AWS re:Inforce

JUNE 10 - 12, 2024 | PHILADELPHIA, PA

APS233

Threat modeling your generative AI workload to evaluate security risk

Ana Malhotra
(she/her)
Security Solutions Architect
AWS

Kareem Abdol-Hamid
(he/him)
Senior Prototype Developer
AWS

Threat ID	Threat statement	Mitigations	Priority	STRIDE
T-001	An external threat actor with access to the public-facing application can inject malicious prompts that overwrite existing system prompts, resulting in healthcare data from other patients being returned, impacting the confidentiality of the data in the database	M-001: Define acceptable use with system prompt M-002: Sanitize for known parameters M-003: Predefine and check against acceptable SQL statements M-004: Validate that output is relevant to user	High	S, I
T-002	A threat actor able to submit content to an LLM system can embed malicious prompts in that content, which can manipulate the LLM into undertaking harmful actions and compromise integrity and availability of LLM system and connected resources	**M-005**: Validate and sanitize input **M-006**: Segregate external content **M-007**: Limit LLM access to other systems	High	S, I, E
T-003	A threat actor able to interact with an LLM system can exploit insufficient output encoding, which leads them to achieve XSS or code injection and results in reduced confidentiality and/or integrity of user data	**M-008**: Encode outputs to prevent unintended code execution **M-009**: Validate and sanitize outputs **M-010**: Apply CORS restrictions	Medium	S, I

What is a threat model?

SHOSTACK'S 4-QUESTION FRAME

What is a threat model?

SHOSTACK'S 4-QUESTION FRAME

What are we working on?

What is a threat model?

SHOSTACK'S 4-QUESTION FRAME

What are we working on?

What can go wrong?

What is a threat model?

SHOSTACK'S 4-QUESTION FRAME

What are we working on?

What can go wrong?

What are we going to do about it?

What is a threat model?

SHOSTACK'S 4-QUESTION FRAME

What are we working on?

What can go wrong?

What are we going to do about it?

Did we do a good enough job?

What is a threat model?

SHOSTACK'S 4-QUESTION FRAME

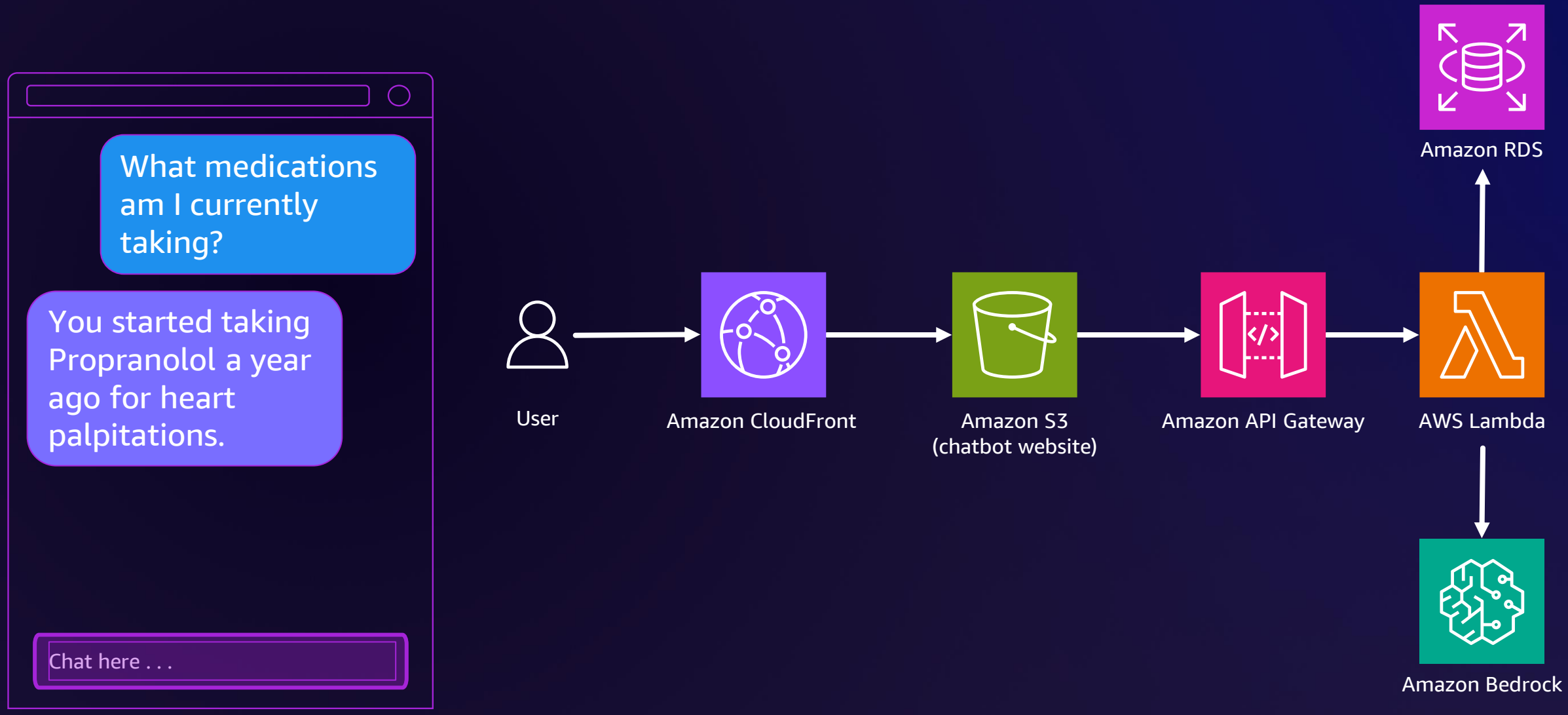
What are we working on?

What can go wrong?

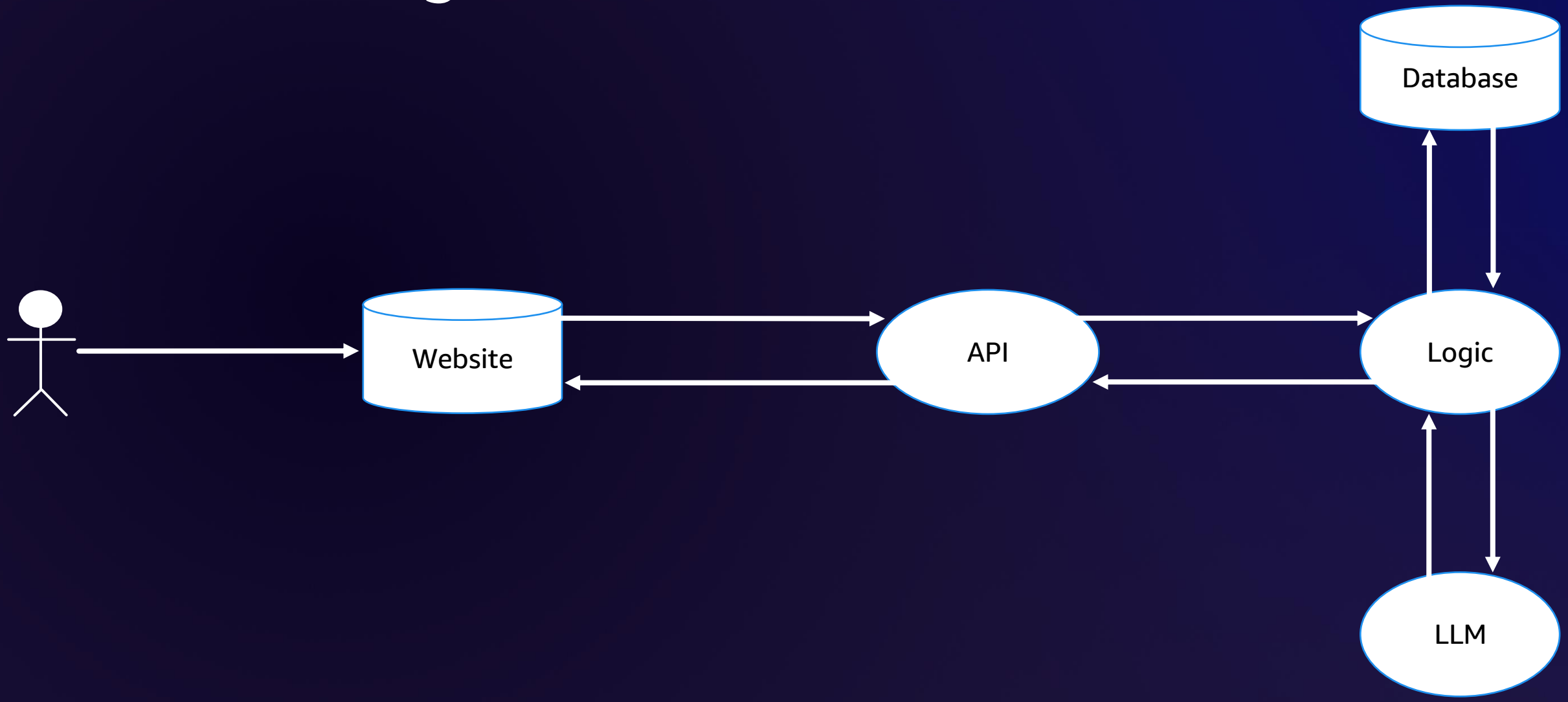
What are we going to do about it?

Did we do a good enough job?

