Cloud Threat Modeling

cloud security alliance®

The permanent and official location for Top Threats Working Group is
https://cloudsecurityalliance.org/research/working-groups/top-threats/

Acknowledgments

Lead Authors:

James Bore
Jon-Michael C. Brook
Alexander Stone Getsin
Vic Hargrave
Vani Murthy
Michael Roza
Vladi Sandler

Contributors:

Randall Brooks
Ken Dunham
Nirenj George
Ebudo Osime
Fadi Sodah
Adalberto Valle

CSA Staff:

Sean Heide
Stephen Lumpe (Cover design)
AnnMarie Ulskey (Layout and card designs)
John Yeoh

Table of Contents

Introduction

Businesses look to cloud technology to enable new business models and scaling of operations with the goal of creating financial opportunities. Some of the pain points in this regard are skills gaps, security, and the technical expertise necessary to support different cloud service providers, models, and technologies.

The identification of attacks and weaknesses that may be exploited before an attacker does so is a crucial practice in secure development and design, allowing feature inclusion decisions and prioritizing security efforts. Threat modeling is an advisable practice for organizations as a cornerstone process in their secure software development lifecycle (including security testing), especially since it is prescribed by leading industry best practices such as OWASP, NIST, and industry secure software design thought leaders (Microsoft). Threat modeling serves to identify and describe a picture of security considerations, primarily threats and preventive measures for a system or application.

The fast pace of cloud adoption surpassed some security methodologies honed through the preceding 40 years of information technology development. Threat modeling is one of those security methodologies that has not kept up with/been on par with/matched the rate of cloud adoption. There is much benefit in aligning the critical practice of threat modeling with cloud services, technologies, and models.

Cloud-specific threat influencers include abstraction models, shared responsibility boundaries, reliability mechanisms, and the diversity in Cloud Service Providers (CSP) offerings for the same technology. The chief factor that contributes to the neglect of threat modeling in cloud systems is the gap in guidance, expertise, and applicability of the practice. This publication attempts to bridge this gap.

Threat modeling for cloud systems, services, and applications, or Cloud Threat Modeling, allows organizations to begin or further security discussions and to assess their security controls and gaps to develop their system design and mitigation decisions in today's cloud-dominant businesses and industry.

Purpose

The purpose of this document is to enable and encourage threat modeling for cloud applications, services, and security decisions. To that end, this resource provides crucial guidance to help identify threat modeling security objectives, set the scope of assessments, decompose systems/applications, identify and rate threats, identify vulnerabilities in the system design, develop and prioritize mitigations and controls, and communicate/report a call-to-action.

Target Audience

The target audience of this document is cloud and security practitioners who analyze threats and assess system preparedness or design cloud systems and services. However, there also are takeaways and insights provided for CIOs, CISOs, and senior managers as a context for what cloud threat modeling is, its unique role, its difference from standard threat modeling, its objectives, and its place within a cybersecurity strategy. Developers and architects will find this document helpful when designing secure cloud systems, as we hope will auditors and regulators when assessing an entity's threat modeling activities.

Key Takeaways

- Threat modeling is beneficial for cloud services, applications, and systems. It enables cloud adoption, selection of the most secure consumption, service and multi-tenancy models, and critical cloud threats mitigation.
- Cloud threat modeling is not conducted differently when compared with standard (on-prem) threat modeling, but it requires unique knowledge and extensive use of industry references and resources, along with many cloud-unique considerations (discussed in this publication).
- Organizations are encouraged to start cloud threat modeling today. See Creating a Cloud Threat Model.

Threat Modeling

Before we discuss cloud threat modeling, we will describe baseline threat modeling qualities and processes taken from various standards and best practices, ones which best serve, in our opinion, as a foundation to cloud threat modeling.

Threat modeling consists techniques of identifying and outlining key threats, attack vectors, and preventive measures for a planned or existing system or application. Thus, relevant controls can be identified early on to help guide the above items.

The purpose of threat modeling is:

- Identifying, analyzing, and rating security threats
- Producing prioritized mitigations
- Assisting and informing attack surface analysis and risk reduction

We have analyzed STRIDE[1], MITRE ATT&CK, OWASP threat modeling, and PASTA[2] and consider the core process steps of threat modeling to be as follows (see more comprehensive list in references):

[1] STRIDE - Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, Elevation of privilege, a widely used threat model.

