January Community Meetup

THREAT MODELING CONNECT
POWERED BY IRIUSRISK

High Assurance Threat Modeling

January 19th | 11:00 ET

Hosted by:

Dave Soldera
Security Architect / Electronic Arts

About the Community Meetup

- **Our Goal**
  Exchange real-world experience, share practical knowledge, validate ideas to improve our own practice.

- **Chatham House Rule**
  Participants are free to use the information received, but neither the identity nor the affiliation of the speaker(s), nor that of any other participant, may be revealed.

- **Video is optional but highly recommended :)**

📅 Agenda of Today

11:00	Welcome, intro, photo
11:05	Presentation
11:35	Open Q&A, Discussion
11:55	Closing & Announcements

Photo Time!

About me

Dave Soldera
Security Architect
Electronic Arts

- Research / Developer / Pen-tester (~10 years)
- Security architect – whatever this means (~12 years)
- Consider myself mostly an appsec person

Disclaimer: Opinions are my own and not necessarily the views of my employer

Contents

- Context
- Origin
- Method
- Value
- Takeaways

Context

High Assurance Threat Modelling?

- Program or Process?
  - Focus is on the actual process of creating a threat model
  - Not focusing on ‘assurance’ of the threat modelling program
- What sort of Threat Modelling?
  - Application/Product security
  - Mostly backend/server systems, small or large, some client systems, some infrastructure
  - Creating a structured document (not brainstorming, or a stand-up chat)
- Assurance of what?
  - Assurance of scope - I can have confidence I have identified everything that is relevant to the system
  - Not assurance of threats
    - Probably can’t have this without scope assurance
    - It could be a whole other talk
- What makes it “High”?
  - It’s an actual part of your threat modelling process, rather than hoping nothing is missed

Spoiler

The basic idea is …

Consistency = High Assurance

By “consistency” I mean

“consistency of the internal references within your threat model”

Could also be called “referential integrity”.

When we capture data in various parts of a threat model, if it references ‘something’, then that ‘something’ must also be captured.

For example:

If capturing first/lastname assets as stored in a database

Then

The database must exist as part of the scope

If the database wasn’t captured we would say the asset capture was ‘inconsistent’ with the scope defined.

Obviously

There are lots of ‘obviously’ good things, that we only do a subset of, as part of our own threat modelling process, like capturing:

- Actors
- Roles
- Boundaries
- Entry Points
- Trust Levels
- Data Classification
- CWE / CVSS
- Attack Trees
- Adversary Motivations
- Tech Stack (frameworks, libraries, etc)
- RACIs
- Maturity
- Exploit History
- Business Value
- Diagrams

So what makes us do some things and not others, when we have so many options?

⇒ <u>The activity must have “value” for the effort required</u>

The Point

This talk wants to convey to you:

- Driving consistency in your threat models has a lot of value (compared to all the other things you could do)
- A method of achieving consistency that delivers that value (at least in my experience)

Origin Story

Starting out

- Security Architect starting at a Healthcare Startup
- First dedicated AppSec person, they needed an SSDLC
- Threat Modelling was going to give best bang for buck

Sidebar

- Used MS Threat Modelling Tool since my pen-test days, always so noisy
- Straight STRIDE was too much to explain, too much like devs needed to be security people, or think "evil"
- Decided I wanted to make things easier by creating my own approach
- Influenced by SABSA, and it's structural approach to security
- Had been doing a lot of work capturing requirements, so using tables capture and present data
- Had started developing my own document template for threat modelling, and customising it for the business I was helping

Starting out

- Security Architect starting at a Healthcare Startup
- First dedicated app sec person, they needed an SSDLC
- Threat Modelling was going to give best bang for buck
- I had already been refining an approach to Threat Modelling that:
  - used a Confluence page as a template
  - devs would populate tables with components and assets
  - gathered information most devs could answer
  - focused on design issues
- Started out with a single friendly team
- Made any completed threat models available to everyone else

Bumpy

- First version asked for too much information
- Tried to incorporate ‘business goals’
- Was too far removed from devs day-to-day

- Over-optimised my template for microservices
- Didn’t ask about authentication, it was handled centrally
- Template was impractical for anything else the company built

QMS

- If you write software to be used as a medical device e.g. doctor video conferencing, then you need the software to be certified
- This is done through a Quality Management System

  > “a structured system that documents the procedures and processes implemented throughout the lifecycle of a medical device”

- Basically a document management system where you are audited by having to evidence that you actually do what the documents say
- Basically a management system for
  - the control of documents (what you say you WILL DO)
  - the control of records (evidence of what you ACTUALLY DID)
- Specifically you are audited against ISO 13485.
- ISO 13485 mentions ‘risk’, but generally looks to ISO 14971 “risk management to medical devices”