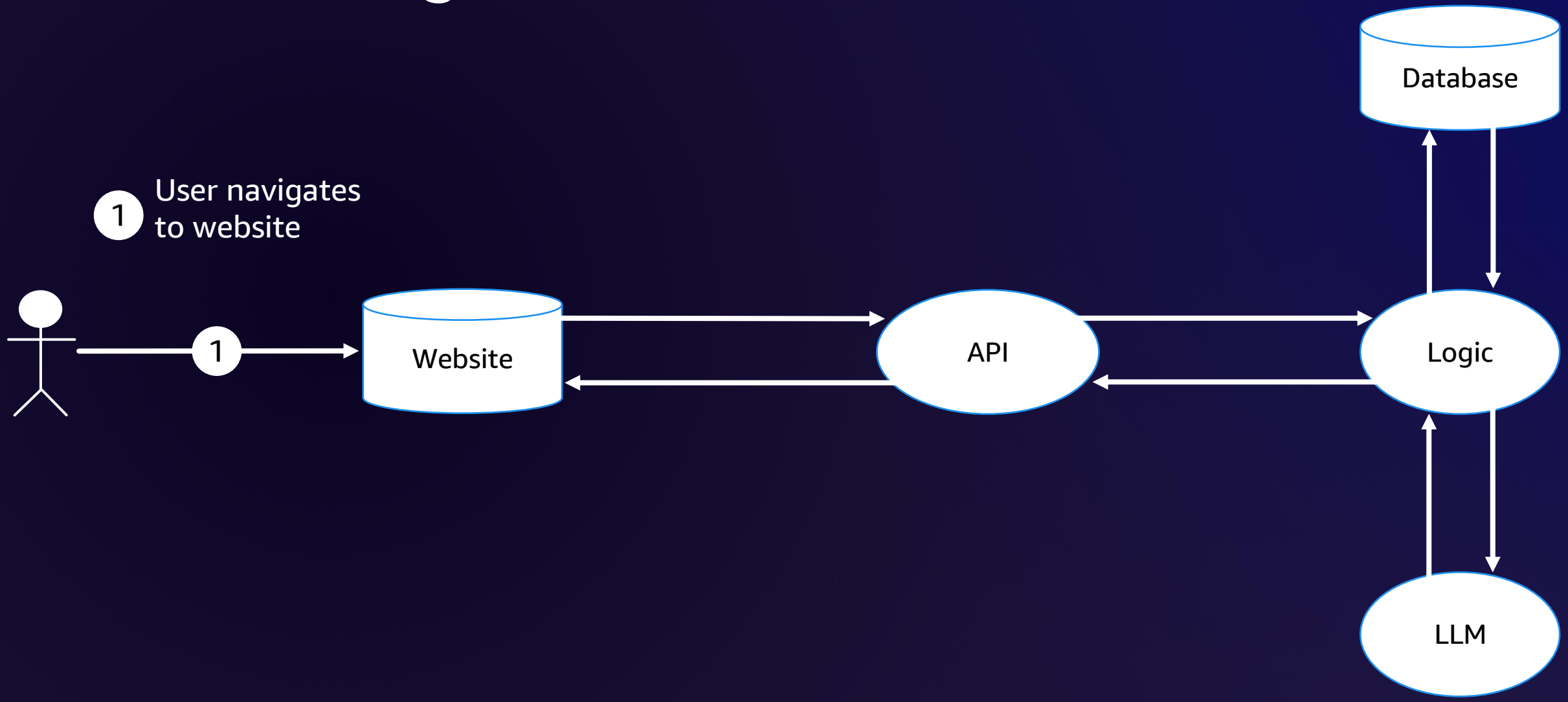
What are we working on?



