

Understanding Cloud Attack Vectors

The IaaS & PaaS Perspective



About the Cloud Security Alliance

The Cloud Security Alliance (CSA) is a not-for-profit organization with a mission to promote the use of best practices for providing security assurance within the cloud computing industry. Furthermore, it provides education on the application of cloud computing and its role in securing all other forms of computing. The CSA is led by a broad coalition of industry practitioners, corporations, associations, and other key stakeholders. For further information, visit www.cloudsecurityalliance.org and follow us on Twitter@cloudsa.

About the CSA Israel Chapter

This document was created by the Israeli chapter of the Cloud Security Alliance (CSA). The CSA Israeli chapter was founded by security professionals united in a desire to promote responsible cloud adoption in the Israeli market while delivering useful knowledge and global best practices to the Israeli innovation scene. Visit our Facebook group at www.facebook.com/groups/789522244477928 for more details.

© 2023 Cloud Security Alliance – All Rights Reserved. You may download, store, display on your computer, view, print, and link to the Cloud Security Alliance at <https://cloudsecurityalliance.org> subject to the following: (a) the draft may be used solely for your personal, informational, non-commercial use; (b) the draft may not be modified or altered in any way; (c) the draft may not be redistributed; and (d) the trademark, copyright or other notices may not be removed. You may quote portions of the draft as permitted by the Fair Use provisions of the United States Copyright Act, provided that you attribute the portions to the Cloud Security Alliance.

Acknowledgments

Contributors

Dina Agafonov
Daniel Begimher
Tony Daskalo
Gidi Farkash
Moshe Ferber
Michael Roza
Yuval Segev
Omri Segev Moyal
Dana Tsymbberg
Zur Ulianitzky

Reviewers

Oren Elimelech
Eyal Estrin
Patrick Gaw
Chris Kirschke
Mauricio Mendoza Clavero
Venu Reddy
Eitan Satmary
Yuval Sinay
Peter van Eijk
Kobi Zvirsh

CSA Global Staff

Frank Guanco
Claire Lehnert
Stephen Lumpe

Table of Contents

Acknowledgments	3
Introduction	5
Purpose of This Document	5
Document Structure & Scope.....	5
Target Audience.....	6
IaaS and PaaS Cloud Attack Vectors Overview	6
1: Exploitable Workloads	8
2: Workloads with Excessive Permissions	11
3: Unsecured Keys, Credentials, and Application Secrets	13
4: Exploitable Authentication or Authorization	16
5: Unauthorized Access to Object Storage	19
6: Third-Party Cross-Environment/Account Access	23
7: Unsecured/Unencrypted Snapshots & Backups	26
8: Compromised Images	29
Final Thoughts	31
Further Reading	32

Introduction

Our knowledge of risks and threats to the cloud is growing and evolving. Working groups, such as CSA Top Threats and other organizations, contribute greatly to our knowledge and understanding of this subject. But while there are many risks and threats documented, with a wide range of business impacts, we see that many of them utilize a small number of attack vectors. And this is the topic of this research.

To perform the research, we initially reviewed a large number of recent IaaS/PaaS-related incidents and refined the details into the actual vector that was exploited. We used the CSA top cloud security threats examples, MITRE-related analysis, and previous research on cloud incidents, to analyze as many incidents as possible.

After we developed a full list of vectors, we involved a group of professionals experienced in cloud-based attacks. We used our collective experience to group the different individual vectors into a group of eight primary vectors that we found worked very well in various attack scenarios.

Purpose of This Document

The goal of this research is to shed light on common IaaS/PaaS attack vectors, list them out, and map them to relevant CSA research and other threat models. By reading this document, organizations will better understand the common attack vectors utilized in attacks against cloud-hosted applications and infrastructure, and where they should focus their controls and security efforts.

Document Structure & Scope

The document is made from eight attack vectors that can be found in IaaS/PaaS implementation; each vector chapter consists of two parts:

- The main part includes a definition of the vector, a description and how it can be exploited, key takeaways on how to avoid or mitigate the vector, and examples of how this vector was utilized in the past
- Maps of the attack vectors to CSA and non-CSA frameworks and research will provide more insights and relevant controls. The mappings include:
 - Map the attack vector to the relevant technique or mitigation in MITRE
 - Map to the shared responsibility model - map of the relevant attack vector to the responsible party: either the Cloud Service Provider (CSP), the Cloud Service Customer (CSC) or shared
 - Map the vector to the relevant domains in the CSA Security Guidance (version 4) to add more knowledge on the vector
 - Map to CSA Cloud Controls Matrix (CCM) version 4.0.X to identify controls relevant to the vector. The X version tag is marking new control mapping therefore our document is relevant to any 4.0 release
 - Map to STRIDE threat model to add more knowledge on the vector
 - Map to CSA top threats research document (Pandemic eleven) to help identify the risks and threats associated with the vector

Target Audience

The target audience for this document is:

- GRC professionals and auditors who are responsible for cloud security environments and interested in learning more on the actual vectors
- Security professionals, DevOps and DevSecOps professionals, software and security architects, and IT security who are building IaaS/PaaS environments and looking for assistance on where to invest in security efforts

IaaS and PaaS Cloud Attack Vectors Overview

Before diving into the details of attack vectors, it's important to understand the different cloud service models: IaaS and PaaS.

- **IaaS (Infrastructure as a Service):** In an IaaS environment, the customer has more control over the infrastructure and operating system. The attack surface is larger because customers are responsible for securing their virtual machines, storage, and networking. Attack vectors in IaaS environments typically target misconfigurations or vulnerabilities in the virtual machines, storage, and network security settings.
- **PaaS (Platform as a Service):** In a PaaS environment, the customer has more control over the software and application code, but less control over the infrastructure. The attack surface is similar to that of IaaS, but with an added focus on vulnerabilities in the application code and configuration settings.