Lucky

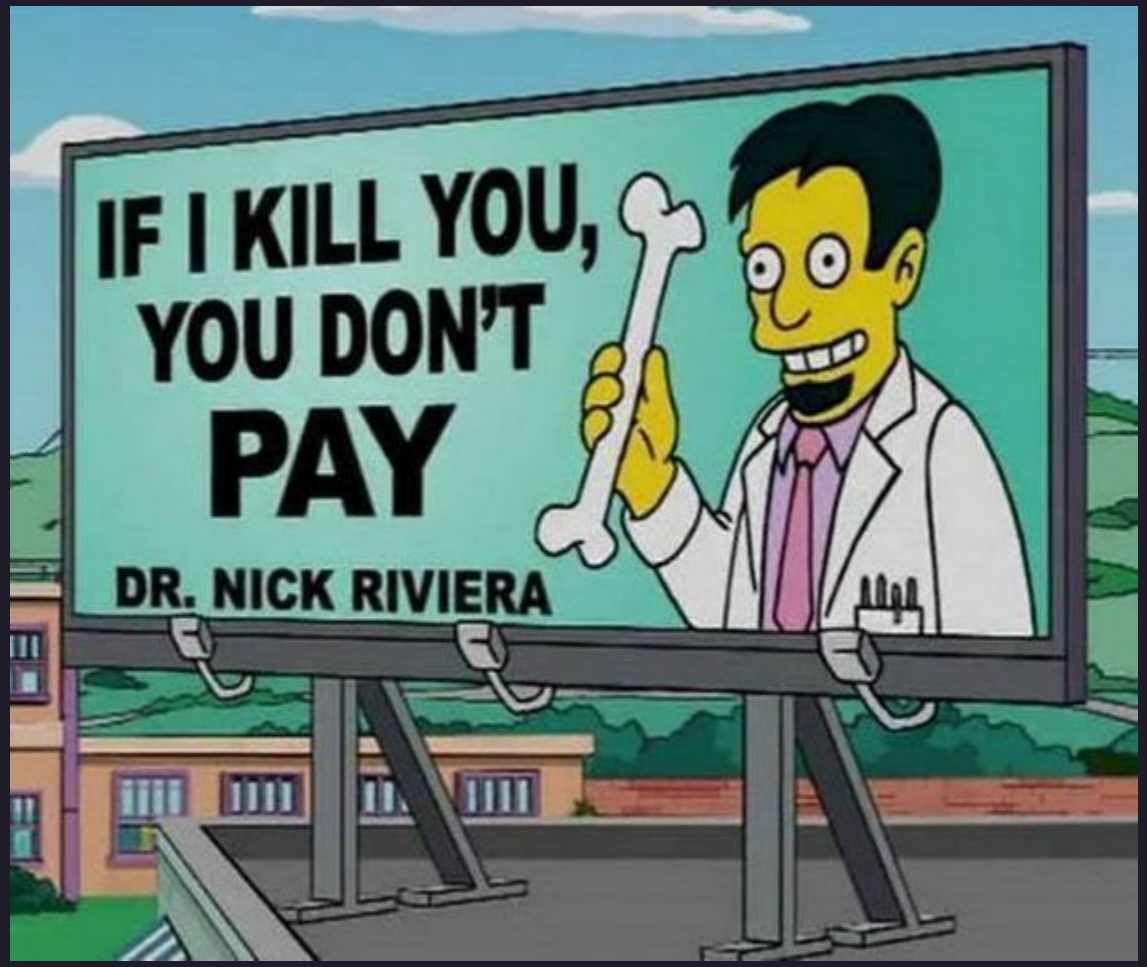
- ISO 14971 doesn't mandate threat modelling, but we decided threat modelling would significantly contribute towards complying with it
- Recipe for success
  - We already had a policy that said we would do threat modelling
  - We were going to be audited against what our policy said
  - We would need to provide evidence during the audit

  ⇒ Dev teams had to do threat models for the software to be certified as a medical device
- Yay! I win threat modelling!

… wait a sec … my threat models WILL BE AUDITED!!!! Oh ****!!!

Guess I better find a way to give myself some *assurance* they will pass an audit

Method



High Assurance Threat Modelling

- Time to come clean
- I didn’t so much “design” assurance into the process
- I more “discovered” it as I went along
- Going to present the idea as a “retrospective design” - as if I planned it all along

Mistakes

Situation: You have defined a threat modelling process. You devs follow this process to create threat models.

But when a threat model contains mistakes, how would your devs know?
How would you know?
When would a mistake be spotted?

People make mistakes. It will happen. Our process needs to accommodate for this fact.

The specific mistake we're focusing on is - failing to include in the scope relevant parts of the system.

Option A: Design a process where people cannot make mistakes.

- If you figure out how to do this, please tell me how?

Option B: Design a process where mistakes can be detected.

☑️☑️☑️ & ⚖️ ⚖️ ⚖️

- Credit card numbers use Luhn's algorithm
- Double entry accounting
- Separate interrogation of suspects
- Video conferencing tool message “You’re talking, but you are on mute”
- Random sampling in mass production
- Smoke alarms
- Audit

Tables

We can detect inconsistencies by capturing information in multiple places that must reference information in those other places.

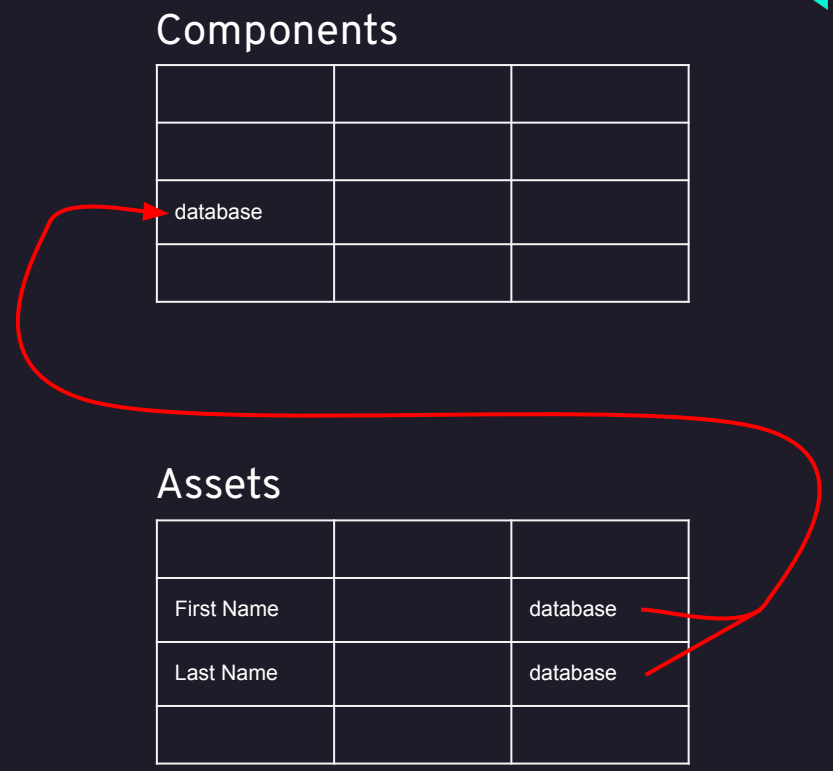
For example:

If capturing first/lastname assets as stored in a database

Then

The database must be listed as part of the system

Capturing data in tables makes for a convenient method to spot inconsistencies.