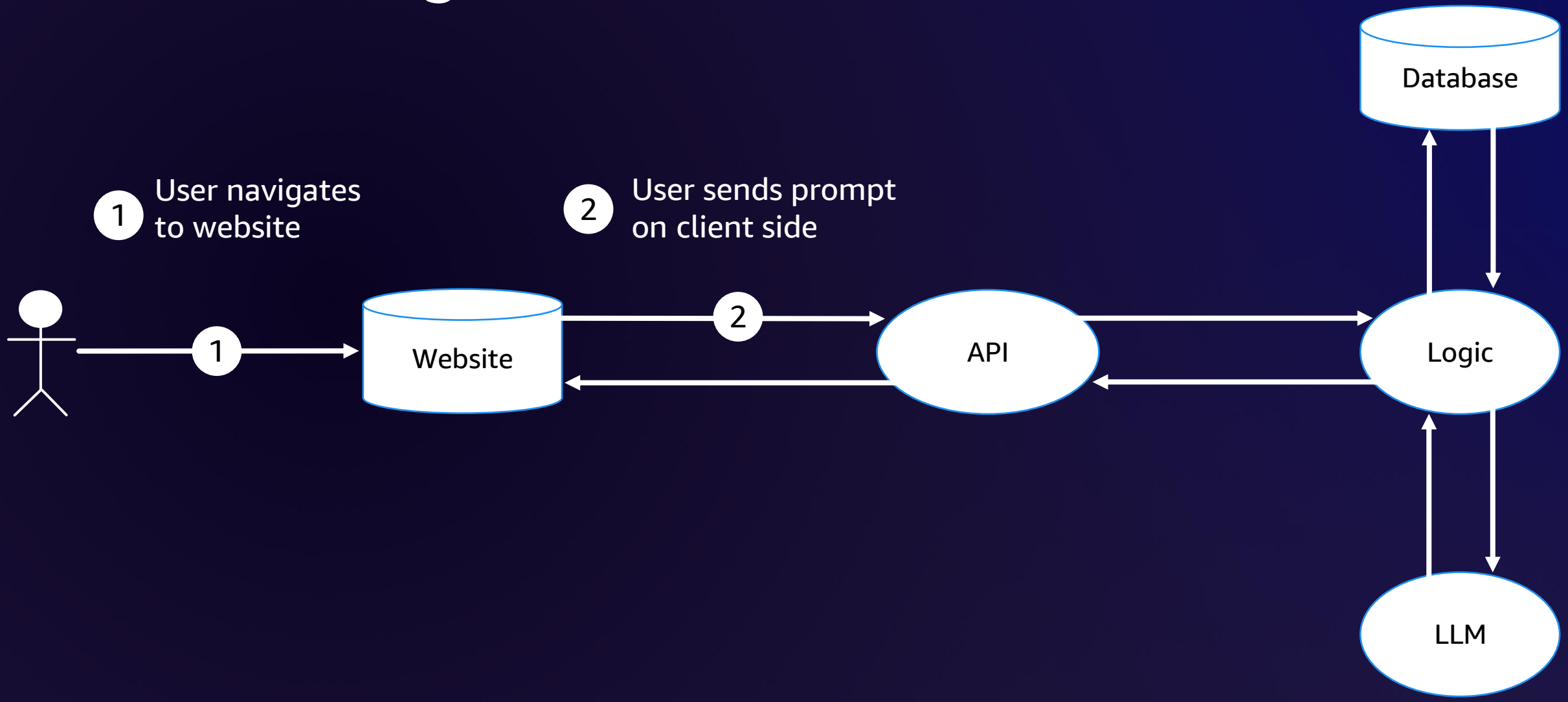
Data flow diagram



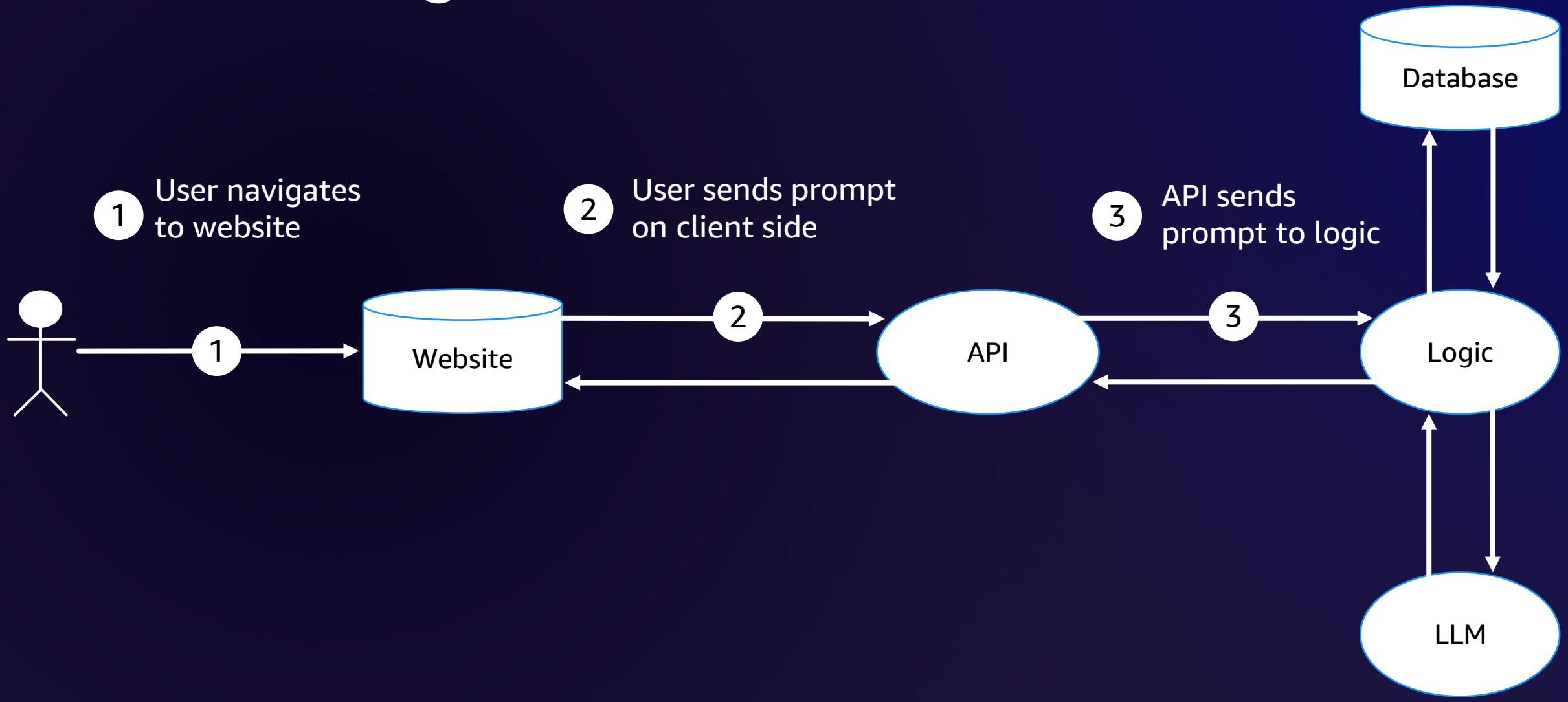
Data flow diagram



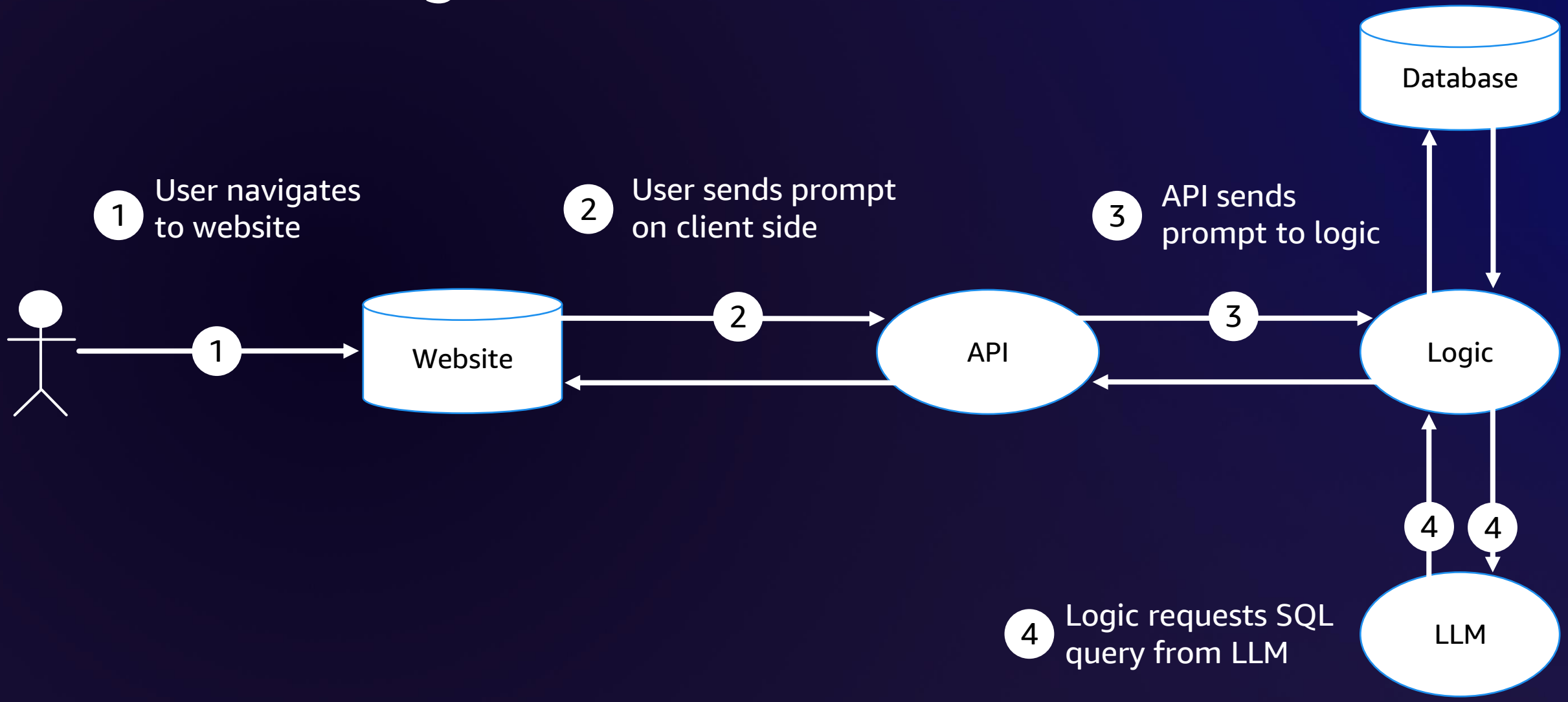
Data flow diagram



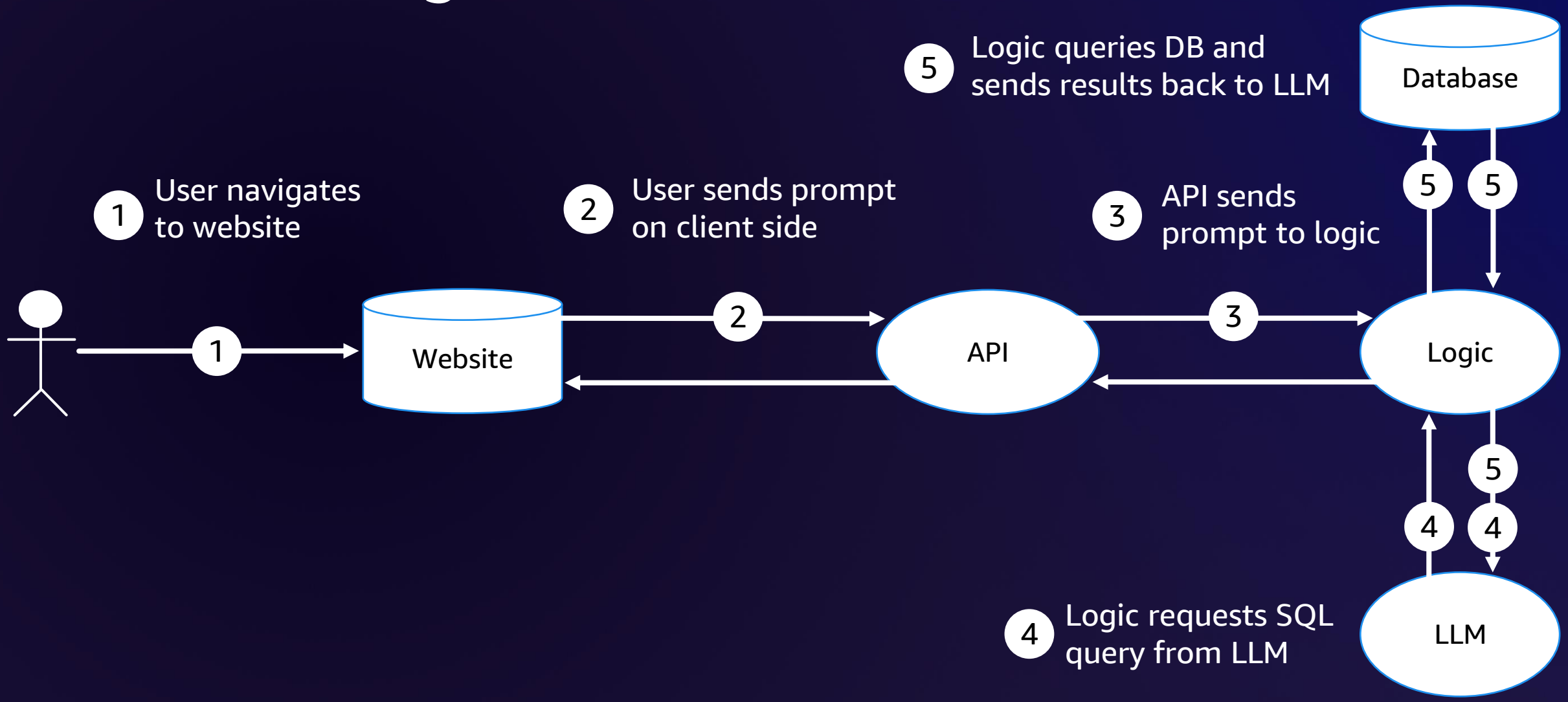
Data flow diagram



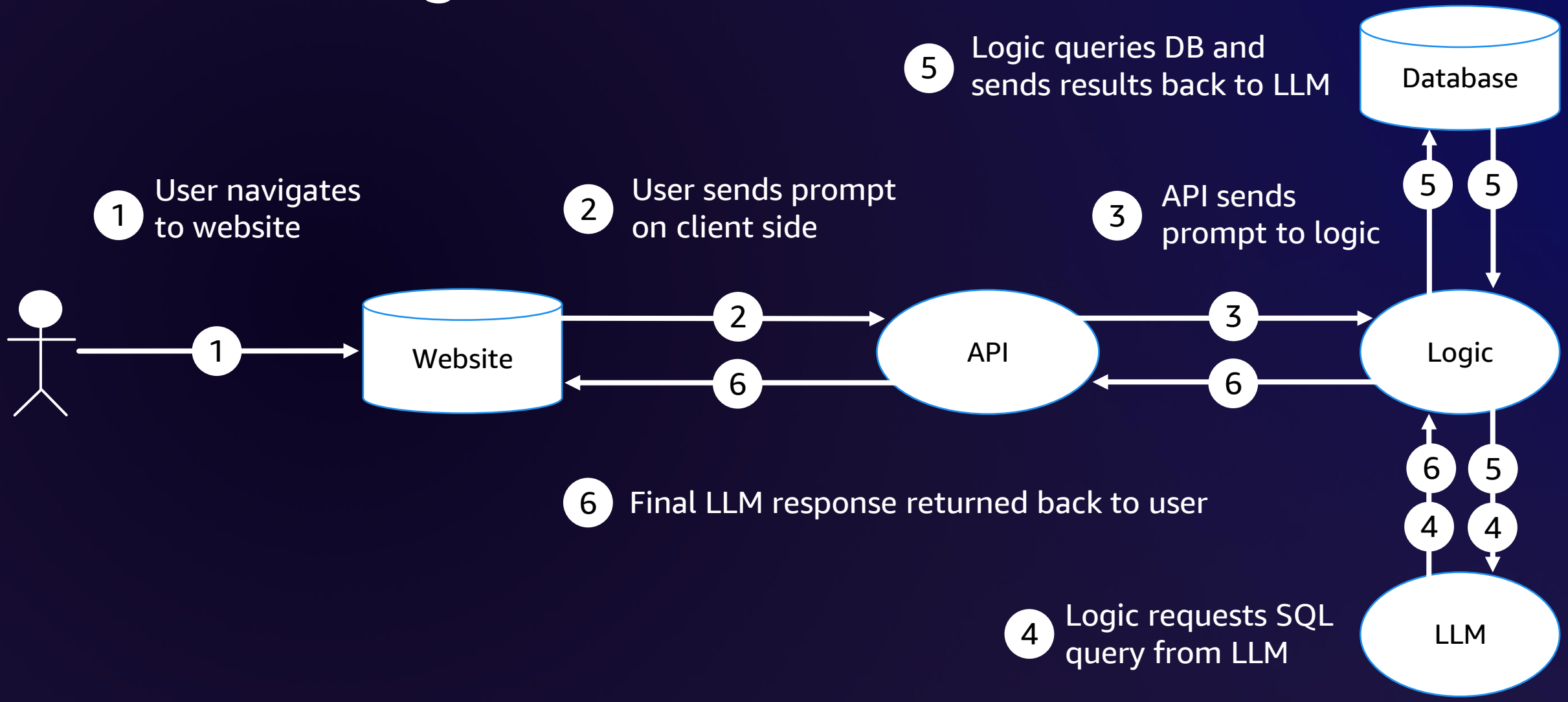
Data flow diagram



Data flow diagram



Data flow diagram



- ✓ 01 Product, features, and use cases
- ✓ 02 Architecture
- 03 Design documentation
- ✓ 04 Dataflows
- 05 Assumptions
- 06 And more

- [x] 01 Product, features, and use cases
- [x] 02 Architecture
- [x] 03 Design documentation
- [x] 04 Dataflows
- [ ] 05 Assumptions
- [ ] 06 And more

- ✓ 01 Product, features, and use cases
- ✓ 02 Architecture
- ✓ 03 Design documentation
- ✓ 04 Dataflows
- ✓ 05 Assumptions
- 06 And more

- 01 Product, features, and use cases
- 02 Architecture
- 03 Design documentation
- 04 Dataflows
- 05 Assumptions
- 06 And more

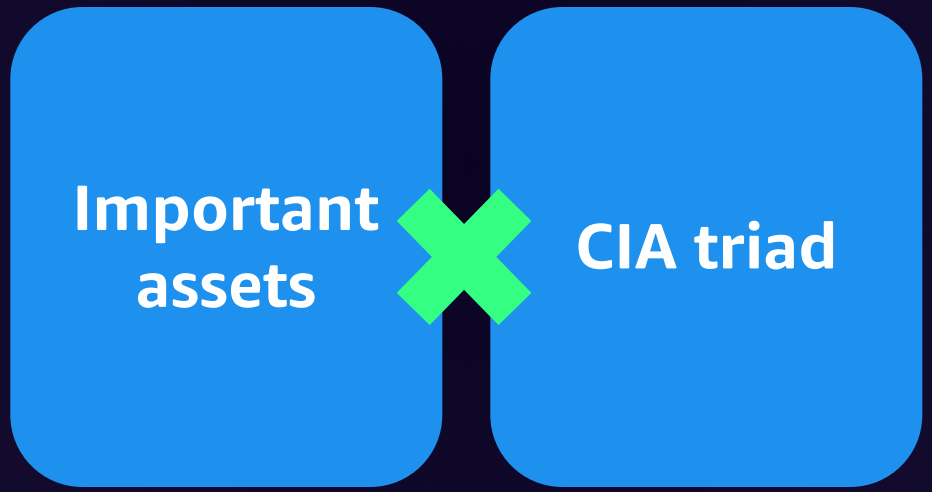
What can go wrong?