A cyber attack vector is a path or means by which an attacker can gain access to a computer or network server in order to deliver a malicious outcome. It can include a range of tactics, such as malware, phishing, social engineering, and more, used to exploit the target system's weaknesses and vulnerabilities.

In a cloud environment, a cyber attack vector can manifest itself in several ways. For example, an attacker may use a phishing email to trick a user into clicking a malicious link or entering their login credentials. Alternatively, an attacker may exploit a vulnerability in a cloud service, such as misconfigured permissions, to gain access to sensitive data.

The difference between a cyber attack vector and a weakness or vulnerability lies in their nature. A weakness or vulnerability is a weakness of an asset or control that can be exploited by a threat. On the other hand, a cyber attack vector is a specific method or technique that an attacker uses to exploit these weaknesses or vulnerabilities.

For example, a vulnerability in a cloud service may be a misconfigured firewall that allows unauthorized access to the network. A cyber attack vector, in this case, could be a SQL injection attack, where an attacker uses a specially crafted SQL query to gain access to sensitive data on the network.

The difference between attack vectors at Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Platform as a Service (PaaS) lies in the level of control that customers have over their environment. While attack vectors exist on all levels of the cloud stack, the service model determines who is responsible for their mitigation.

This research is focused on attack vectors targeting IaaS and PaaS consumers. Attack vectors in IaaS environments typically target misconfigurations or vulnerabilities in the virtual machines, storage, and network security settings.

In a PaaS environment, the customer has more control over the software and application code, but less control over the infrastructure. The attack surface is similar to that of IaaS, but with an added focus on vulnerabilities in the application code and configuration settings.

The consumer of SaaS has little control over cloud attack vectors other than through user's credentials. The bulk of the mitigation of the attack vectors in this research would then be a responsibility of the SaaS provider. Of course, there are more attack vectors, such as exploiting vulnerabilities in a web application, that are also under the responsibility of the SaaS provider.

It's important to note that these attack vectors are not mutually exclusive, and attackers may use a combination of tactics to compromise a target. As such, customers need to follow security best practices and apply multiple layers of security controls to mitigate the risks of cyber attacks.

1: Exploitable Workloads

Definition

A workload is defined by CSA as a “unit of processing, which can be in a virtual machine, a container, or other abstraction” (CSA Security Guidance 4, 7.4). Exploitable workloads are an attack vector that details an attacker’s ability to exploit a workload’s vulnerabilities and gain initial footholds into the cloud environment. The vulnerability can be either well-known or zero-day. In both cases, if not mitigated, this kind of initial access into the cloud environment can increase the realization of the risk related to the attack vector. One of the more common implications of this attack method is leveraging this into running crypto miners or ransomware attacks. In other cases, advanced techniques are used to pivot in the environment using attached or stored cloud credentials to gain data access or perform privilege escalation and lateral movement within the cloud environment.

Description

Exploitable workloads refer to any virtual machine or container accessible to an internal or external attacker due to misconfiguration of the cloud networks. Accessibility is achieved if the workload has a public IP address or the attacker already has access to the internal environment. “Exploitable workload” means the asset is vulnerable to misconfigurations, has known Common Vulnerability and Exposures (CVEs), application vulnerabilities, and so on. Using the combination of both access and existing vulnerability, an attacker can successfully obtain access, control, or leverage the cloud workload to further their attack. **In this attack vector, the compromised asset can lead to the attacker achieving persistence, data access, or privilege escalation in the cloud environment. The attacker can then leverage this asset to strengthen access and control to perform lateral movements in search of additional assets.**

Key Takeaways

To harden the cloud environment, security should occur across all points of the attack path.

- **Network:** Limit network access as much as possible, preferably private subnets and using security groups and micro segmentation. Use IP restrictions to specifically designate Classless **Inter-Domain Routing** (CIDR) ranges and open access to selected ports instead of ranges that are too wide. Whenever possible, expose services using a security barrier, such as a Load balancer, API Gateways, and/or a Web application firewall (WAF).
- **Vulnerabilities:** Scan all workloads regularly to detect known vulnerabilities and fix them through suggested remediation. Conducting a security assessment dedicated to the organization’s cloud environment is also recommended to ensure the discovery of unknown risks. Keep in mind that different workloads require different vulnerability scanning tools (i.e., containers might require different tools over VMs). A vulnerability scan should be performed using automated tools (Software Composition Analysis tools to detect vulnerabilities in open source packages, SAST/DAST/IAST (depending on your application type) to detect vulnerabilities in code, and so on) and be embedded as part of the continuous integration/continuous delivery (CI/CD) process.

- **Identity and Access Management (IAM):** All access to resources by identities must be authenticated and authorized. Follow the principle of least privilege/need to know and provide the minimum required permissions to the workloads that allow them to perform their designated task and store application secrets in a secure location.
- **Application:** Ensure design of the applications is in accordance with accepted principles for secure development, including auto-patching.

Anecdotes and Examples










See the following anecdotes and examples:

- [New TeamTNT Cryptojacking Malware Targeting Kubernetes](#)
- [Atlassian Confluence Servers Hacked via Zero-Day Vulnerability](#)

Applicable MITRE ATT&CK for Enterprise TTP

See "Exploit Public-Facing Application," MITRE ID: [T1190](#).