[2] PASTA - Process for Attack Simulation and Threat Analysis (PASTA)

Core Threat Modeling Activities:

1. **Identify threat modeling security objectives** for the threat modeling exercise, focusing on critical aspects such as confidentiality, integrity, availability, and privacy, e.g.,
   a. protect the company's databases containing customer or regulated information from external attackers;
   b. ensure high availability for the e-commerce web application; and
   c. select a cloud application model with the least attack surface or customer security responsibility.
2. **Set the scope of the assessment** with respect to the systems and/or cloud infrastructure under consideration by providing an overview of the system or cloud application. This typically covers areas such as the various organizational assets, including technology stack used, existing security controls, deployment scenario, type of users, and any specific security or regulatory requirement which needs to be addressed in the threat modeling.
3. **System/application decomposition** covers breaking down the system into subsystems and examining the interaction among the various components. The key activities done in this phase are:
   a. **Understand trust boundaries** (external and internal facing, privileged, unauthenticated, etc.).
   b. **Identify input and egress** to the system (input and output), as well as data format.
   c. **Map the data flows** in the system/s.
4. **Identify and rate the potential threats**; that is, identify the threats, type of attacks, and how the given system or its functionalities can be misused by a malicious user. Some of the common threats are related to unauthorized access, denial of service, information disclosure, etc. The severity of the threat can be rated by using a framework such as DREAD[3].
5. **Identify weaknesses and gaps in the system design and components** to aid the security decisions and define the scope and nature of security testing.
6. **Design and prioritize mitigations and controls** applicable to the predetermined threats and reflect on how those controls would reduce the threat or risk level.
7. **Communicate and create call to action:** Communicate the identified threats, their potential impact and severity, as well as the applicable and proposed controls. Make the modeling data and insights available and call to the action of threat mitigation by design or effect.

Furthermore, the following steps can and often should be undertaken although they are not necessarily a part of threat modeling:

- Assessment of existing controls can be conducted and taken into account (if the systems are in existence, rather than in design or development).
- Security testing of the system in scope (like penetration testing, security requirements testing).
- Measurement of metrics and assessment of key performance indicators for controls.
- The identified threats can be categorized based on well-known models such as Microsoft's STRIDE model.
- Model the Attack: Creating visual representations of the threat or attack surface.
- Devising of threat modeling and/or security concept reports that touch on scope, threat modeling considerations, provide further guidance on the mitigation plan and threats. See appendix for more information.

[3] DREAD (D = damage, R = reliability, E = exploitability, A = affected users, D = discoverability)

- Devising of a mitigation plan or more comprehensive risk analysis and treatment plan.
- Threat actor analysis (motivations, means, methods, techniques, sophistication, industry threat actor matching, etc.)

We provide a Detailed Threat Modeling Reporting Guidance in Appendix 1 for consideration.

The Secure Software Development Lifecycle (SSDLC)[4] best practices framework mandates threat modeling for software as a key early step in secure software design. Furthermore, threat modeling informs active security testing engagements.

Threat modeling is often considered a best practice for security (penetration) testing; it allows to focus on key testing objectives and a better return on investment for the testing. Testing post-implementation of the conceived controls in threat modeling should also be conducted after being implemented to assess their effectiveness and establish assurance and trust.

Finally, threat modeling is always a good exercise to conduct at any system life cycle stage.

We further describe our insights on how cloud threat modeling is different from its on-premise counterpart and our proposal on how it should be conducted.

Cloud Threat Modeling

Cloud threat modeling expands on standard threat modeling practices to account for unique cloud services and an application's qualities and considerations. The observations made below explain how cloud threat modeling differs from non-cloud threat modeling. It is suggested that security and cloud practitioners consider these insights and suggestions to apply threat modeling more effectively and frequently to their advantage.

Cloud threat modeling complements attack surface identification and reduction, assists with the abstraction of security requirements for different cloud service providers, and informs risk management.

Is the Purpose of Cloud Threat Modeling Different from Non-cloud Threat Modeling?

Cloud threat modeling serves a similar purpose [as non-cloud threat modeling] but does offer additional benefits.

The purpose of cloud threat modeling (threat modeling of cloud systems and services) is like that of non-cloud threat modeling. In both cases prioritized mitigations are derived, an assessment of security considerations is made, and threats are identified.

[4] CSA Six Pillars of DevSecOps: Automation page 10 Figure 1 Secure Development Lifecycle

First, cloud threat modeling enables and drives secure cloud adoption. From the onset of cloud technology, more so than with other technologies or changes, security was considered preventative to cloud adoption. Although most of the security-related barriers, such as technology, regulation, or risk, have been lifted or overcome, decision makers still ask the following questions:

- Can I trust cloud services and infrastructure with company X, in a multi-tenant fashion?
- Is it safe to move key business and financial processes to SaaS from our premises?
- Can the cloud offer sufficient privacy and confidentiality controls for sensitive and regulated data?