What threats should we care about?

What threats should we care about?

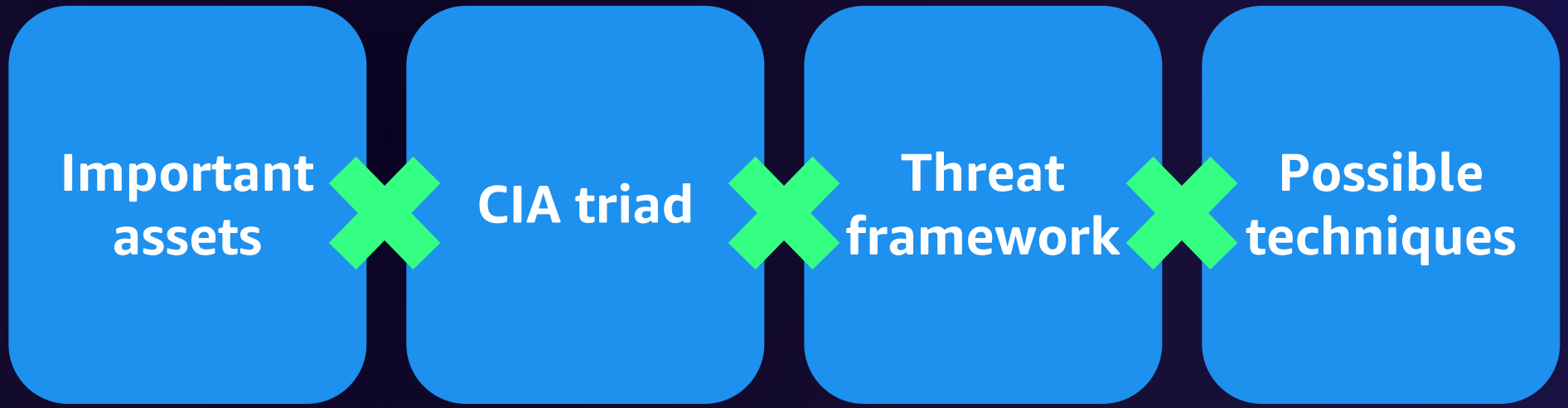
Important assets

What threats should we care about?



What threats should we care about?

What threats should we care about?

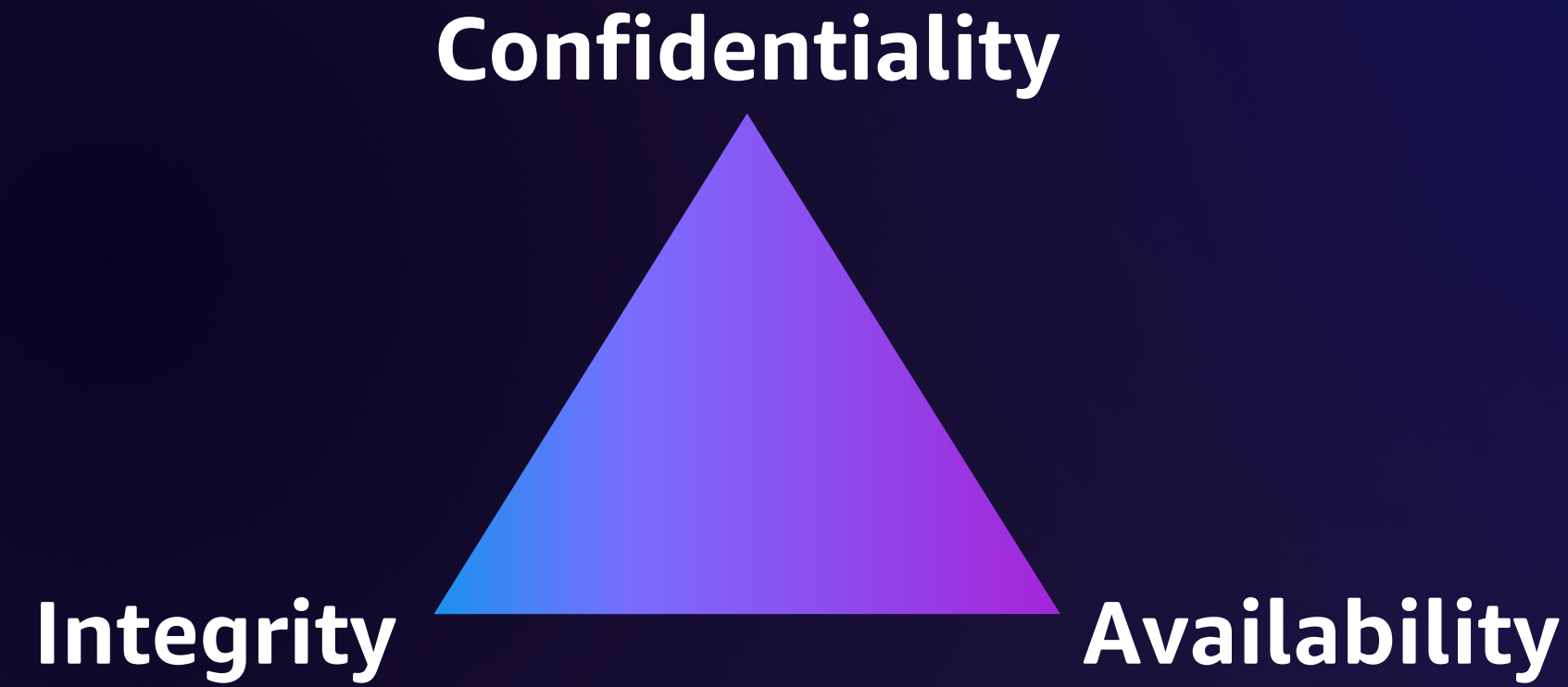


What threats should we care about?

What threats should we care about?

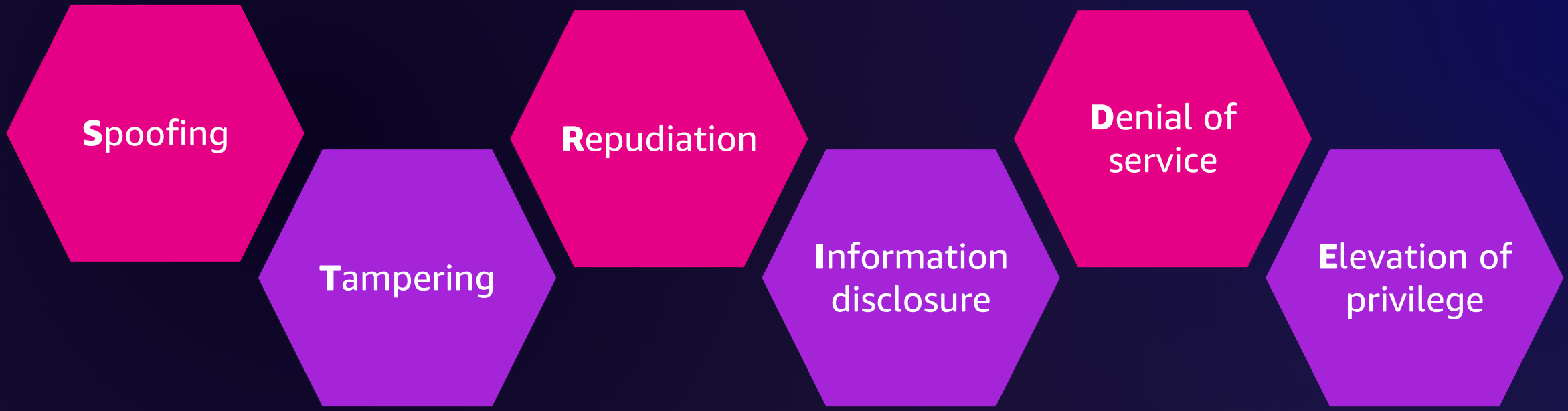
What threats should we care about?

CIA triad



What threats should we care about?

Threat framework: STRIDE



Threat framework: STRIDE

What threats should we care about?

OWASP Top 10 for LLM Applications

LLM01

Prompt Injection

This manipulates a large language model (LLM) through crafty inputs, causing unintended actions by the LLM. Direct injections overwrite system prompts, while indirect ones manipulate inputs from external sources.

LLM02

Insecure Output Handling

This vulnerability occurs when an LLM output is accepted without scrutiny, exposing backend systems. Misuse may lead to severe consequences like XSS, CSRF, SSRF, privilege escalation, or remote code execution.

LLM03

Training Data Poisoning

This occurs when LLM training data is tampered, introducing vulnerabilities or biases that compromise security, effectiveness, or ethical behavior. Sources include Common Crawl, WebText, OpenWebText, & books.

LLM04

Model Denial of Service

Attackers cause resource-heavy operations on LLMs, leading to service degradation or high costs. The vulnerability is magnified due to the resource-intensive nature of LLMs and unpredictability of user inputs.

LLM05

Supply Chain Vulnerabilities

LLM application lifecycle can be compromised by vulnerable components or services, leading to security attacks. Using third-party datasets, pre- trained models, and plugins can add vulnerabilities.

LLM06

Sensitive Information Disclosure

LLMs may inadvertently reveal confidential data in its responses, leading to unauthorized data access, privacy violations, and security breaches. It's crucial to implement data sanitization and strict user policies to mitigate this.

LLM07

Insecure Plugin Design

LLM plugins can have insecure inputs and insufficient access control. This lack of application control makes them easier to exploit and can result in consequences like remote code execution.

LLM08

Excessive Agency

LLM-based systems may undertake actions leading to unintended consequences. The issue arises from excessive functionality, permissions, or autonomy granted to the LLM-based systems.

LLM09

Overreliance

Systems or people overly depending on LLMs without oversight may face misinformation, miscommunication, legal issues, and security vulnerabilities due to incorrect or inappropriate content generated by LLMs.

LLM10

Model Theft

This involves unauthorized access, copying, or exfiltration of proprietary LLM models. The impact includes economic losses, compromised competitive advantage, and potential access to sensitive information.