Shared Responsibility Model and STRIDE Threat Modeling Map

Security Responsibility	STRIDE
 Customer	 Spoofing Identity
 Cloud Service Provider	 Tampering with data
 Shared	 Repudiation
	 Information Disclosure
	 Denial of service
	 Elevation of privilege

CSA CBK Security Guidance Version 4.0

Domain 4: Compliance and Audit Management

Domain 7: Infrastructure Security

Domain 11: Data Security and Encryption

CSA CCM Controls Version 4.0.X

AIS - Application and Interface Security

AIS-06 - Automated Secure Application Deployment

AIS-07 - Application Vulnerability Remediation

CCC - Change Control and Configuration Management

CCC-06 - Change Management Baseline

CCC-07 - Detection of Baseline Deviation

IVS - Infrastructure and Virtualization Security

ISV-04 - OS Hardening and Base Controls

TVM - Threat and Vulnerability Management

TVM-01 - Threats and Vulnerability Management Policy and Procedures

TVM-03 - Vulnerability Remediation Schedule

TVM-04 - Detection Updates

TVM-07 - Vulnerability Identification

TVM-08 - Vulnerability Prioritization

Mapping to CSA Top Threats

This attack vector is relevant to:

Security Issue 7: System Vulnerabilities

Security issue 9: Misconfiguration and Exploitation of Serverless and Containers Workloads

2: Workloads with Excessive Permissions

Definition

The following attack vector describes a workload with excessive permissions in a cloud environment. In cloud environments, workloads such as VMs or containers often receive an identity or a role to perform operations on the cloud infrastructure. A typical example is providing a role to a VM, allowing access to cloud storage. By following this vector, attackers with access to the workload can leverage their permissions and gain excessive permissions to the environment.

Description

A workload is a unit of processing, which can be in a virtual machine, a container, or other abstraction such as serverless or FaaS (Function as a service). Workloads are generally created to run several jobs or tasks, with different access patterns and authentication complexities, requiring different permissions for multiple services. These permissions are assigned to the workload as a policy or role. This complexity creates challenges in managing access to specific services and resources, leading to bad security practices such as granting excessive permissions.

This attack vector emphasizes the potential for the elevation of privileges. The attacker usually gains first access with low-level permissions, so access to the workload with excessive permissions can result in the elevation of privileges and the attacker gaining better persistence and the ability to create more damage.

Key Takeaways

- Always implement the principle of least privilege by giving the workloads the minimum permissions necessary to perform their assigned tasks.
- Prefer temporary access tokens instead of permanent permissions.
- Prefer customized policy and roles over pre-defined and general roles.
- Try to be more granular in your policies by using features, such as permission boundaries, and assume roles.
- Prevent the possibility of activating local user accounts with high privileges, especially when working with containers.
- Verify and audit the permissions of the workloads to ensure that they do not have excessive permissions.

Anecdotes and Examples










See the following anecdotes and examples:

- [Lessons learned from the Capital One breach](#)
- [The attack on ONUS – A real-life case of the Log4Shell vulnerability](#)

Applicable MITRE ATT&CK for Enterprise TTPs

See "Privileged Account Manager," at MITRE ID: [M1026](#).

Shared Responsibility Model and STRIDE Threat Modeling Map

Security Responsibility	STRIDE
 Customer	 Spoofing Identity
 Cloud Service Provider	 Tampering with data
 Shared	 Repudiation
	 Information Disclosure
	 Denial of service
	 Elevation of privilege

CSA CBK Security Guidance Version 4.0

Domain 6: Management Plane

Domain 7: Infrastructure Security

Domain 12: Identity, Entitlement, and Access Management

CSA CCM Controls Version 4.0.X

CCC - Change Control and Configuration Management

CCC-06 - Change Management Baseline

CCC-07 - Detection of Baseline Deviation

IAM - Identity & Access Management

IAM-01 - Identity and Access Management Policy and Procedures

IAM-05 - Least Privilege

IAM-06 - User Access Provisioning

Mapping to CSA Top Threats

This attack vector is relevant to:

Security Issue 1: Insufficient Identity, Credential, Access, and Key Mgt, Privileged Accounts

Security issue 2: Insecure Interfaces & API

Security Issue 3: Misconfiguration and Inadequate Change Control

3: Unsecured Keys, Credentials, and Application Secrets

Definition

This attack vector details the existence of cleartext credentials or unprotected credentials on a cloud workload, service or code repository. Credentials can be of different types and located in various places. The popular vectors include IAM access or API keys embedded inside a configuration file, template, or actual code. Or SSH keys embedded into an image or workload. These types of credentials are known as secrets.

Description

Cloud services interact among themselves and/or with external services. This interaction requires a set of permissions, so a mechanism of authentication and authorization is involved. This mechanism is activated using API or Access keys that authenticate the consuming service or workload. For example, a known use case is when an EC2 instance needs access to an S3 bucket to store or retrieve data, or a CI service requires an API key to authenticate to an external code repository.

Another common scenario is using SSH keys to manage a fleet of virtual machines leading to challenges in managing SSH key pairs. Due to the complexity of managing access keys, organizations use the same keys for multiple compute instances. As a result, an attacker that compromises one server can access all servers that use the same SSH key pair. This is an easy way to move inside the cloud environment, gather more information, and search for higher permissions.

Many organizations fail to establish clear policies for secret life cycle management (generate, store, retrieve, rotate, and decommission), resulting in unauthorized access to resources, the ability to perform lateral movement, and the elevation of privileges in the environment.

Key Takeaways

The recommended best practices for storing and using API keys, secrets, passwords, SSH keys, or certificates depend on the actual access scenario. Here are a few examples:

- Cloud provider workload accessing the same provider service: The recommended way to grant this access to the workload is by attaching it to an identity with the required permissions. This will eliminate the need for static access keys and utilize dynamically assigned keys. Examples of this type of solution are AWS STS, GCP OICD, or Azure SAS.
- Application components interacting among themselves, or with cloud external services, can store secrets in designated, secure storage. Examples are AWS Secrets Manager, Azure Key Vault, or GCP Secret Manager.
- For SSH keys, secured bastion hosts ("jump box") with a one-time SSH key, or an IAM-based auth solution (i.e., AWS Session Manager, Azure Bastion, or Google Identity-Aware Proxy)

can be used to maximize security. Some enterprise solutions also support MFA alongside the SSH authentication mechanism. It is also recommended to use different SSH keys for different environments and classifications.