Cloud threat modeling is a helpful and enabling step in responding to questions such as introduced above. It brings an understanding of threats, assets, and security controls and therefore instills confidence by informing stakeholders and assisting in decisions.

Secondly, cloud threat modeling helps select the most secure, service, deployment, and multi-tenancy model configuration. When selecting service models (SaaS, PaaS, IaaS), deployment models (private, public, hybrid, community), and multi-tenancy environment, the primary considerations are business objectives, security, and regulations. Cloud threat modeling helps determine what threats are inherent to a certain model or design and what controls are available and, therefore, what security efforts are implied as well.

Are the Model Components of Cloud Threat Modeling Different from Non-Cloud Threat Modeling?

The model components are the same but would describe or include unique cloud features.

A cloud threat modeling exercise still considers threats, assets, controls in place, vulnerabilities, appropriate controls, ratings, etc.

- **Scope** considerations of threat modeling for cloud systems and services examine more of the identity management, cloud service configurations, and even underlying cloud accounts.
- **Assets** continue to remain the main concern of cloud threat modeling — data, key system components, monetary instruments, and identities. But new assets are introduced as well, such as cloud accounts, SaaS subscriptions, and services.
- **Threats** to cloud systems, applications, and environments are unique. Different technologies such as instance metadata service and cross-account IAM access federation come into play. Different technologies and consumption models describe cloud systems. Therefore, different attacks are viable against them and to different impacts and impact gravity than otherwise.
- **Controls in Effect** may be built-in with the CSP or may fall within the responsibility of the CSP. Generally, cloud systems and services benefit from more built-it controls than non-cloud systems and services.
- **Rating** of the threats by severity is necessary, whether the cloud is in scope or not; the same considerations apply — how vulnerable the system is to a certain threat, what assets would be impacted, and to what extent. Some threats are inherently more severe in the cloud

(such as administrative account compromise) and others, less so (such as infrastructure/ protocol denial of service) due to cloud accounts, systems, and services qualities. Severity = Likelihood x Impact.

- **Proposed mitigations** are different for cloud systems and applications to a large extent, as they are subject to different threats. Furthermore, some controls are available or applicable only to cloud systems and accounts as they were developed and designed for them, e.g., Service Control Policies for AWS accounts. Furthermore, new, unique cloud controls and technologies are applicable and to be considered for cloud services and applications (such as metadata service protections, CSPM[5] and others).

Are the Threats Considered in Cloud Threat Modeling Different from Non-Cloud Threat Modeling?

Threats to cloud systems, applications, and environments are unique. Different technologies such as instance metadata service and cross-account IAM access federation come into play. Different technology and consumption models describe cloud systems. Therefore, different attacks are viable against them and to different impacts and impact gravity than otherwise.

Much of the work of this research group is aimed at learning about cloud threats. While we determine that the risk or impact might seem similar to the non-cloud (such as Data Breach, Egregious Eleven #1), the threats are unique (like AWS EC2 instance metadata account hijacking, Imperva breach, 2018).

Are the Products of Cloud Threat Modeling Different Non-Cloud Threat Modeling?

The product and outputs of cloud threat modeling are like those of standard threat modeling.

However, threat modeling for cloud applications and services strongly impacts unique cloud-related decisions.

The products or output of threat modeling are

1. the threat model, usually presented in an Excel sheet or via a visual model such as tree-type mapping;
2. prioritized mitigating controls, ranked, and
3. security and design decisions or a more conclusive and detailed call to action.

The assessment data and its visualization, the model, is unique for every application, cloud or not; thus, there is no unique cloud distinction to be found on those terms. So are the prioritized mitigating controls, unique for every scope, cloud or not.
Finally, there is a distinction in the outputs of design decisions (or a more conclusive and detailed call to action). As described in cloud threat modeling purpose (page 8), the unique and highly impactful insights around cloud security help to decide whether cloud adoption is a viable option and which

[5] CSPM - Cloud Security Posture Management (CSPM), monitor for, find and remediate cloud misconfigurations.

cloud model is the best fit. While other security activities and architectural assessments do yield design decisions around technological components, threat modeling for cloud applications and services strongly impacts cloud-related decisions.