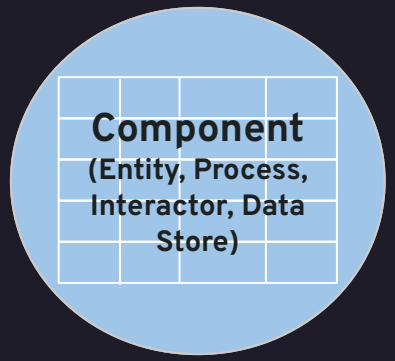
But not just Tables

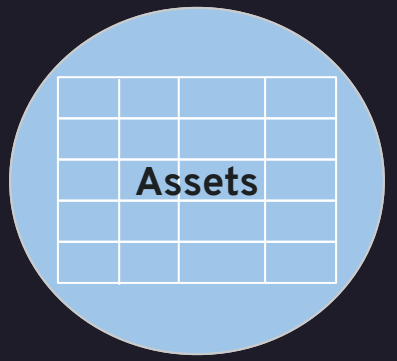
I used a templated Confluence page (but any doc format would work) to essentially capture:

- A table of the different parts of the system, both in and out of scope e.g. Components
- A table of assets processed or stored by the system
- A table of threats and controls

But you can actually use multiple places in your threat model to detect inconsistencies.

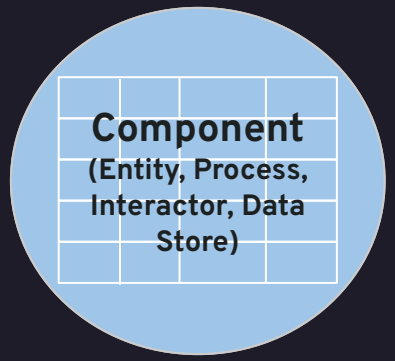
Let’s visualise the ones I used …


Component
(Entity, Process, Interactor, Data Store)