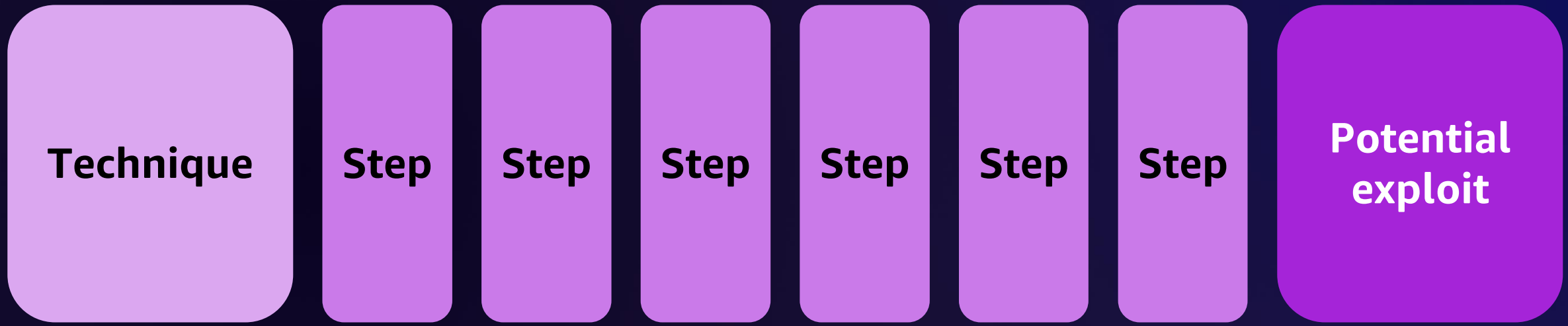
What threats should we care about?

Threat statement

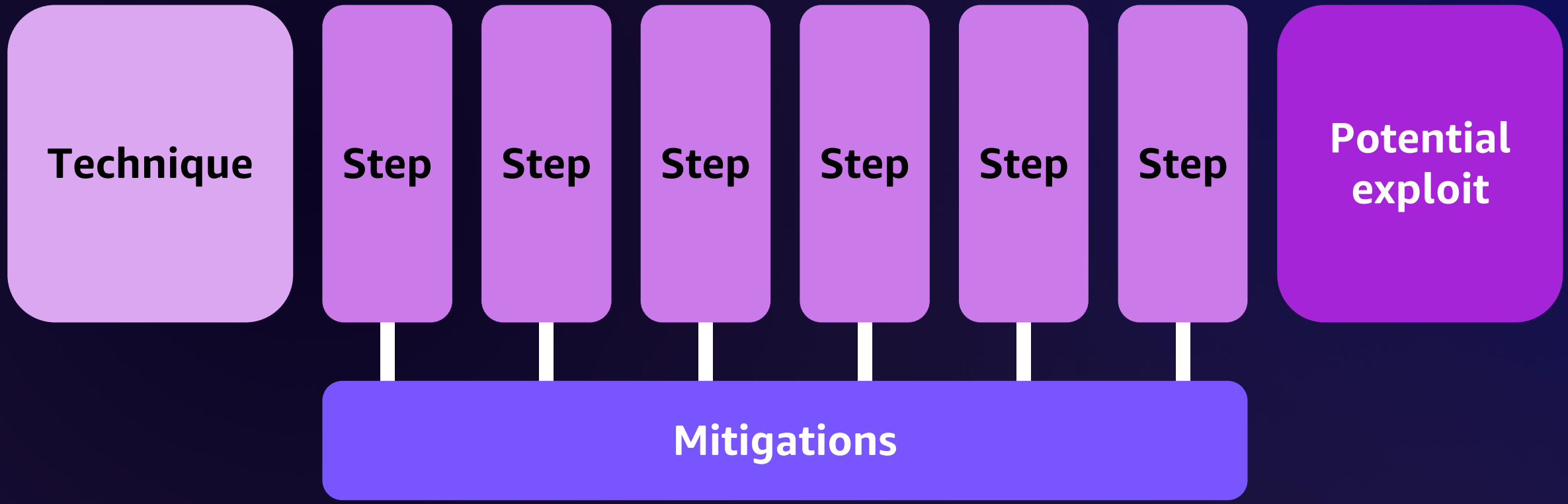
A **threat source** with **prerequisites**,
can **threat action**,
which leads to **threat impact**,
negatively impacting **goal** of
impacted assets

High-level attack steps

High-level attack steps

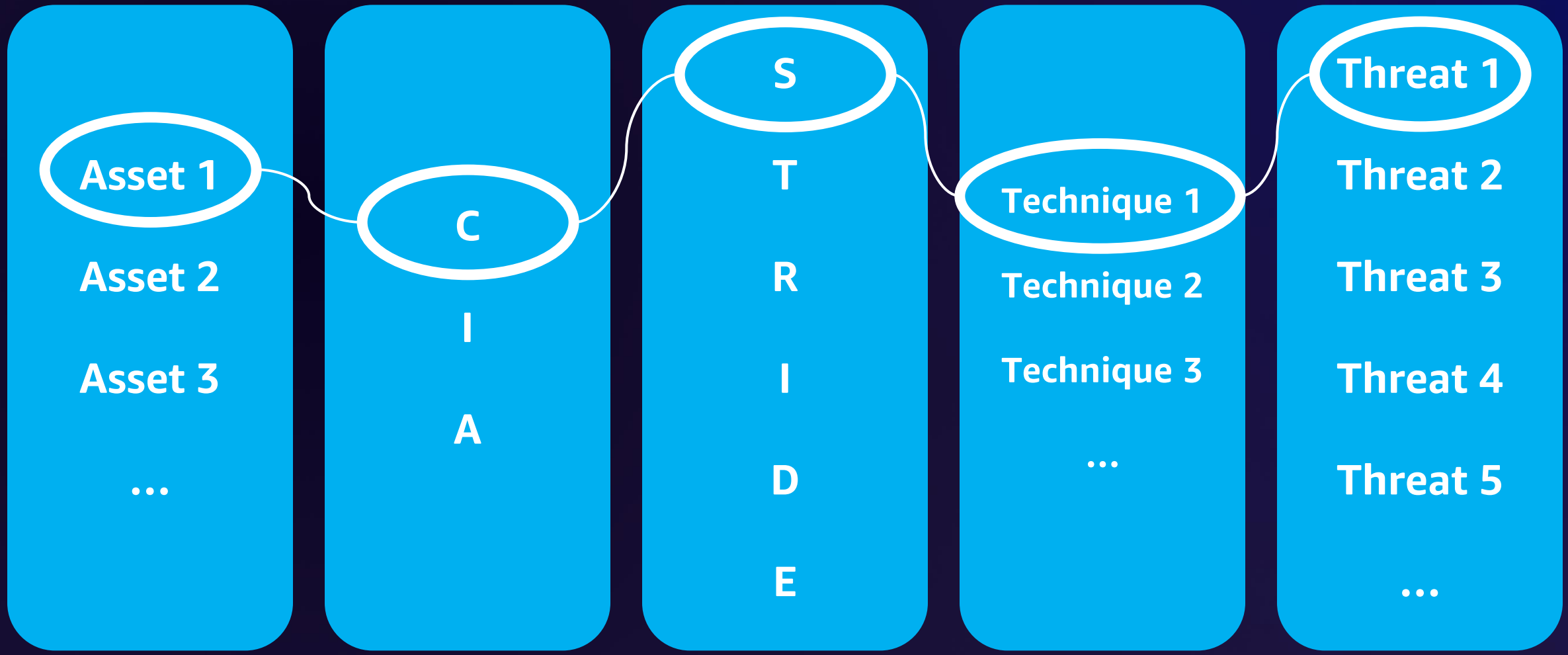


High-level attack steps

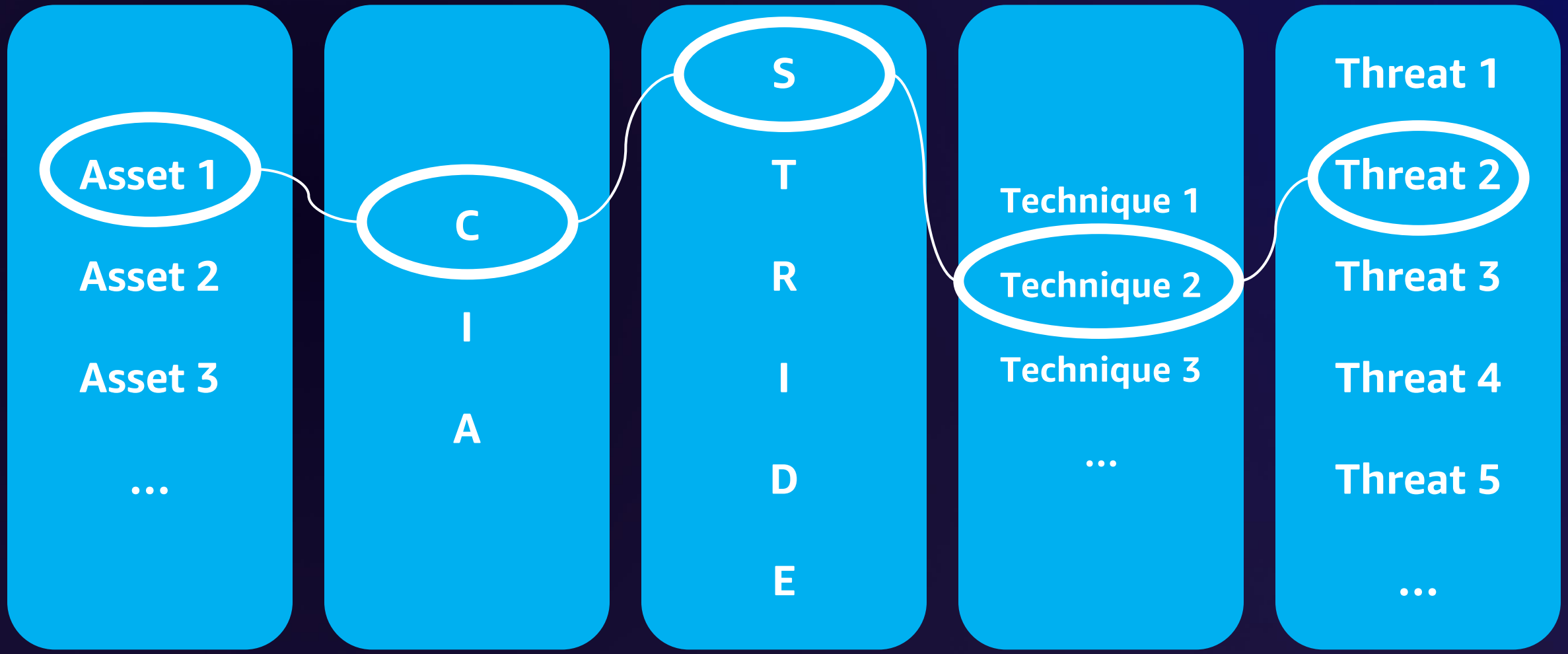


Why threat model this way?