Who should be responsible for building the Cloud Threat Model?

Cloud threat modeling requires cloud expertise on the threat modeling team.

Ideally, a cloud threat model should be composed by individuals and teams with security and cloud expertise. An accurate threat model requires understanding of the overall architecture, components/services, infrastructure, and business context of the system and a relevant adversarial perspective on the system in question or design.

Individuals and roles such as application architect/analyst, cloud architect/analyst, security architect/analyst and other technical positions should be consulted and are usually well- enough informed and expertise-equipped to lead cloud threat modeling.

Cloud Threat Modeling Process

Is the Process of Cloud Threat Modeling Different from Non-Cloud Threat Modeling?

The process will not be different, but it combines several other procedures, methodology, and needs such as a holistic review of the cloud infrastructure (cloud stack) such as focus on the configuration stage and detection of the root cause for attack/failure kill chain as an approach to handle the risk, etc. It is important to keep in mind that this process is repeatable for each system/application that is profiled.

We also need to clearly understand the Cloud Service Model adopted (e.g., IaaS/PaaS/SaaS) and with it, the shared responsibility model which defines the responsibility (including security) of the CSP and its users (CSCs). The accountable party will be responsible for the actions necessary to identify and mitigate the potential threats. For example, most cloud providers include basic DDoS protection for their services, but advanced protection may be optional.

Core Threat Modeling Activities

1. **Identify threat modeling security objectives:** Identify security objectives and requirements. What are the organization's policies and business needs? Also identify the target system's architecture, including its components, roles, services, and dependencies. As a complement to standard goals setting for a threat modeling exercise (identifying most impactful controls or most pressing threats, protecting confidentiality, etc.), it is important to set cloud threat modeling security objectives, such as identifying the most risk-averse cloud service and deployment models (IaaS, PaaS, SaaS) to determine whether "cloud" or cloud architecture (such as PaaS and multi-tenancy) are permissible.
2. **Set the scope of the assessment:** Provide an overview of the systems or applications under review (data, applications, systems, users, controls, etc.) by identifying and scoring assets.

Consider the cloud-stack related questions such as
   a. Is the PaaS control plane in scope?
   b. Is the cloud account in scope?
   c. Is it advisable to make an inclusive scoping?
3. **System and application decomposition:** This typically covers breaking down the system into subsystems and examines the interaction among the various smaller components. Decompose the system further to identify how it functions and how those functions can be vulnerable. For example, how does the application ensure the confidentiality of data in transmission?
   a. **Understand trust boundaries** (external facing, internal facing, privileged, unauthenticated, etc.). Understand cloud trust boundaries like trust in and segregation controls against the CSP#, cross service and account trust, and multi-tenancy segregation controls.
   b. **Identify entry and exit points** to the system (input and output) and format. Consider cloud unique entry points such as cloud management API, managed API gateways, inbound and outbound connectivity, and integrations. Furthermore, map cross-cloudcross cloud services relationships.
   c. **Map the data flows** in the system. Consider cloud unique data flows and stores such as cloud EMR# and ETL services, blob storage, and account log trails.
4. **Identify and rate the threats:** Identify the threats, type of attacks, and the various ways a given system, or its functionalities can be misused by an external attacker or malicious user. Identify cloud unique threats using industry resources such as the CSA Top Threats. Do not neglect to assess threats to "Availability" even though many controls to that end would be baked in the CSA platform and infrastructure. Give special consideration to human errors, insider threats, misconfigurations, and poor design.
5. **Identify weaknesses and gaps in the system design and components** to aid the security decisions and define the scope and nature of security testing. Consider common and impactful cloud design, implementation weaknesses, and account for defense-in-depth design/controls:
   a. **Egregious Eleven[6] #2** - Misconfiguration and Inadequate Change Control
   b. **EE #3** - Lack of Cloud Security Architecture and Strategy
   c. **EE #4** - Insufficient Identity, Credential, Access, and Key Management
   d. **EE #7** - Insecure Interfaces and APIs
5. **Design and prioritize mitigations and controls** applicable to the predetermined threats and reflect how those controls would reduce the threat or risk level. Leverage cloud security controls (matrix)[7] and focus on controls that disrupt cloud threats, including "attack kill chains," even when some cloud and application misconfigurations and weaknesses are in place.
6. **Communicate and call to action:** Communicate the identified threats, their potential impact and severity as well as the applicable and proposed controls. Make the modeling data and insights available. Communicate cloud design decisions and core enabling cloud controls.
7. **Periodic reevaluation:** Cloud platforms are rapidly evolving and never static; hence, the threat model needs to be, too. Periodic review and update of the threat model will ensure