Assets

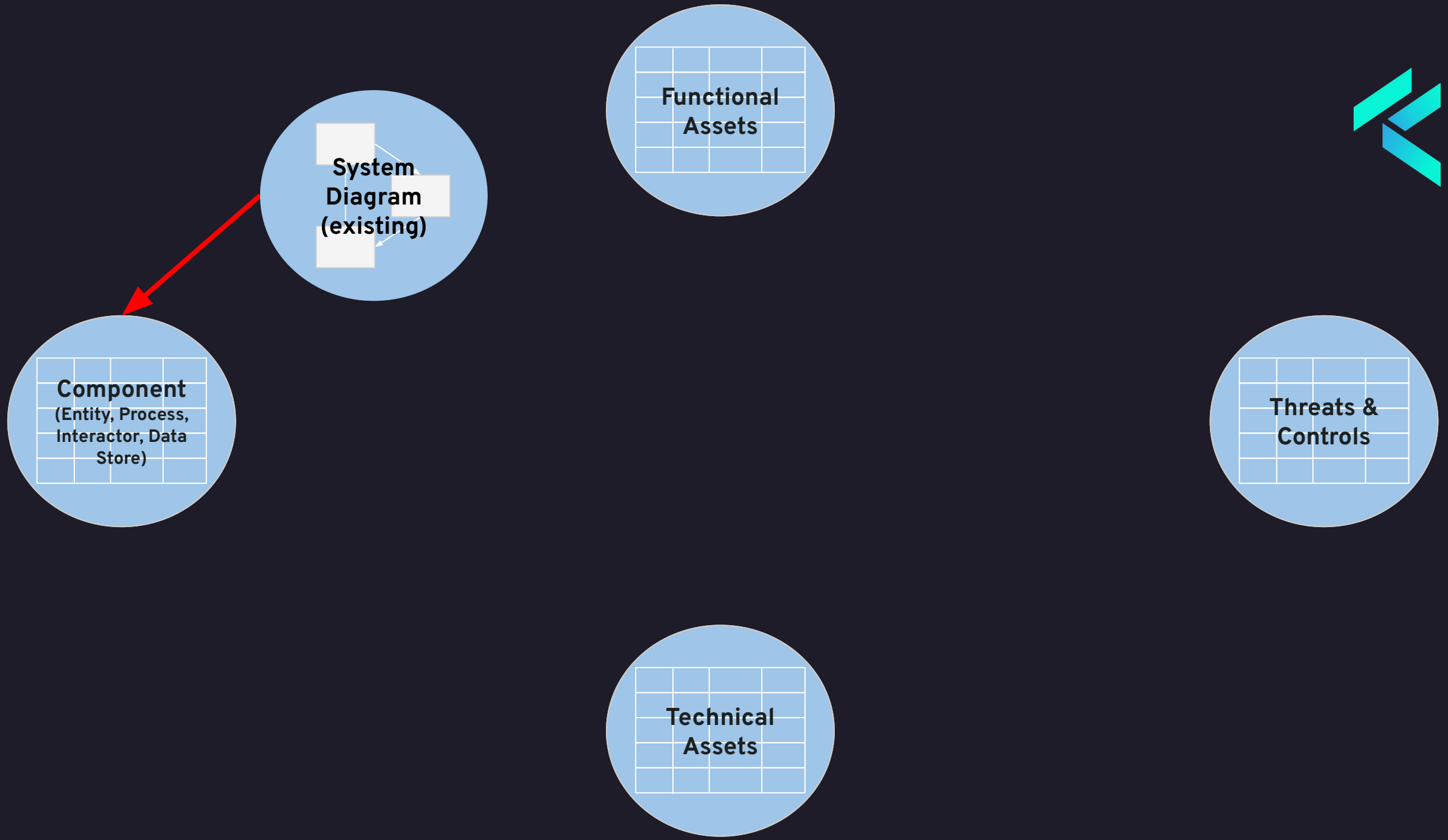

Threats &
Controls

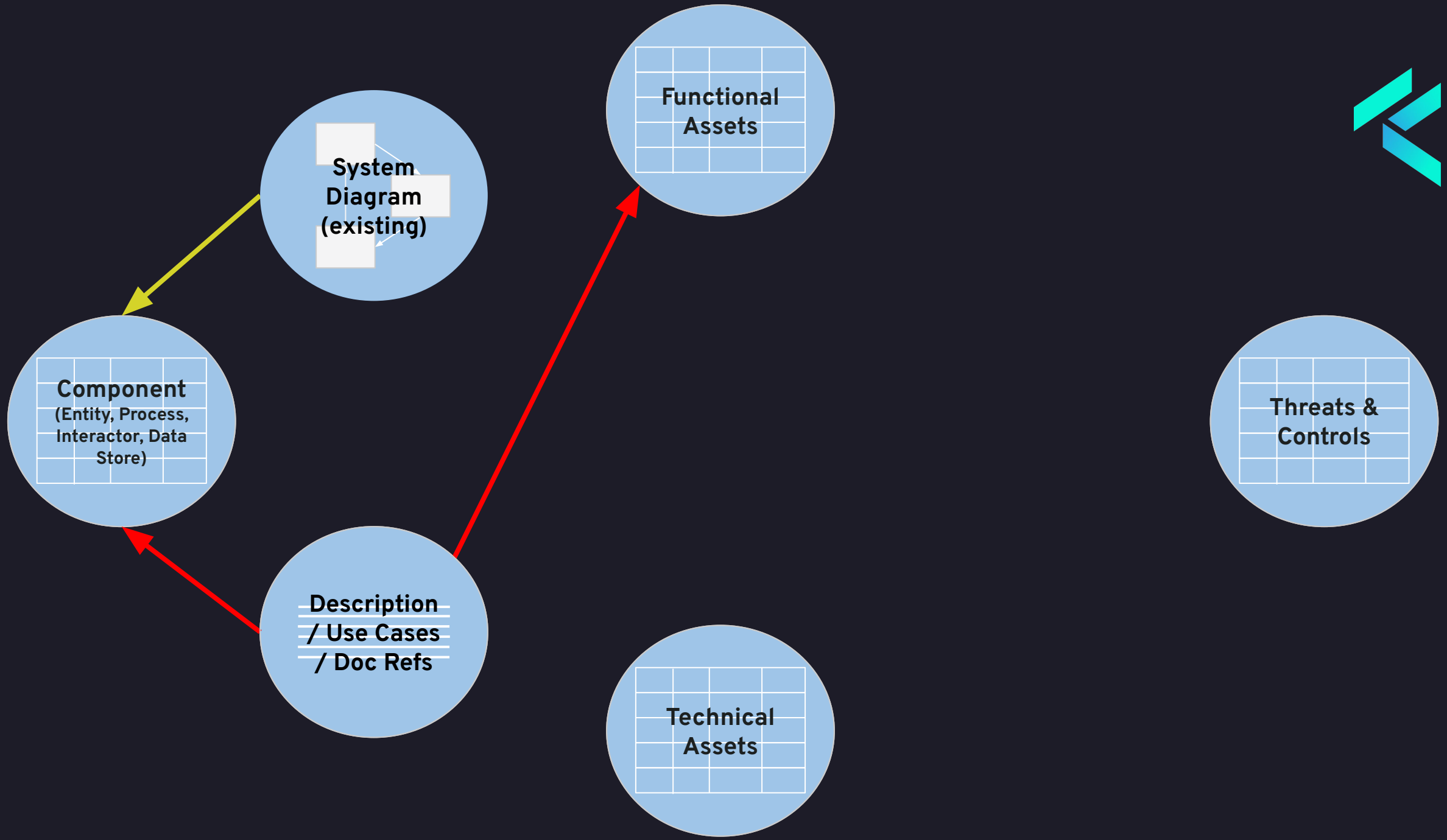