- Follow secure procedure for any key lifecycle from generation to revocation.
- Store keys in a designated enterprise service (such as AWS Secrets Manager, Azure Key Vault, or GCP Secret Manager).
- Intelligently manage the public distribution process of open source solutions, such as containers publish process to public container registry or code libraries that the organization develops and shares in common open source repositories.
- Monitor access logs to detect any suspicious activity related to access keys and certificates.

To detect cleartext secrets on different locations, we recommend regularly scanning cloud workload configuration, Infrastructure as Code templates, and code repositories to search for static credentials and keys/secrets.

Anecdotes and Examples










See the following anecdotes and examples:

- [CircleCI says hackers stole encryption keys and customers' secrets](#)
- [Howebrew Security Incident Disclosure](#)
- [Samsung spilled SmartThings app source code and secret keys](#)
- [website Animal Jam breached after miscreants spot private AWS key](#)
- [GotRoot! AWS root Account Takeover](#)

Applicable MITRE ATT&CK for Enterprise TTPs

See "Unsecured Credentials," at MITRE ID: [T1552](#).

Shared Responsibility Model & STRIDE Threat Modeling Map

Security Responsibility	STRIDE
 Customer	 Spoofing Identity
 Cloud Service Provider	 Tampering with data
 Shared	 Repudiation
	 Information Disclosure
	 Denial of service
	 Elevation of privilege

CSA CBK Security Guidance Version 4.0

Domain 6: Management Plane

Domain 7: Infrastructure Security

Domain 10: Application Security

Domain 12: Identity, Entitlement, and Access Management

CSA CCM Controls Version 4.0.X

AIS - Application and Interface Security

AIS-06- Automated Secure Application Deployment

CCC - Change Control and Configuration Management

CCC-03 - Change Management Technology

CCC-07 - Detection of Baseline Deviation

CEK - Cryptography, Encryption & Key Management

CEK-21 - Key Inventory Management

IAM - Identity & Access Management

IAM-06 - User Access Provisioning

IAM-13 Uniquely Identifiable Users

IAM-15 Password Management

IPY - Interoperability & Portability

IPY-02 Application Interface Availability

Mapping to CSA Top Threats

This attack vector is relevant to:

Security Issue 1: Insufficient Identity, Credential, Access, and Key Mgt, Privileged Accounts

Security issue 2: Insecure interfaces & API

Security Issue 3: Misconfiguration and Inadequate Change Control

Security Issue 11: Cloud Storage Data Exfiltration

4: Exploitable Authentication or Authorization

Definition

This attack vector details an entity's improper management and authentication, such as user, workload, function, role, group, and so on. Each identity should have security controls and be properly maintained. A neglected identity, one that is not in use for a long time or in no use at all and has excessive permissions can lead to an undetected breach that compromises this identity.

Description

It is crucial to maintain a healthy security environment. Although this is not an easy task, it is important to pay attention to IAM best practices, as it is usually the first thing attackers will examine and try to exploit.

Improper management comes in many forms:

- **Using default password.** Some tools are installed with the default password, and administrators fail to change the default password, resulting in security exposure.
- **Weak password policies.** Without defining a strong password policy, attackers can easily guess passwords using well-known techniques (dictionary attacks, brute force, etc.) and gain access to the target user's identity to log in to its cloud environment.
- **Empty groups.** Groups are aggregated entities usually created for users. In the cloud environment, permissions can be assigned to the group and granted to users. Those permissions are also granted to any users in the group. New users added to the group will also be granted the existing permissions of this group. Therefore, having a group without assigned users imposes a risk on the environment if an attacker gets access to IAM and ability to re-populate this group.
- **Excessive permissions.** Granting specific permissions to resources can be a tiresome task, this is why it is very common to see users/groups with excessive permissions with no legitimate reason. This security issue can lead to gaining access to sensitive resources and performing actions that can lead to environmental compromise.
- **Not enforcing MFA.** Multifactor Authentication (MFA) provides an additional protection layer for users. When MFA is not enabled, and the authentication is performed with static credentials (i.e., username and password), attackers can easily compromise the environment once they gain access to those credentials, such as by phishing. The preferred method is using authenticator apps like Google Authenticator.
- **Inactive identities.** Activities for inactive identities/users can be tracked using audit logs and last login records. Those identities are considered neglected identities that can pose a risk to the environment because, usually, they are not monitored or maintained currently.
- **No Logging.** The origination does not log AAA operations resulting in inability to detect malicious use.
- **Misconfigured IAM Trust Policy.** Identity and Access Management trust policies to external identities and cross-account access that are misconfigured may be enumerated and lead to initial access to the victim's cloud account.

Key Takeaways

To proactively prevent this kind of attack path, organizations should:

- **Enable MFA.** MFA should be mandatory for all users and all applications.
- **Create strong password policies.** Ensure user passwords won't be compromised easily. In addition to the standard password policies complexity (length, uppercase, lowercase alphabet, numerical and non-alphanumeric characters), administrators should also set password expirations and configure the settings so that expired passwords require administrator resets and prevent password reuse.
- **Temporary access.** Use temporary access (such as AWS Assume Role or Azure Just-in-Time access) instead of static credentials/permissions.
- **Reevaluate permissions.** Audit and monitor policies and make sure to follow the concept of least privileges.
- **Empty groups.** If there are groups without users, delete the group.
- **Inactive identities.** Monitor identity activities using audit logs and last login records. If an identity looks inactive for a defined period (as determined by the organization), it is recommended to delete it. Ensure accounts are removed as part of off-boarding processes.
- **Enforcing Logging.** Enforcing logging and audit on any AAA operations in the cloud and application level.