[6] Egregious Eleven cloud security concern, more in the CSA Top Cloud Threats Research work group publication *Top Threats to Cloud Computing: Egregious Eleven*

[7] The CSA Cloud Controls Matrix (CCM) is a cybersecurity control framework for cloud computing that maps and categorizes applicable cloud controls.

that the threat model remains relevant and highly useful to the team. This is especially true If there are material changes to the overall architecture (for example, components/services added or removed, etc.). An outdated threat model may lull the team into a false sense of security where new threat vectors are not considered and thus not evaluated.

Creating a Cloud Threat Model

The key outcome we would like to see is for more threat modeling to be applied to cloud systems and services in the design or as part of assessments. We encourage readers to create a cloud threat model today and leverage this resource.

How to start from scratch

One does not need to be a security expert or be proficient at threat modeling to start cloud threat modeling. Experts can use this document as a complement to their threat modeling practices.

For those starting from scratch, start small and begin with what is familiar:

1. Choose one of our cloud threat modeling cards from Appendix 2, threat, vulnerability, or control, or another one which is of most concern or is most familiar.
2. Determine whether any of the other cards is related to yours and align or visually place them together by the following suggested order: Threats, Vulnerabilities, Controls, Assets.
3. Identify more Threats, Vulnerabilities, Controls, and Assets related to the developing model and introduce them into the "mix" for visualization or analysis. Consider referring to the latest CSA Cloud Top Threats Egregious Eleven for references.
   - Repeat until there is at least one of each card type.
4. Ensure that every threat and vulnerability is addressed with at least one or two appropriate, specific controls.
5. The process is complete, congratulations!

Alternatively, if a more detailed and comprehensive approach is needed, please consider following the steps described in Cloud Threat Modeling Process.

Once the modeling part is done, you've completed a security assessment and determined actionable steps to take or have already taken them. Act on the identified controls and mitigate the vulnerabilities (or establish assurance of their lack thereof).

The result should look like a basic threat model, with cloud-specific considerations, like ours, in the next section, "A Cloud Threat Model Reference."

A Cloud Threat Model Reference

Below is a basic cloud threat model reference we created by using the steps described in prior sections.

Referring to the CSA Cloud Top Threats Deep Dives Egregious Eleven resource, we recall the 2019 Dow Jones data exposure.

> - **Actor:** An authorized third-party vendor for Dow Jones failed to password protect an AWS-hosted Elasticsearch database belonging to Dow Jones.
> - **Attack:** With no password protection, the database was available to anyone without restriction and could be found with commonly available IoT search engines. The misconfigured database was discovered in 2019 by a security researcher who reported it to Dow Jones.
> - **Vulnerabilities:** The Dow Jones database was not password protected by one of their authorized and presumably trusted security vendors.

Below is a basic threat model composed of just the cloud threat modeling cards provided in Appendix 2:

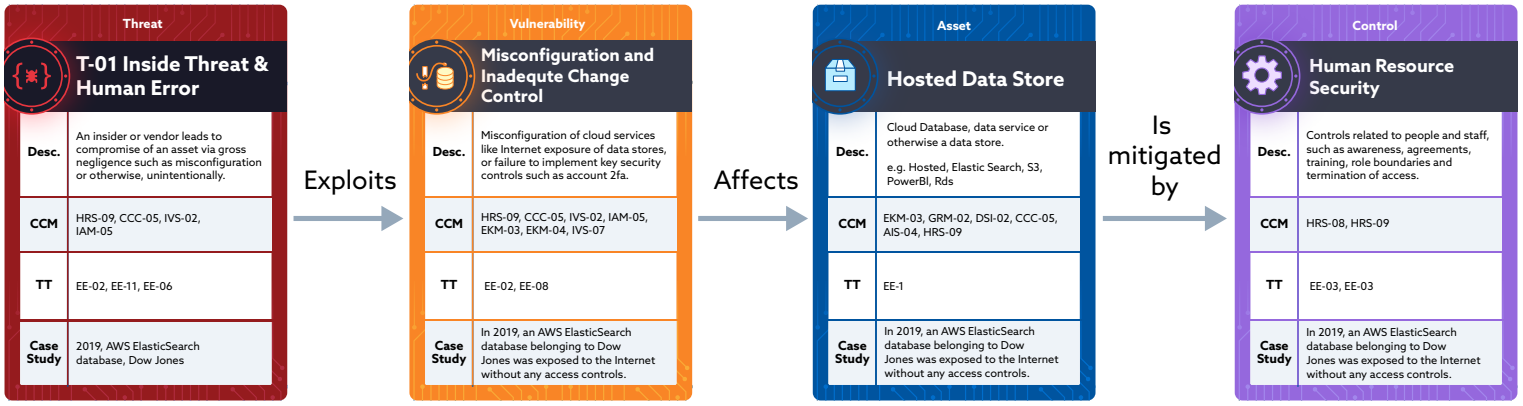


Figure 1

However, this model is far from complete; our cloud threat modeling cards are a good place to start to complete the model. After the cards are incorporated, one would be able to expand on applicable vulnerabilities and controls.

Conclusions

In summary, threat modeling is an essential practice for software and systems security, doubly so for cloud software, systems, and services. Organizations need to develop a structured and repeatable approach for modeling these threats. Threat modeling will enable teams to anticipate the process of a cyber-attack and provide a threat's impact and possible countermeasures in a proactive manner before the attacks happen.

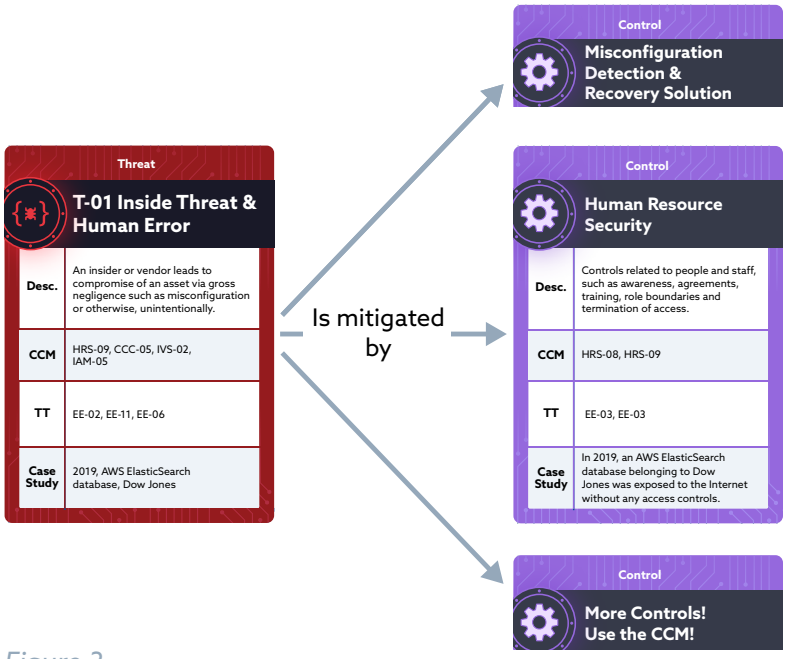


Figure 2

This is doubly true for cloud in-scope threat modeling exercises where much abstraction, ubiquity and shared trust boundaries persist. Threat modeling allows an easy start to a security discussion and establishes and helps communicate an understanding of concerns and applicable controls. It allows organizations to reach cloud design and threat mitigation decisions.

Cybersecurity risk management necessarily involves prioritization of people, processes, and technologies related to the protection of core organizational risk goals, objectives, and high-value assets. This is achieved, in part, by the identification of threats, tools, tactics, and procedures (TTP). Once such threats and TTP are identified, an organization can map those back to preventive, detective, and corrective controls to manage risk. Threat modeling cards can be used to better introduce consideration of threats into a risk management process to mature a cybersecurity program over time.

We hope this resource and guidance helps organizations begin, or improve, their threat modeling journey.

In future publications, we hope to provide further insights on cloud threat modeling, with cloud threat scoring, ATTACK ID and other industry references, and a more complete cloud threat modeling cards deck.

Stay tuned!