Functional
Assets

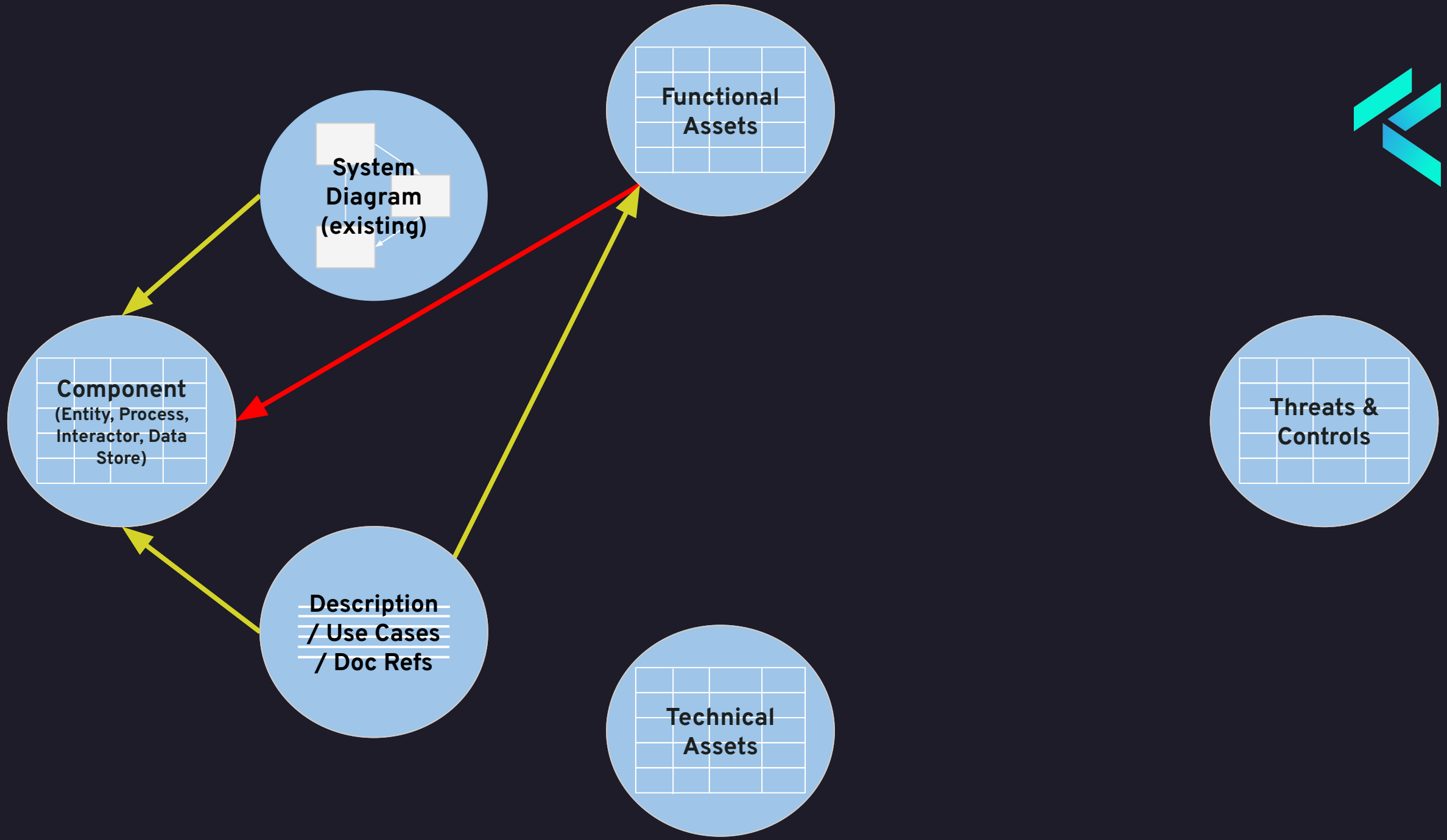

Component
(Entity, Process,
Interactor, Data
Store)