Anecdotes and Examples










See the following anecdotes and examples:

- [Hacker Puts Hosting Service Code Spaces Out of Business](#)
- [Admin Accounts With No Passwords at the Heart of Recent MongoDB Ransom Attacks](#)
- [Equifax used the word 'admin' for the login and password of a database](#)

Applicable MITRE ATT&CK for Enterprise TTPs

See "Valid Accounts: Cloud Accounts," at MITRE ID: [T1078](#).

Shared Responsibility Model & STRIDE Threat Modeling Map

Security Responsibility	STRIDE
 Customer	 Spoofing Identity
 Cloud Service Provider	 Tampering with data
 Shared	 Repudiation
	 Information Disclosure
	 Denial of service
	 Elevation of privilege

CSA CBK Security Guidance Version 4.0

Domain 2: Governance and Enterprise Risk Management

Domain 4: Compliance and Audit Management

Domain 5: Information Governance

Domain 6: Management Plane and Business Continuity

Domain 12: Identity, Entitlement, and Access Management

CSA CCM Controls Version 4.0.X

AIS - Application and Interface Security

AIS-01: Application and Interface Security Policy and Procedures

AIS-02: Application Security Baseline Requirements

AIS-03: Application Security Metrics

IAM - Identity and Access Management

IAM-01: Identity and Access Management Policy and Procedures

IAM-02: Strong Password Policy and Procedures

IAM-03: Identity Inventory

IAM-14: Strong Authentication

LOG - Logging and Monitoring

LOG-01: Logging and Monitoring Policy and Procedures

Mapping to CSA Top Threats

This attack vector is relevant to:

Security Issue 1: Insufficient Identity, Credential, Access, and Key Mgt, Privileged Accounts

Security Issue 2: Insecure Interfaces & API

5: Unauthorized Access to Object Storage

Definition

Object storage is one of the most common services in the cloud. Because of its wide-spread use, it is also a known attack vector. This attack vector details the existence of cloud-hosted storage with public objects that don't require user authentication or authorization, usually by mistake. With no authorization, attackers can exploit it to read and/or write data using common tools to compromise the stored data's availability, integrity, or confidentiality.

Description

Cloud object storage is a key factor in any deployment, most applications need some form of storage. Suppose those cloud storages are misconfigured to be publicly reachable without additional authorization. In that case, attackers can use readily available utilities to compromise the datastore and/or the data within, with a high likelihood of attacking without detection.

There are several main security configurations on which object storage services are built:

- **Public/Private.** When discussing public cloud object storage, the differentiation between public and private refers to both network access (via an endpoint or a hostname) and additional identity controls, such as allowing anonymous users to read and/or write from the storage. When creating a storage component in object storage, the user can choose if the storage will be public or private. This configuration controls the access settings of the storage. When access is public, the storage does not require authorization, and any entity can access it without special restrictions.
- **Encryption.** Post-incident response reports show that many cloud storage components were public and unencrypted at rest. Cloud-native encryption typically requires additional permissions to use the encryption keys, which can prevent the data itself from being compromised in some cases.
- **Authentication.** Some public cloud object storage offers multiple configuration options for the authentication to the resource. Mostly, there are options for no authentication or basic authentication (username and password). Other advanced options include Security Assertion Markup Language (SAML) or OpenID connect (OIDC), or cloud-native IAM-based authentication. The latter authentication methods are more hardened against external attacks than the former.
- In addition to these security configurations, there are other preventative measures that can be implemented:
- **Security Awareness Training.** Training users on object storage security awareness is crucial to maintaining the confidentiality, integrity, and availability of sensitive data stored in the cloud. Training can also help users recognize phishing attempts and other social engineering tactics that cybercriminals use to trick users into revealing sensitive information or clicking on malicious links. Ultimately, a well-trained user base can serve as a first line of defense

against cyber threats and help ensure the security of an organization's data.

- **Security Scanning.** Security scanning of object storage can be an effective tool for identifying misconfigurations that could leave data vulnerable to attack. By scanning object storage for misconfigured buckets, publicly accessible data, or improperly set access controls, organizations can proactively detect and remediate security issues before they can be exploited by cybercriminals. Regular security scans can be integrated into an organization's security program to provide ongoing visibility into the security posture of their object storage and help ensure the continued protection of their data.

Key Takeaways

To proactively secure the cloud environment from unauthorized access to object storage attack vectors, the organization can:

- **Keep all object storage private.** If the object storage is not intended to be public or contains any data that is not supposed to be public, the best practice is to change the settings of the storage asset to be private.
- **Use a secure sharing process.** If the data needs to be shared with external parties, solutions such as a pre-signed URL, AWS STS or Azure SAS for providing temporary access.
- **Network security controls.** Keep all networks to the object storage within your private subnets and the CSP backbone instead of traversing over the public Internet, using services such as private endpoints.
- **Use encryption keys.** Ensure all object storage resources are encrypted in transit and at rest. Prefer to use customer-managed encryption keys and rotate the keys according to predefined policy (at least once a year). Manage access to the encryption keys using the CSP identity and access management mechanism.
- **Avoid using basic or no-authentication.** While additional resources and services may be required, it is recommended to use a cloud-native IAM-based authentication or another open standard such as SAML, OIDC, etc.
- **Ensure a thorough understanding of the shared responsibility model** - This involves understanding the division of security responsibilities between the cloud service provider and the organization using the cloud service, and identifying which security measures are the responsibility of each party. By understanding this model and implementing the appropriate security measures, organizations can help ensure the security of their object storage in the cloud.