References

- https://en.wikipedia.org/wiki/Threat_model
- https://www.microsoft.com/en-us/securityengineering/sdl/threatmodeling
- https://www.microsoft.com/en-us/securityengineering/sdl/practices
- Microsoft Threat Modeling Tool: https://docs.microsoft.com/en-us/azure/security/develop/threat-modeling-tool
- Model vs methodology: https://drive.google.com/file/d/1n_uMBckp8UMBA1oq1kcKTjvXX6Ea_tLF/view?usp=sharing
- CSA Cloud Top Threats Egregious Eleven: https://cloudsecurityalliance.org/artifacts/top-threats-to-cloud-computing-egregious-eleven/
- CSA Cloud Top Threats Deep Dives Egregious Eleven: https://cloudsecurityalliance.org/artifacts/top-threats-egregious-11-deep-dive/
- CSA CCMv4 Matrix: https://cloudsecurityalliance.org/artifacts/cloud-controls-matrix-v4/
- Security Guidance for Critical Areas of Focus in Cloud Computing v4.0: https://cloudsecurityalliance.org/artifacts/security-guidance-v4/
- CSA The Six Pillars of DevSecOps: Automation (Pillar 5): https://cloudsecurityalliance.org/artifacts/devsecops-automation/
- Chapter 4 - A Threat Analysis Methodology for Cloud Using CCM" in the CSA's Certificate of Cloud Audit Knowledge Common Body of Knowledge
- PASTA (Process for Attack Simulation and Threat Analysis)
- VAST (Visual, Agile, and Simple Threat Modeling)
- OCTAVE (Operationally Critical Threat, Asset, and Vulnerability Evaluation).
- DREAD risk assessment model
- NIST Special Publication 800-150 Guide to Cyber Threat Information Sharing: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-150.pdf
- MITRE ATT&CK - attack.mitre.org

Appendix 1: Threat Modeling Reporting Detailed Guidance

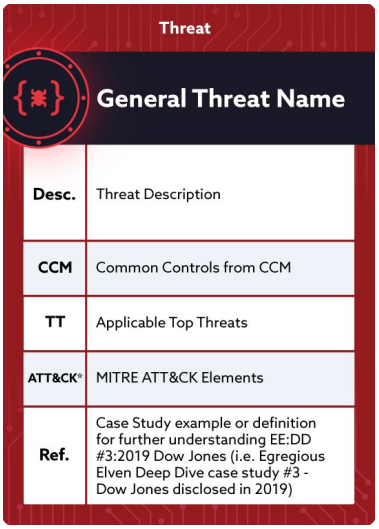
Devise a security design report which will be mapped to a threat modeling report and will include a technical level explanation of proposed security control with detailed requirements and references to relevant risks.

The document may be constructed from the following chapters:

- **Executive Summary**: General explanation of the project, primary goals, and cloud architecture snapshot.
- **System/application decomposition**: This typically covers breaking down the system into subsystems and examines the interaction among the various smaller components.
- **Attack Vectors Mapping**: Explanation of the top five detected risks in the architecture, including the explanations, cloud architecture snapshot with detected risks' icons on it, total security risk score, and summary chapter.
- **The Threat Model** visual, relational, textual, or otherwise, represents the various threat model components and insights and mapping their relationships.
- **Mitigation Plan**: This chapter will include a list of the required implementation of the security controls ordered by priority. Each security control will consist of the following fields:
  - Security Control Name (e.g., SC1 - Implementation of Authentication SSO Mechanism)
  - Priority - This field is required to define the priority of the implementation and can be critical/high/medium/low
  - Risks - Reference to the relevant risks in the threat modeling report (e.g., R1, R2, R5)
  - Requirements - List of technical level requirements that are relevant for this security control. Each requirement starts with the words must, must not, should, and should not. For example: "The system must implement OAuth 2.0 standard using JSON Web Token (JWT) format."

Appendix 2: Cloud Threat Modeling Cards

Please see the example of a threat modeling card with a description of elements below:

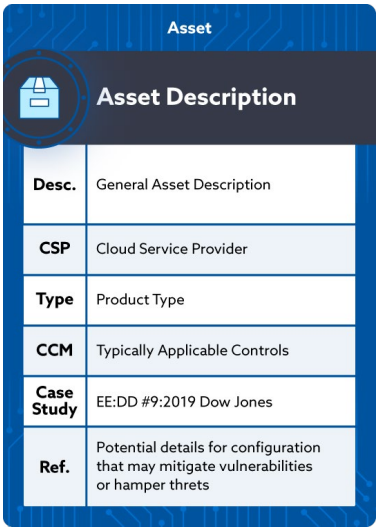


General Threat Name and Description: While a taxonomy for the "General Threat Name" will be an area for future research, both the description and threat name should reinforce the situation for users.

Common Controls and Top Threats: These are references to CSA's research on controls (CCM) and cloud threats (Top Threats). Some threats align well with certain mitigations, such as a security guard at a gate preventing uninvited visitors. List these Cloud Controls Matrix domains in the common controls section. Likewise, if the threat corresponds to the latest *Top Threats* Working Group's publications, such as the Egregious Eleven, include those here.