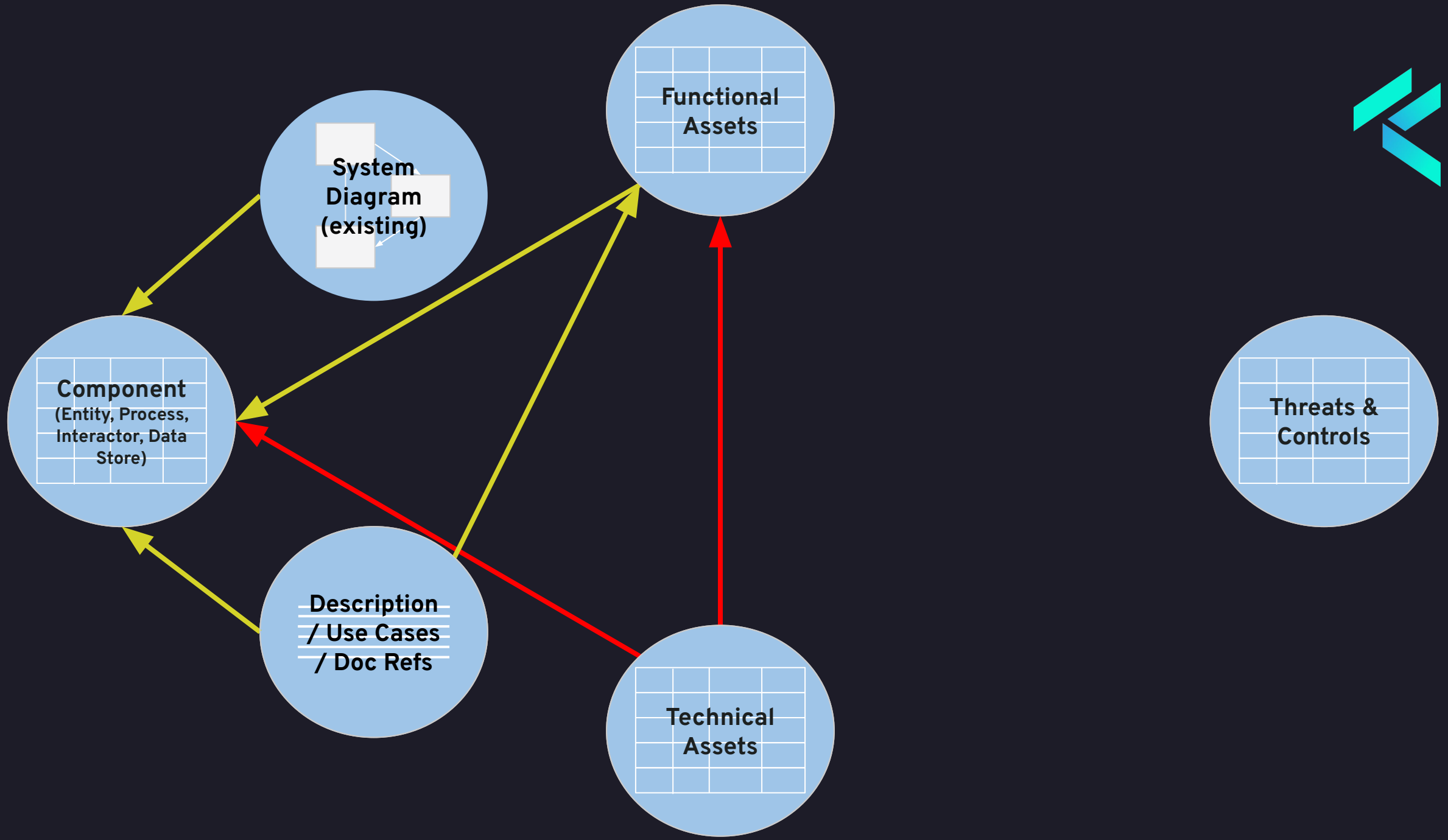

Threats &
Controls

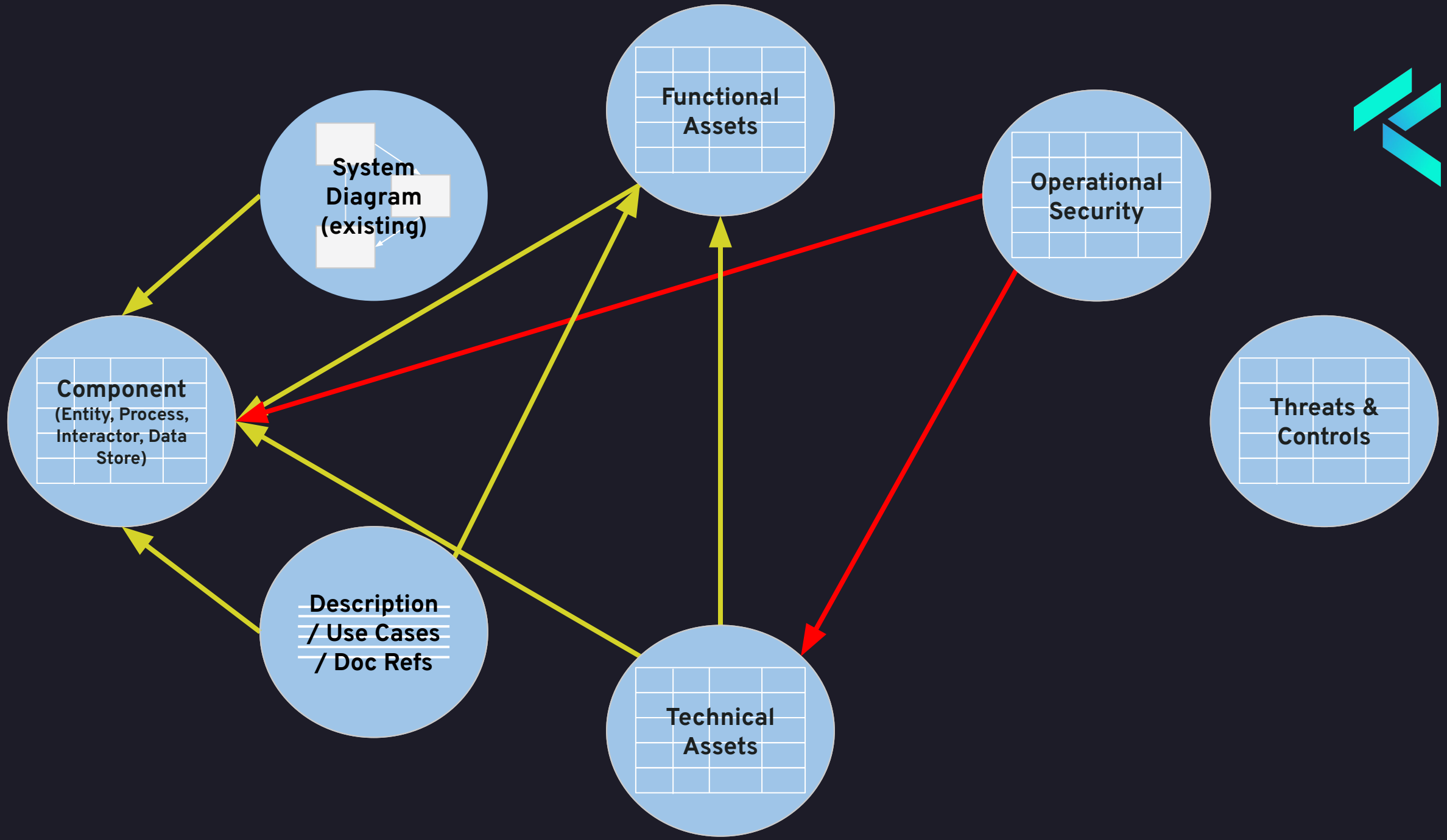

Technical
Assets

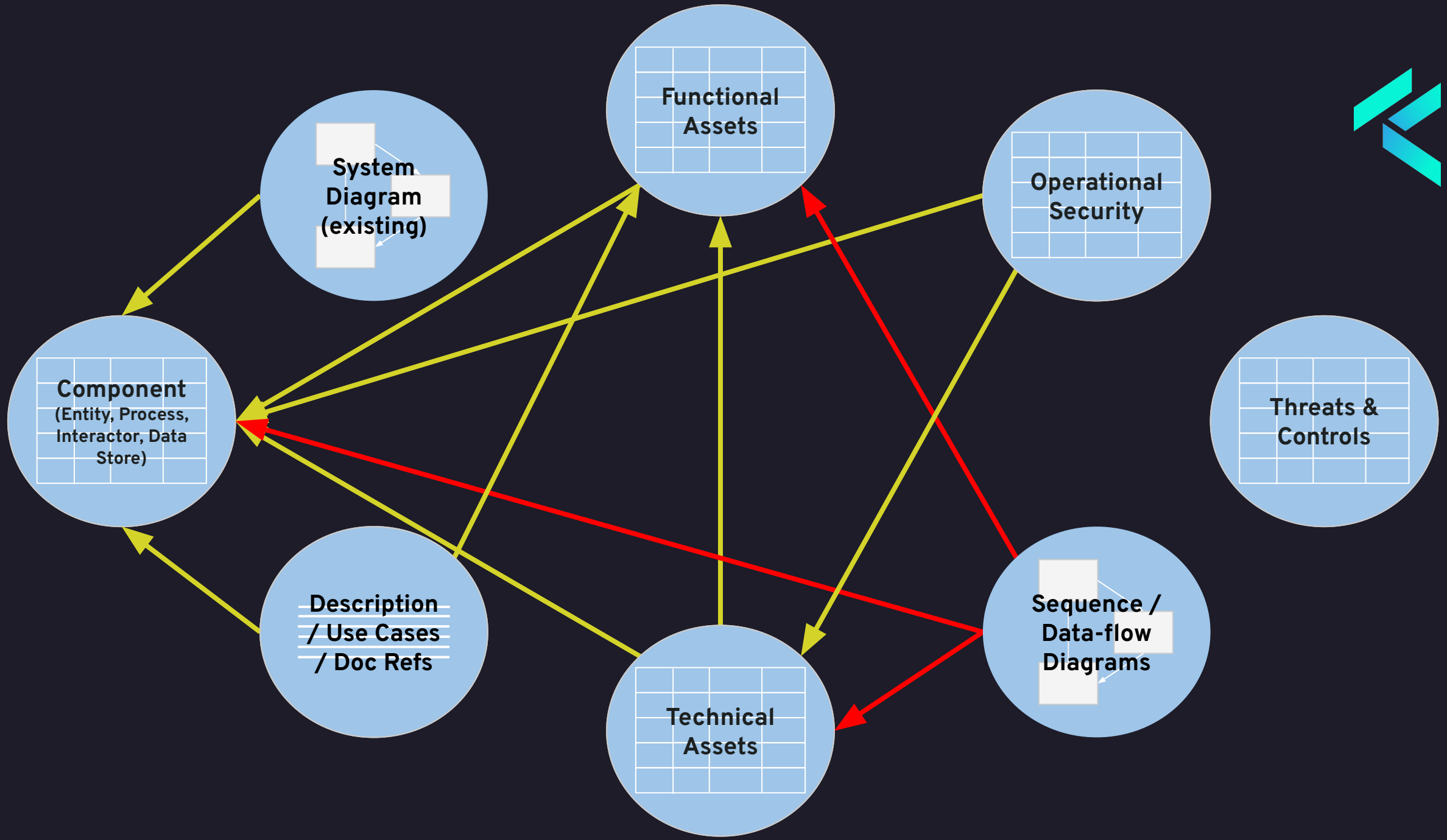

System Diagram (existing)
Functional Assets
Component (Entity, Process, Interactor, Data Store)
Threats & Controls
Technical Assets