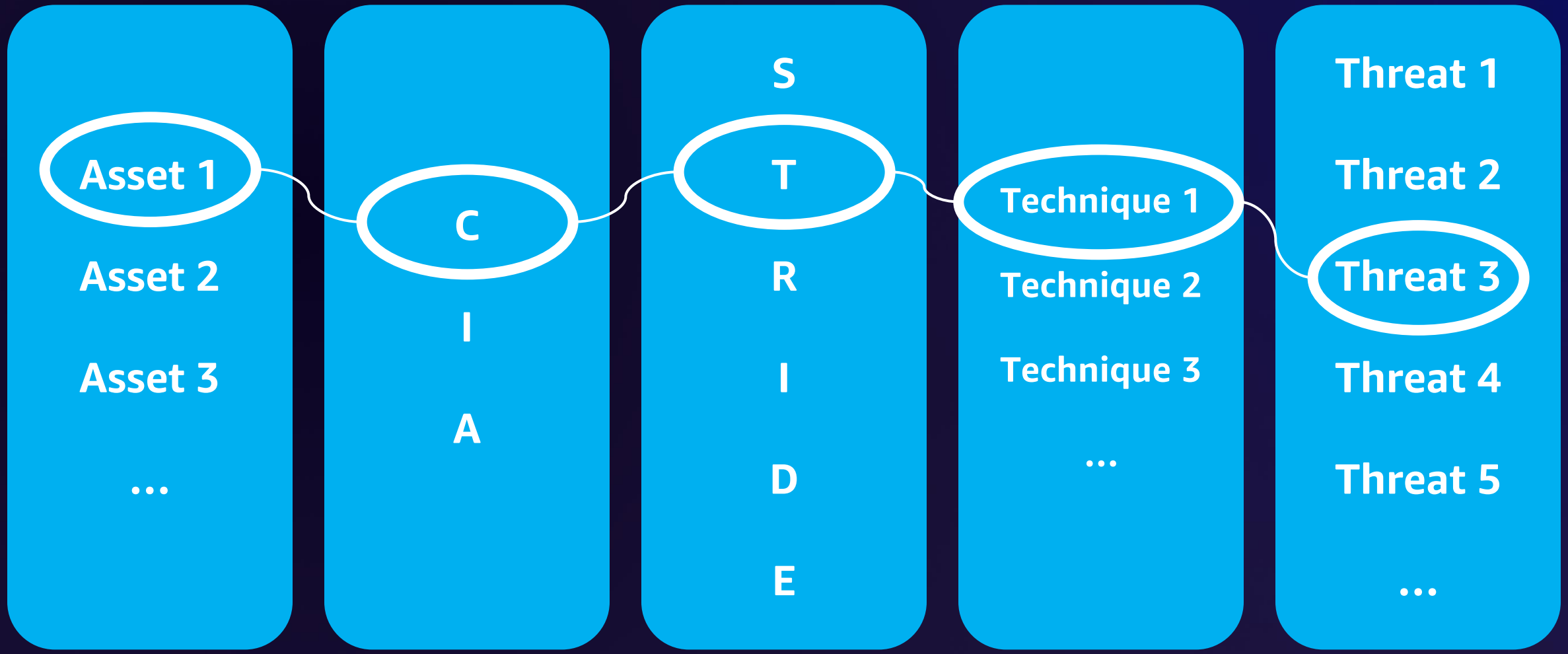
What threats should we care about?



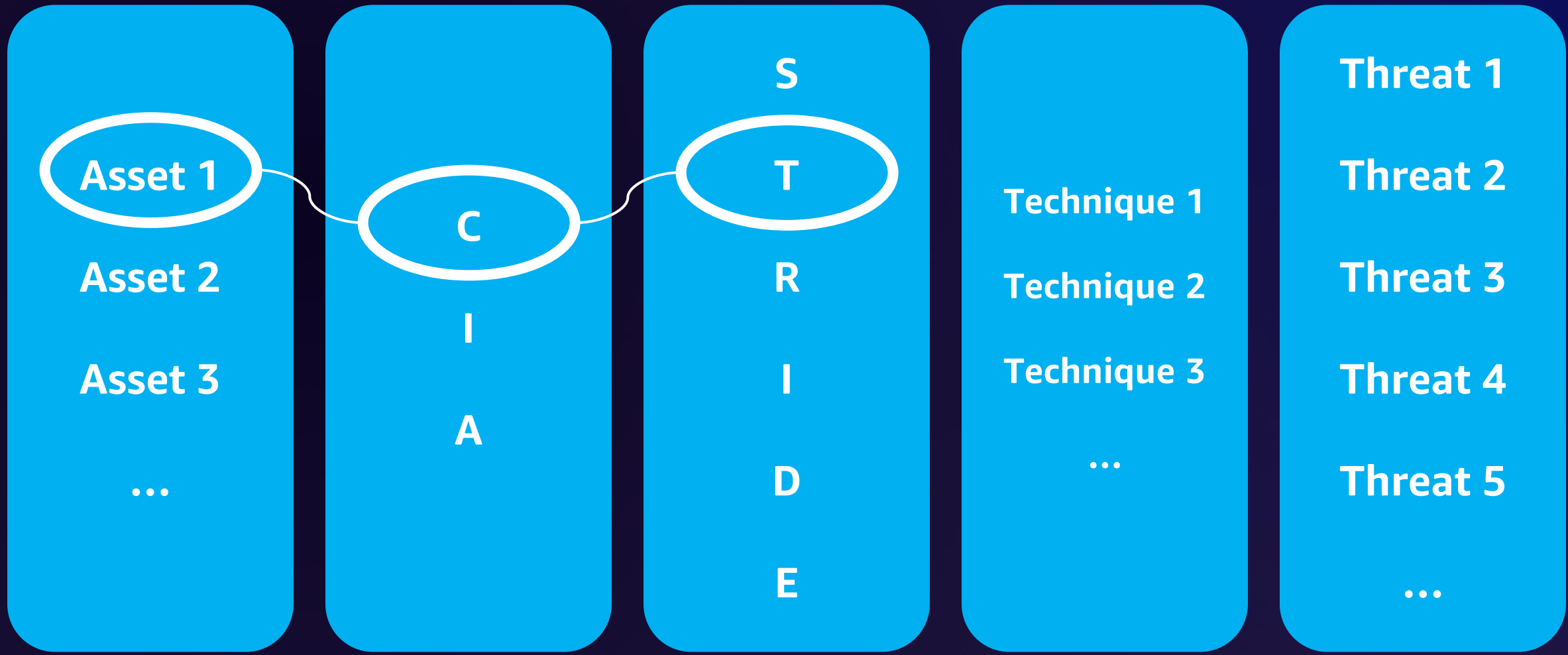
What threats should we care about?



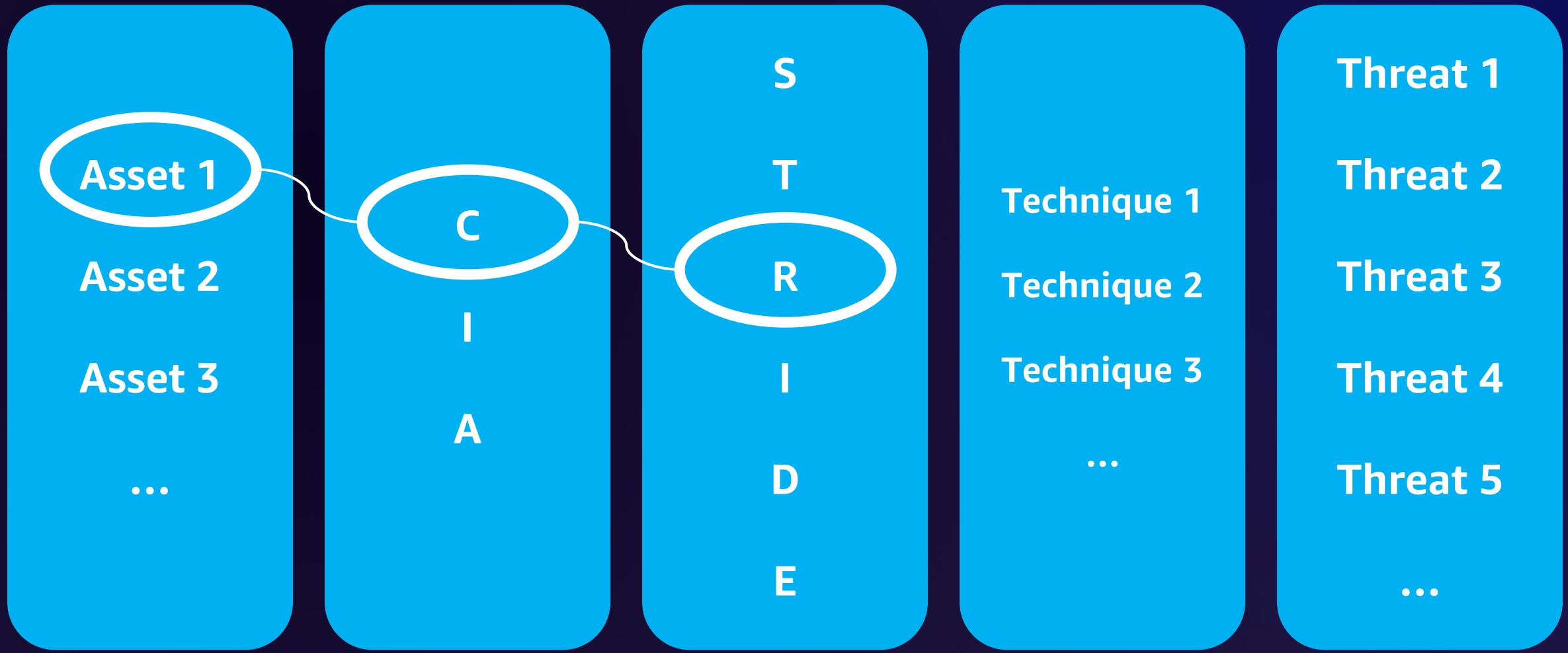
What threats should we care about?



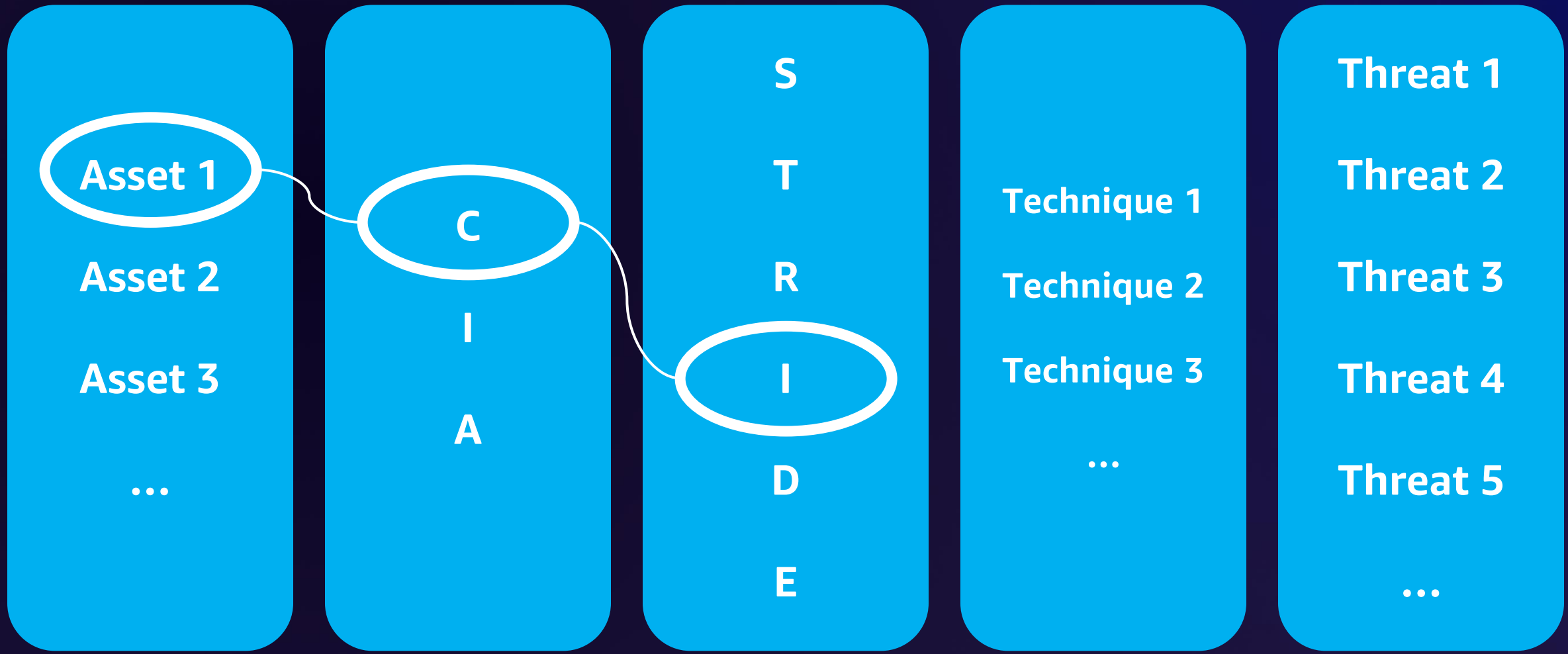
What threats should we care about?