Anecdotes and Examples










See the following anecdotes and examples:

- [How a Misconfigured Storage Bucket Exposed Medical Data](#)
- [Pfizer suffers huge data breach on unsecured cloud storage](#)
- [Millions of Verizon Customer Records Exposed through Open Amazon S3 Bucket](#)

Applicable MITRE ATT&CK for Enterprise TTPs

See "Restrict File and Directory Permissions," at MITRE ID: [M1022](#).

Shared Responsibility Model & STRIDE Threat Modeling Map

Security Responsibility	STRIDE
 Customer	 Spoofing Identity
 Cloud Service Provider	 Tampering with data
 Shared	 Repudiation
	 Information Disclosure
	 Denial of service
	 Elevation of privilege

CSA CBK Security Guidance Version 4.0

Domain 5 - Information Governance

Domain 7 - Infrastructure Security

Domain 12 - Identity, Entitlement and Access Management

CSA CCM Controls Version 4.0.X

A&A - Audit & Assurance

A&A-06: Remediation

AIS - Application and Interface Security

AIS-01: Application and Interface Security Policy and Procedures

AIS-02: Application Security Baseline Requirements

AIS-04: Secure Application Design and Development

CCC - Change Control and Configuration Management

CCC-04: Unauthorized Change Protection

CCC-06: Change Management Baseline

CCC-07: Detection of Baseline Deviation

DSP - Data Security and Privacy Lifecycle Management

DSP-01: Security and Privacy Policy and Procedures

DSP-07: Data Protection By Design and Default

DSP-12: Limitation of Purpose in Personal Data Processing

DSP-13: Personal Data Sub Processing

DSP-17: Sensitive Data Protection

HRS: Human Resources
HRS-03: Clean Desk Policy and Procedures
HRS-12: Personal and Sensitive Data Awareness and Training
IAM: Identity And Access Management
IAM-05: Least Privilege
IAM-06: User Access Provisioning
STA-03: SSRM Guidance
STA-04: SSRM Control Ownership
STA-06: SSRM Control Implementation
UEM - Universal Endpoint Management
UEM-11: Data Loss Prevention

Mapping to CSA Top Threats

This attack vector is relevant to:

Security Issue 8: Accidental Data Disclosure
Security Issue 11: Cloud Storage Data Exfiltration

6: Third-Party Cross-Environment/ Account Access

Definition

This attack vector details the trust given to a third-party entity or resource to access the customer's cloud environment. In this attack vector, the permissions given are overly permissive and can lead to account takeover from the third-party company. In some cases, the granted permissions are not powerful enough but might be abused in order to gain additional permissions or identities which can lead to account takeover.

Description

Organizations may rely on third-party vendors and managed service providers to support, monitor, or secure their environment. There are multiple ways for this access to take place: by API, IAM Role, VPC peering, agent software, or specific VM or container that is residing in the customer environment but controlled by a third party. If the third party is compromised, it exposes the company to certain risks.

Key Takeaways

- Choose your third-party providers wisely, after executing a detailed security assessment making sure the vendors are in the required security level
- Follow the principle of least privilege by providing minimum required permissions, and have a process in place to review and reduce permissions
- If possible (per Legal department approval) delete every identity and third-party accounts once they are not in use
- Ensure MFA is configured for external users
- Use technologies such as AWS STS, Azure Privileged Identity Management or Azure Just-in-Time access for providing access to external support services
- Audit all actions done by all identities and make sure there are logging mechanisms in place to detect anomalous behavior
- Ensure security mechanisms are in place to prevent unauthorized log in activities, for example, external ID for assuming IAM roles
- Periodically map the shared resources, and analyze the risk and the derived meanings.

Anecdotes and Examples










See the following anecdotes and examples:

- [Customer Guidance on Recent Nation-State Cyber Attacks](#)
- [CircleCI warns customers to rotate 'any and all secrets' after hack](#)
- [criminal actor targeting organizations for data exfiltration and destruction](#)
- [LiveAuctioneers Security Breach](#)

Applicable MITRE ATT&CK for Enterprise TTPs

- [Valid Accounts](#)
- [Valid Accounts: Cloud Accounts](#)
- [Steal Application Access Token](#)
- [Account Manipulation](#)
- [Exploitation for Privilege Escalation](#)
- [Trusted Relationship](#)

Shared Responsibility Model & STRIDE Threat Modeling Map

Security Responsibility	STRIDE
 Customer	 Spoofing Identity
 Cloud Service Provider	 Tampering with data
 Shared	 Repudiation
	 Information Disclosure
	 Denial of service
	 Elevation of privilege

CSA CBK Security Guidance Version 4.0

Domain 2: Governance and Enterprise Risk Management

Domain 4: Compliance and Audit Management

Domain 6: Management Plane and Business Continuity

Domain 7: Infrastructure Security

Domain 12: Identity, Entitlement and Access Management

CSA CCM Controls Version 4.0.X

IAM-01 Identity and Access Management Policy and Procedures
IAM-02 Strong Password Policy and Procedures
IAM-03 Identity Inventory
IAM-04 Separation of Duties
IAM-05 Least Privilege
IAM-06 User Access Provisioning
IAM-07 User Access Changes and Revocation
IAM-08 User Access Review
IAM-09 Segregation of Privileged Access Roles
IAM-10 Management of Privileged Access Roles
IAM-11 CSCs Approval for Agreed Privileged Access Roles
IAM-12 Safeguard Logs Integrity
IAM-13 Uniquely Identifiable Users
IAM-14 Strong Authentication
IAM-15 Passwords Management
IAM-16 Authorization Mechanisms
CCC-04 Unauthorized Change Protection
DSP-07 Data Protection by Design and Default
HRS-10 Non-Disclosure Agreements
LOG-01 Logging and Monitoring Policy and Procedures
LOG-02 Audit Logs Protection
LOG-03 Security Monitoring and Alerting
LOG-04 Audit Logs Access and Accountability
LOG-05 Audit Logs Monitoring and Response
LOG-06 Clock Synchronization
LOG-07 Logging Scope
LOG-08 Log Records
LOG-09 Log Protection
LOG-11 Transaction/Activity Logging
LOG-12 Access Control Logs
LOG-13 Failures and Anomalies Reporting

