploiting this could provide privilege escalation, discovery, and exfiltration opportunities.

Examples

|  |  |
| --- | --- |
| **Name** | **Description** |
| DEF CON 27: Cloud Village Presentation | If docker.sock (which is normally open on port 2376) is exposed and you can interact with it, you can create a new container and mount the host’s filesystem (as root), run the container, and the adversary can have root privileges to the filesystem. BOtB, is an open source tool created to automate recon and exploit of containers. |
| Exposed Docker Port | Identifying an open port associated with Docker (2376 is the most common open port by default) can lead to finding running containers and images. Most containers run with default root privileges, so running *docker -H 192.168.1.7:2376 exec -it <container name> /bin/bash* will automatically give you root once logged in. |

Mitigations

|  |  |
| --- | --- |
| **Mitigation** | **Description** |
| Restrict Access to Specific Networks | Restricting the access that Containers have to specific networks will limit the availability of data if credentials are compromised. |
| Limit Access to Accounts | A user or process with read privileges to a registry also has the ability to list and pull any images in a registry. Make sure only those that need privileges have it. |
| Disable Unnecessary Ports and Services | Ensure ports and services that are publicly accessible are locked down and disabled if necessary. In the case of Docker, port 2376 is the ssl version of Dockerd socket should not be open to the Internet. |

Detection

This can be detected by monitoring the commands being run through the cloud API. Most users/processes should not be listing and pulling images in a registry. If this happens it should be investigated. Any type of suspicious account activity should be monitored and flagged. This can include unusual log on times, connecting from unexpected devices, password changes, as well as multiple log on attempts.

|  |  |
| --- | --- |
| **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* |

References

1. [Rhino Labs. (2019, August). Exploiting AWS ECR and ECS with the Cloud Container Attack Tool (CCAT). Retrieved September 12, 2019.](https://rhinosecuritylabs.com/aws/cloud-container-attack-tool/)

1. [Rhino Labs. (2019, September). Cloud Container Attack Tool (CCAT). Retrieved September 12, 2019.](https://github.com/RhinoSecurityLabs/ccat" \t "_blank)
2. [DEF CON 27 Cloud Village. (2019, December). Chris Le Roy - Build to Hack Hack to Build. Retrieved June 17, 2020.](http://1.https:/www.youtube.com/watch?v=1FB58EVWAOU)
3. https://docs.docker.com/engine/security/https Retrieved July 29, 2020.