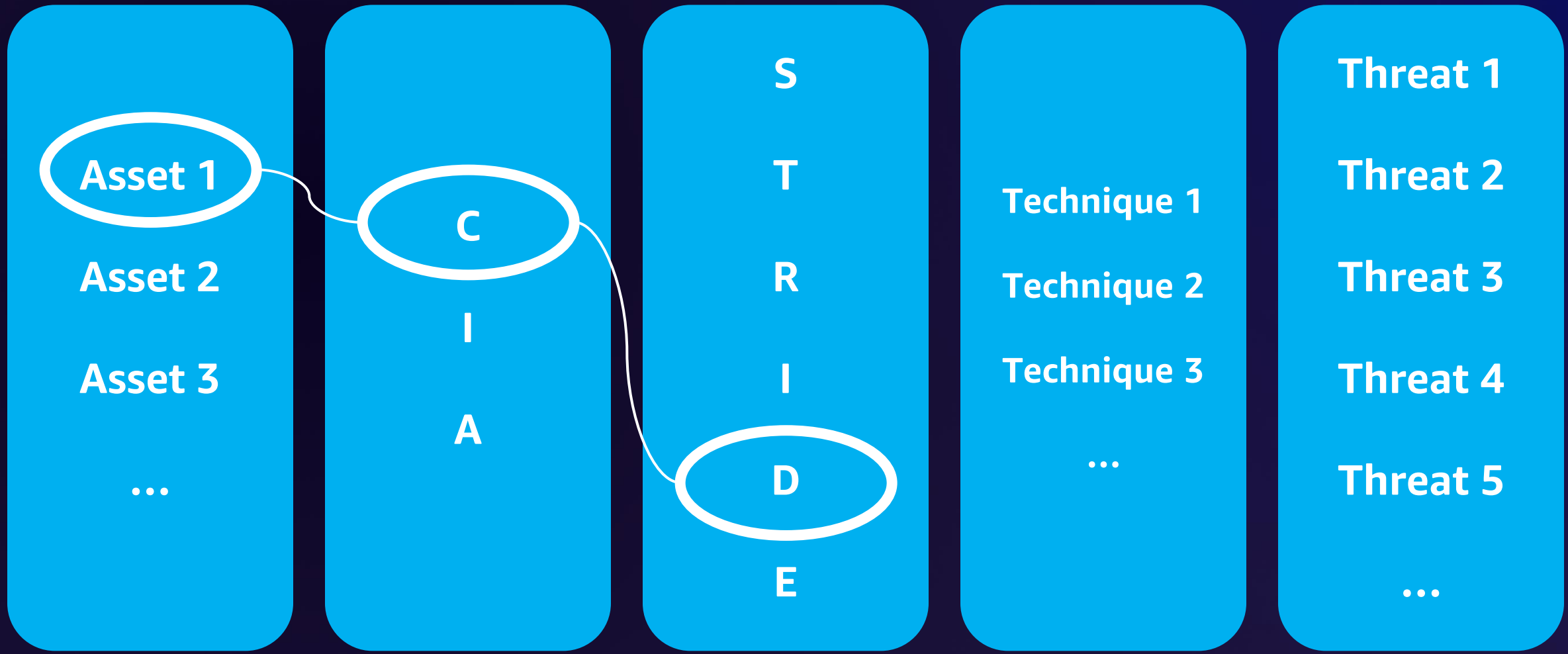
What threats should we care about?



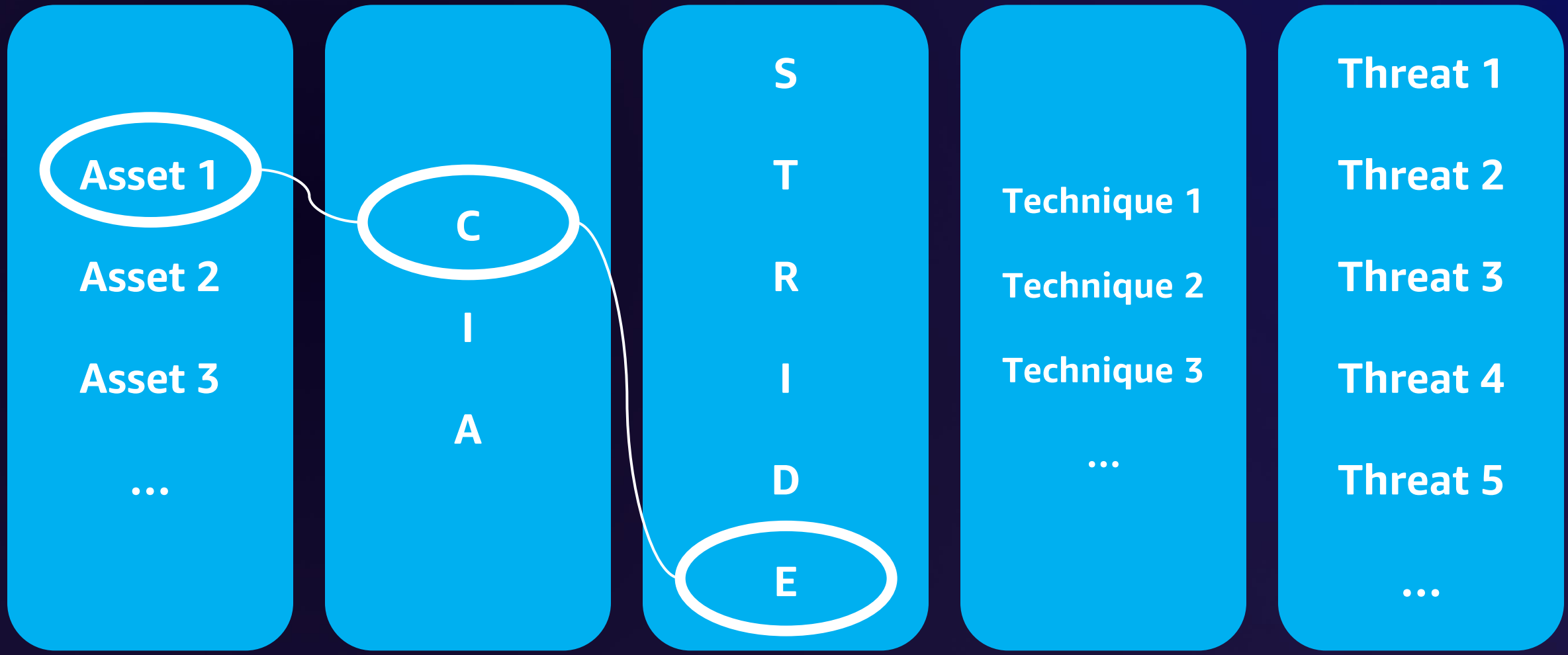
What threats should we care about?



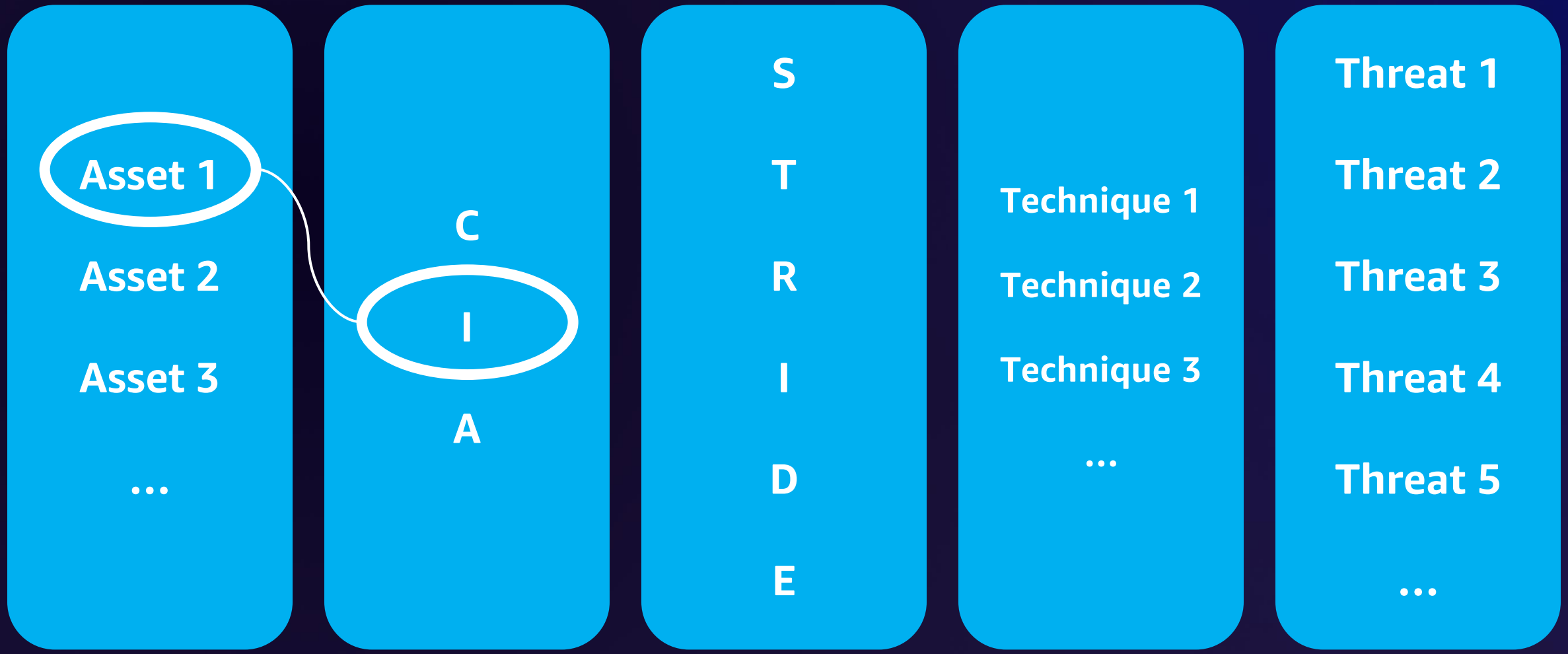
What threats should we care about?