ATT&CK: External references to the MITRE ATT&CK framework provide additional understanding for researchers and card users, including a set of nomenclature and definitions.

Reference: Relevant case studies or reference to Top Threats Deep Dive publication.



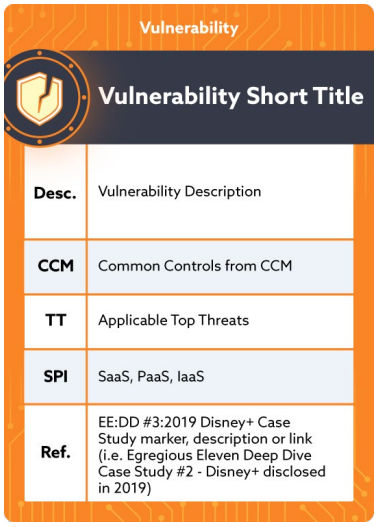
SPI[8] Asset Description: Fill the blank with the system's cloud instantiation, either SaaS, PaaS, or IaaS. The Asset Title and Description should provide enough information for someone else using the cards.

Cloud Service Provider: Identify a CSP and service example of the asset in general terms, such as AWS and ElasticSearch.

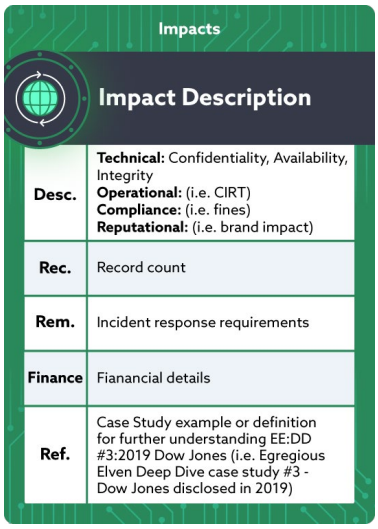
Common CCM Controls and Case Study: Applicable and standard CSA Cloud Control Matrix, controls typically applicable to the type of asset. The case study follows the description in the Threat section.

[8] SPI - SaaS/PaaS/IaaS cloud service models

Descriptions for Sections Added for the Vulnerability cards



SPI Applicability: Vulnerabilities do not apply to various SPI (SaaS/PaaS/IaaS) instantiations based on the Shared Responsibilities Model and the vulnerabilities themselves. This location allows a quick go/no-go for a vulnerability against an asset card.



Impact Title and Description: Choose from either Technical Impacts or Business Impacts and a short description of the circumstance. More information on impacts may be found in "Chapter 4 - A Threat Analysis Methodology for Cloud Using CCM" in the CSA's Certificate of Cloud Audit Knowledge Common Body of Knowledge.

Technical Impacts may be broken into the following categories:

Confidentiality: Unauthorized disclosure of information resulting in limited, serious, or severe damage to the organization's operations, assets, and human resources.

Availability: Unauthorized modification or destruction of information resulting in minor damage to the organization's operations, assets, and human resources.

Integrity: Restricted access to or use of information or an information system resulting in limited, serious, or severe damage to the organization's operations, assets, or human resources.

Business Impacts may be broken into the following categories:

Financial Impact examples include compensation or revenue loss, technical investigations, ransomware system upgrades, insurance premium increases, litigation costs, increased financing costs, reduced investments, severance pay, and other staff termination and recruitment costs.

Operational Impact examples include reduced product/service sales, production/service delays, BOMs (bills of materials) may be corrupted, production/service planning files may be corrupted,

product/service quality, product/service delivery, new product/service introduction delays, and production/service reporting systems.

Compliance Impact examples include regulatory investigations and fines, litigation against affected individuals, litigation against other third parties, need to hire attorneys and experts to defend against regulatory investigations and litigation, and costs to improve the legal and compliance function.

Reputational Impact examples include damaged public perception, damaged customer relationships, damaged supplier relationships, reduced business opportunities, recruitment difficulties, key staff loss, and media scrutiny.

Remediation: Incident Response Requirements

Additions for the Control cards

Domain and Control Title: Add a family or domain shortcode from the Cloud Controls Matrix and description title.

Processes: Name any procedure associated with this control.

Top Threat: These are references to CSA's research on controls (CCM) and cloud threats (Top Threats)

Case Study Mitigations: In situations where mitigations are known, especially in those externally well publicized or internally derived from incident response experiences.

Associated Security Tools (AST): Security tools may help apply controls, from the detective, preventive to detective controls.