Mapping to CSA Top Threats

This attack vector is relevant to:

Security Issue 1: Insufficient Identity, Credential, Access, and Key Mgt, Privileged Accounts
Security Issue 3: Misconfiguration and Inadequate Change Control
Security Issue 4: Lack of Cloud Security Architecture and Strategy
Security Issue 6: Unsecure Third-Party Resources
Security Issue 11: Cloud Storage Data Exfiltration

7: Unsecured/Unencrypted Snapshots & Backups

Definition

This attack vector refers to the presence of unsecured or unencrypted snapshots or backups on a cloud platform or service. These snapshots/backups may contain sensitive information, such as passwords, personal data, or confidential business information. They can be accessed by unauthorized parties if not properly secured with encryption or other security measures. Properly secured, for this matter, means the same level of the original data is secured.

These snapshots/backups can be stored in various locations, and be used to restore data in the event of data loss or other disruptions, so having access to it and maintaining the level of integrity are also important.

This document will use the phrase “unsecured backup” as an unsecured or unencrypted snapshot or backup.

Description

Several potential vector attacks are using unsecured cloud backups, including:

1. **Elevation of privileges:** Attackers may use unsecured backups to move laterally or gain excessive permissions (using the backup to locate keys and passwords).
2. **Enhancement to ransomware:** Attackers may delete unsecured backups to prevent the organization from recovering from a ransomware attack.
3. **Misconfigured permissions:** If backups are not properly configured with the appropriate permissions, they may be accessible to unauthorized parties, who can then access and manipulate sensitive data.
4. **Malicious code injection:** Attackers may use unsecured backups as a means of injecting malicious code into systems or networks, which can result in data loss, system disruption, and other negative impacts.

Key Takeaways

Here are several recommendations, and best practices for preventing potential vector attacks using unsecured backups as follows:

- Apply data minimization (via data retention policy) also on your cloud backup, removing unnecessary information, thus reducing the risk.
- Use strong, unique passwords for all cloud backup accounts and regularly update them to prevent unauthorized access.
- Enable MFA for cloud backup human accounts and restrict access of service accounts to

- known traffic sources, to add an extra layer of security.
- Use encryption for all cloud backups and snapshots to protect sensitive data from being accessed by unauthorized parties
 - Use customer-managed encryption keys, and store them in a secured vault (such as AWS KMS, Azure Key Vault, or Google Cloud KMS).
 - Quarterly review and update permissions for all cloud backups and snapshots to ensure that only authorized identities can access them.
 - Implement a backup and recovery plan to ensure that data is properly backed up and easily restored during data loss or other disruptions.
 - Regularly review and update security measures to ensure they effectively protect against potential threats to the backups and snapshots.
 - Keep backups and snapshots in an archive tier or another cloud provider, with proper access management, store them in immutable form, and monitor any access to them.

Anecdotes and Examples










See the following anecdotes and examples:

- [Bonobos clothing store suffers a data breach](#)
- [Finding Secrets In Publicly Exposed Ebs Volumes](#)
- [Loot Public EBS Snapshots](#)

Applicable MITRE ATT&CK for Enterprise TTPs

See "Data Backup," at MITRE ID: [M1053](#).

Shared Responsibility Model & STRIDE Threat Modeling Map

Security Responsibility	STRIDE
 Customer	 Spoofing Identity
 Cloud Service Provider	 Tampering with data
 Shared	 Repudiation
	 Information Disclosure
	 Denial of service
	 Elevation of privilege

CSA CBK Security Guidance Version 4.0

Domain 5: Information Governance

Domain 7: Infrastructure Security

Domain 11: Data Security and Encryption

Domain 12: Identity, Entitlement and Access Management

CSA CCM Controls Version 4.0.X

BCR-08: Backup

DSP -Data Security and Privacy Lifecycle Management

DSP-01: Security and Privacy Policy and Procedures

DSP-07: Data Protection By Design and Default

DSP-12: Limitation of Purpose in Personal Data Processing

DSP-13: Personal Data Sub Processing

DSP-17: Sensitive Data Protection

HRS-12: Personal and Sensitive Data Awareness and Training

IAM Identity and Access Management

IAM-01: Identity and Access Management Policy and Procedures

IAM-02: Strong Password Policy and Procedures

IAM-03: Identity Inventory

IAM-05: Least Privilege

IAM-14: Strong Authentication

LOG - Logging and Monitoring

LOG-01: Logging and Monitoring Policy and Procedures

UEM-11: Data Loss Prevention

Mapping to CSA Top Threats

This attack vector is relevant to:

Security Issue 8: Accidental Cloud Data Disclosure

Security Issue 11: Cloud Storage Data Exfiltration

8: Compromised Images

Definition

Images are files (or sets of files) that provide the initial installation files for a workload. This vector describes images that have been maliciously modified to exploit vulnerabilities and allow attackers to gain access to cloud resources. This is usually done by creating a back-door in the image or obfuscating malware inside the image.

Description

It is important for cloud users to understand the potential risks associated with VM/Container images and to take precautions against cloud attacks originating from compromised VM/Container images. Images that have been compromised can be used to launch cloud-based attacks, such as cloud malware injection, mining cryptocurrency, data exfiltration, or account takeover.