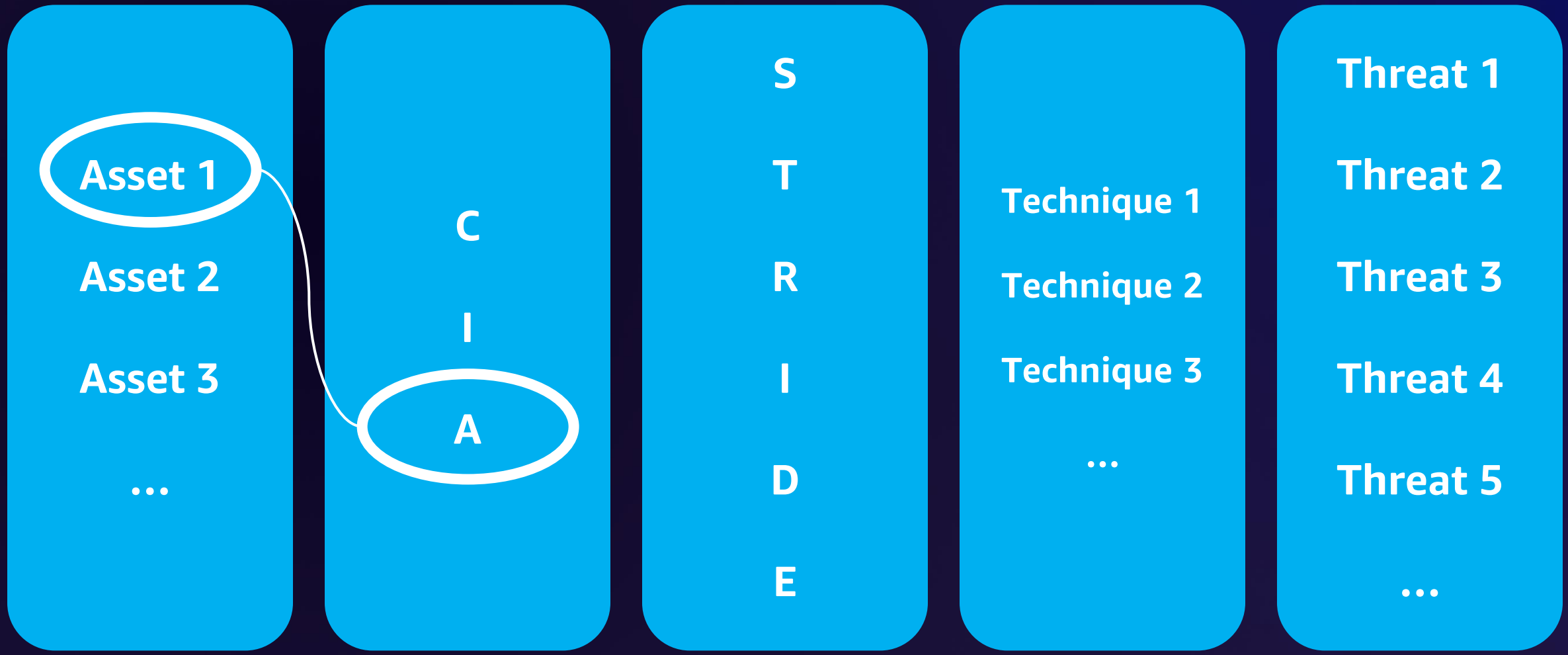
What threats should we care about?



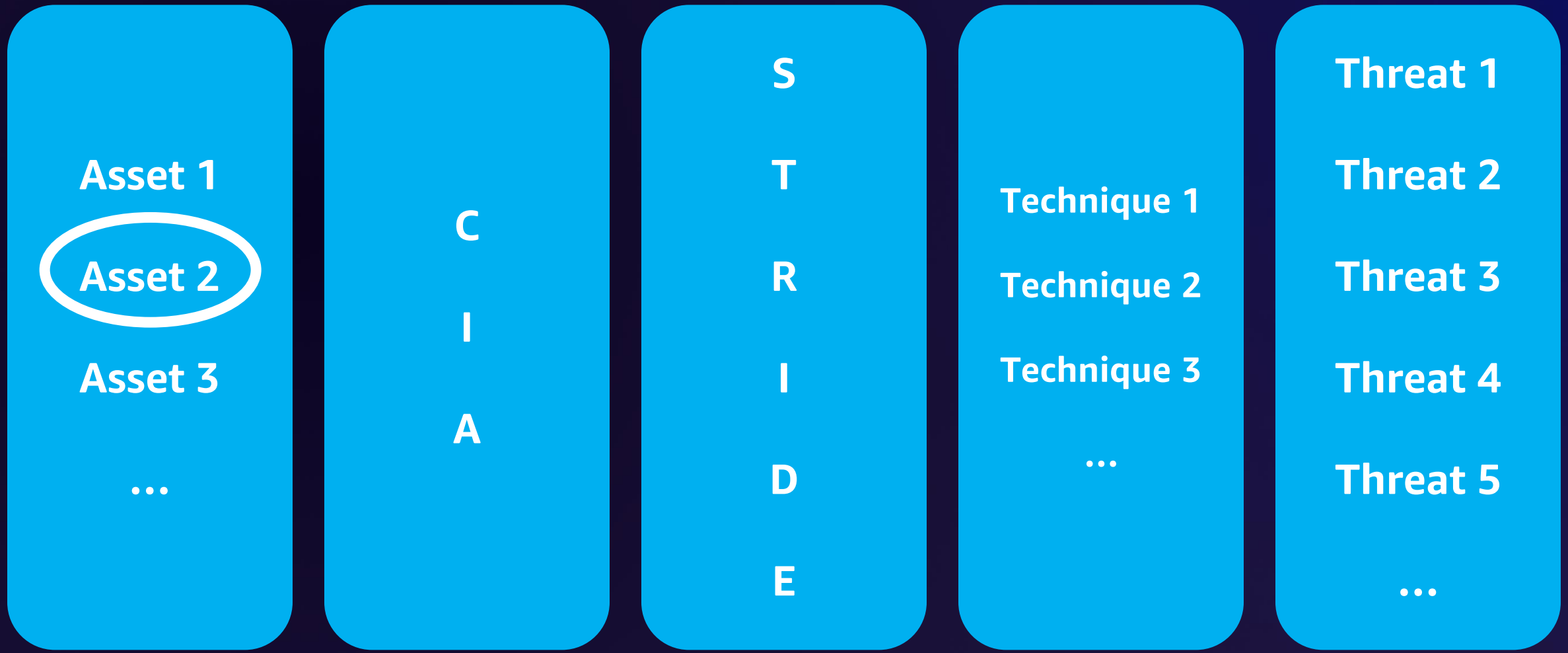
What threats should we care about?



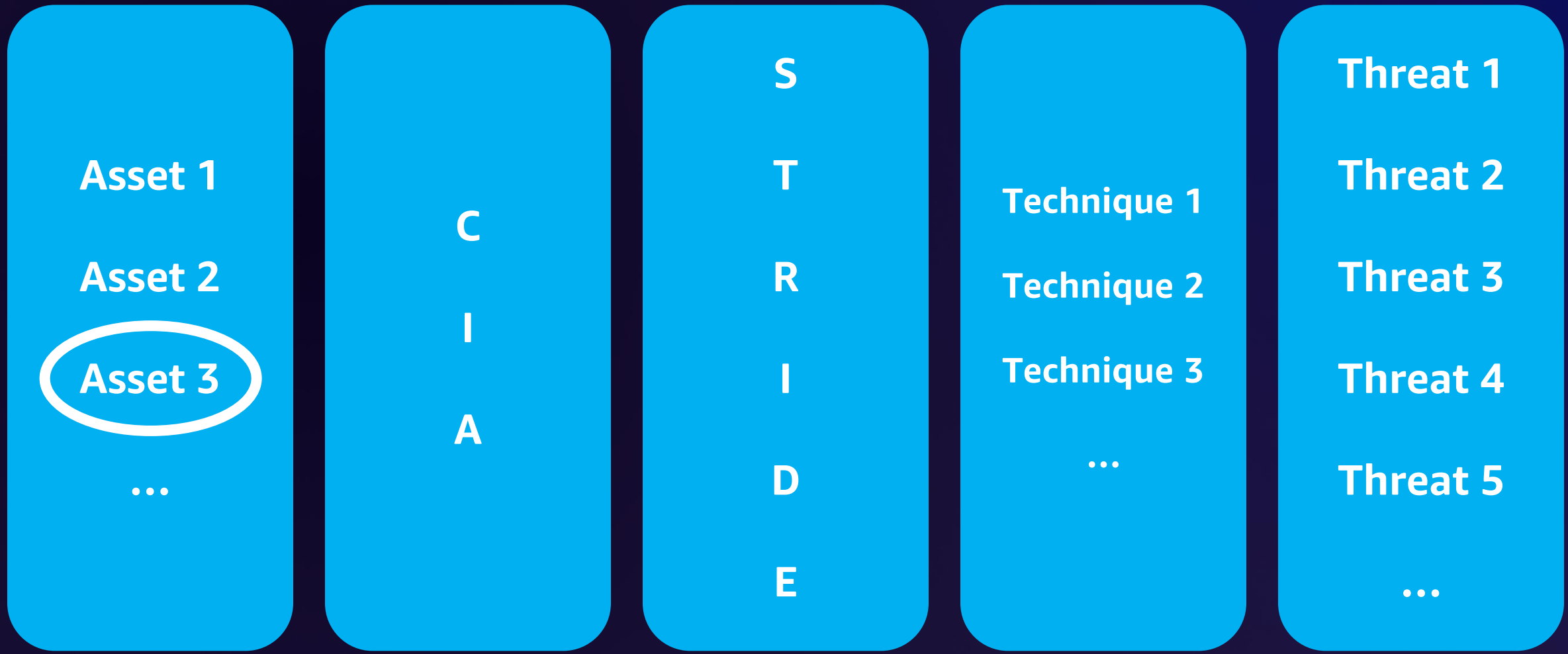
What threats should we care about?



What threats should we care about?



What threats should we care about?



Workshop
Threat modeling for builders

Framework
OWASP Top 10 for LLMs

Tool
Threat composer

Framework
Generative AI Security Scoping Matrix

Resource
Full threat model of example generative AI chatbot

Framework
MITRE ATLAS