The source of the compromised images can be internal, images that were created by the cloud customer from untrusted sources or accessed later on by a malicious threat actor; or external, malicious images that came from image stores, public image repositories, or even from the official marketplace that due to lack of governance allowed the malicious content to be distributed.

Key Takeaways

Safeguards against compromised VM/Container images include using images from trusted sources only, monitoring the access to the image store and verifying image integrity, alerting for changes or modifications, ensuring VM/Container images are up-to-date with the most recent security patches, and performing regular scans of VM/Container images. In addition, cloud administrators should implement policy regarding the usage of open source software and images, design access control policies and use cloud-agnostic cloud management tools to manage cloud deployments. By taking these additional precautions, cloud administrators can protect deployments from attack vectors like compromised images. In addition, cloud security administrators should consider implementing cloud-native container security solutions to bolster protections against compromised images and other cloud attack vectors. Kubernetes network security policies, cloud-native security solutions, and cloud application firewalls can further strengthen cloud security. Utilizing cloud security tools allows administrators to respond quickly to any risks or threats to the cloud by detecting malicious activity linked to compromised images.

Anecdotes and Examples










See the following anecdotes and examples:

- [CodeCov Kills Off Bash Uploader Blamed for Supply Chain Hack](#)
- [Docker Hub repositories hide over 1,650 malicious containers](#)
- [An AWS Virtual Machine Is Infected With Mining Malware. There Could Be Others](#)
- [Analysis on Docker Hub malicious images: Attacks through public container images – Sysdig](#)

Applicable MITRE ATT&CK for Enterprise TTPs

See "Building Image on Host," Mitre ID: [T1612](#).

Shared Responsibility Model & STRIDE Threat Modeling Map

Security Responsibility	STRIDE
 Customer	 Spoofing Identity
 Cloud Service Provider	 Tampering with data
 Shared	 Repudiation
	 Information Disclosure
	 Denial of service
	 Elevation of privilege

CSA CBK Security Guidance Version 4.0

- Domain 7 - Infrastructure Security
- Domain 8 - Virtualization and Containers

CSA CCM Controls Version 4.0.X

Infrastructure & Virtualization Security IVS

ISV-01 Infrastructure and Virtualization Security Policy and Procedures

ISV-04 Harden host and guest OS, hypervisor or infrastructure control plane according to their respective best practices, and supported by technical controls, as part of a security baseline

Supply Chain Management, Transparency, and Accountability - STA

STA01 SSRM Policy and Procedures

STA07 Develop and maintain an inventory of all supply chain relationships

Mapping to CSA Top Threats

This attack vector is relevant to:

Security issue 4: Lack of cloud security architecture and strategy

Security Issue 6: Use insecure third-party resources

Security Issue 9: Misconfiguration and exploitation of Serverless and containers workloads

Final Thoughts

In conclusion, the increasing adoption of cloud computing has led to a significant rise in cyber threats and attacks. However, while the number and significance of risks, threats, and vulnerabilities has risen, the attack vectors that are being used remain relatively steady. Our goal in this paper is to direct attention to those vectors.

Another interesting observation that we came across while working on the research is the diversity of the attack techniques. Some attack vectors are old, well-known, and not cloud-specific (e.g., vulnerable VM). Still, some are utilizing the new features of the cloud (e.g., cross-account attacks).

While exploring the document, remember that the same attack vectors can produce different risks, or business impact. For example, broken permissions around object storage can result in loss of file availability (e.g., ransomware encryption). It can also result in confidentiality (e.g., data leaked) or integrity risks (e.g., changing the content of files). Each of those risks requires different mitigation. However, they are all related directly to the same vector.

Overall, as cloud computing continues to evolve, organizations must prioritize cloud security and stay vigilant against potential threats and attacks. By understanding the relationship between the risks and the attack vector used, cloud security professionals can better understand where to focus their efforts.

Further Reading

- NIST SP 800-161 - Supply Chain Risk Management Practices for Federal Information Systems and Organizations
<https://csrc.nist.gov/publications/detail/sp/800-161/rev-1/final>
- NSIT SP 800-218 - Secure Software Development Framework (SSDF) Version 1.1 Recommendations for Mitigating the Risk of Software Vulnerabilities
<https://csrc.nist.gov/publications/detail/sp/800-218/final>
- Strengthening the Connection: VERIS and MITRE ATT&CK®
<https://medium.com/mitre-engenuity/strengthening-the-connection-veris-and-mitre-att-ck-c3aac3fa9cd>
- CSA Cloud Controls Matrix
<https://cloudsecurityalliance.org/research/cloud-controls-matrix/>
- Microsoft STRIDE threat modeling
<https://learn.microsoft.com/en-us/azure/security/develop/threat-modeling-tool-threats>
- Ramimac - AWS-customer-security-incidents at Github
<https://github.com/ramimac/aws-customer-security-incidents>
- AWS Security Incident Response Guide
<https://docs.aws.amazon.com/whitepapers/latest/aws-security-incident-response-guide/aws-security-incident-response-guide.html>
- Microsoft cloud security benchmark documentation
<https://learn.microsoft.com/en-us/security/benchmark/azure/>
- AWS Well-Architected Framework - Security Pillar
<https://docs.aws.amazon.com/wellarchitected/latest/security-pillar/welcome.html>
- Microsoft Azure Well-Architected Framework - Security documentation
<https://learn.microsoft.com/en-us/azure/architecture/framework/security>
- Google Cloud Architecture Framework: Security, privacy, and compliance
<https://cloud.google.com/architecture/framework/security>
- CSA Security Guidance for Critical Areas of Focus in Cloud Computing
<https://cloudsecurityalliance.org/artifacts/security-guidance-v4/>
- CSA Top Threats to Cloud Computing Pandemic Eleven
<https://cloudsecurityalliance.org/artifacts/top-threats-to-cloud-computing-pandemic-eleven/>