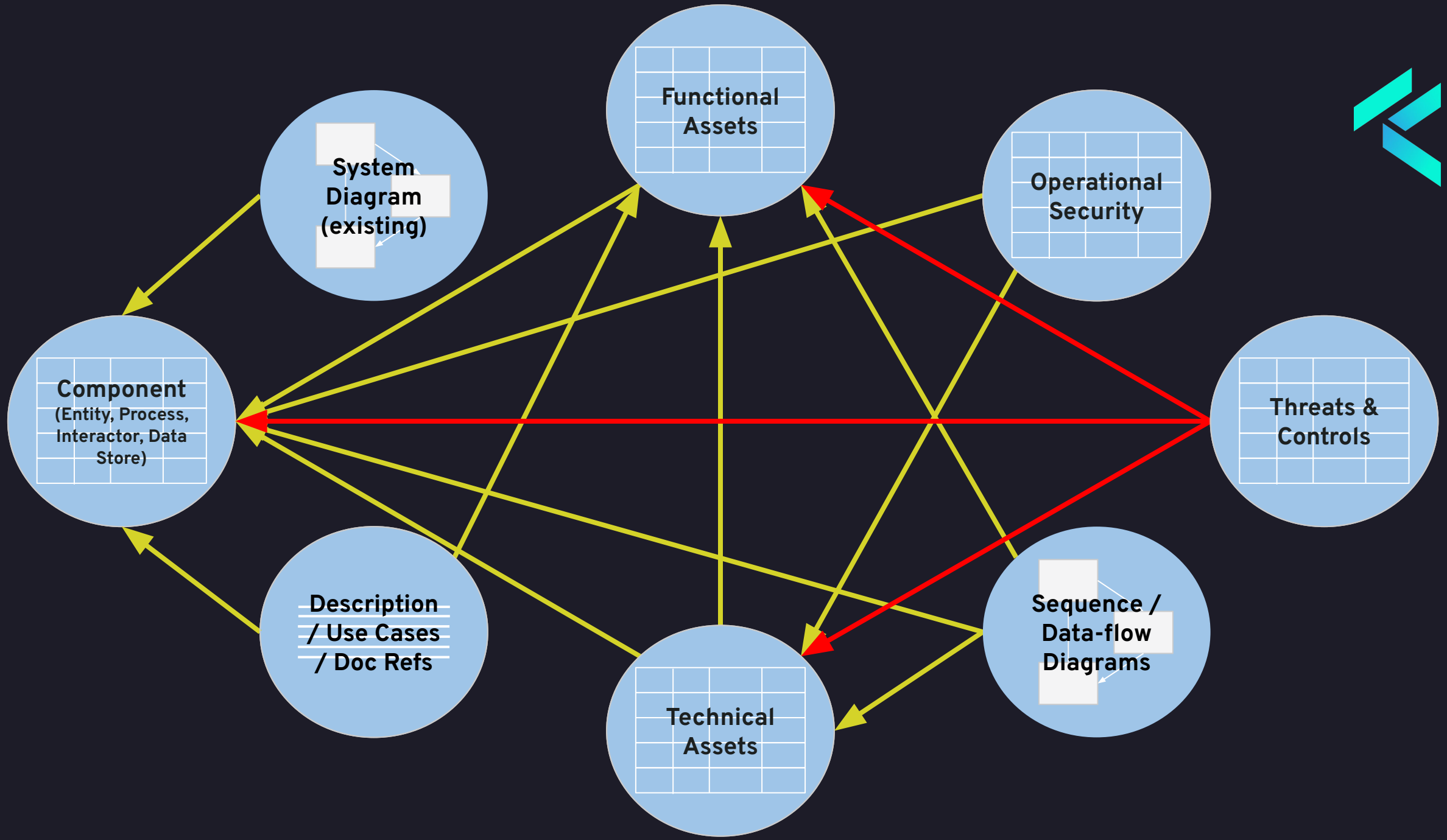

System Diagram (existing)
Functional Assets
Component (Entity, Process, Interactor, Data Store)
Threats & Controls
Description / Use Cases / Doc Refs
Technical Assets

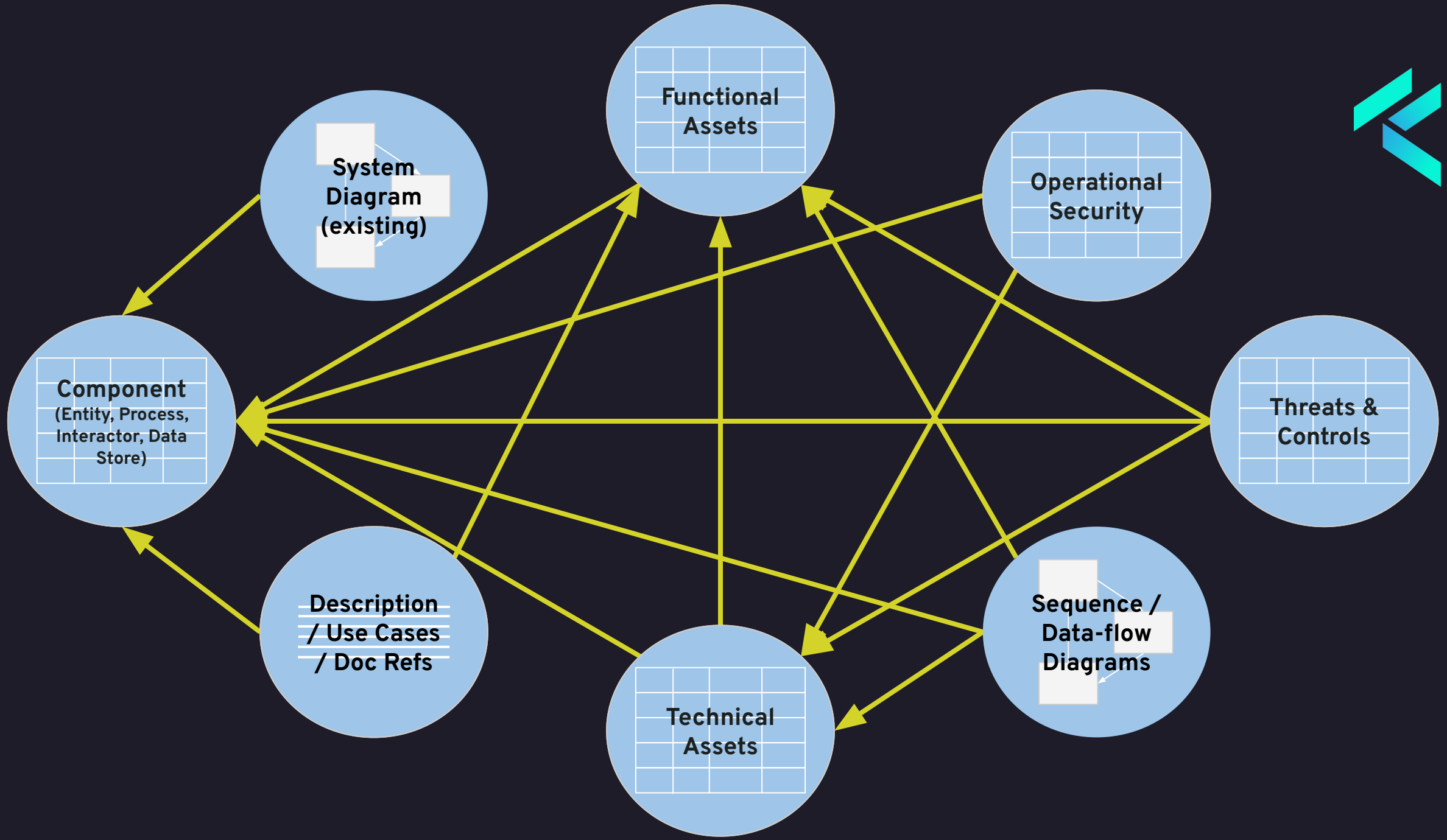

System Diagram (existing)
Functional Assets
Component (Entity, Process, Interactor, Data Store)
Threats & Controls
Description / Use Cases / Doc Refs
Technical Assets


Functional Assets
System Diagram (existing)
Component (Entity, Process, Interactor, Data Store)
Threats & Controls
Description / Use Cases / Doc Refs
Technical Assets

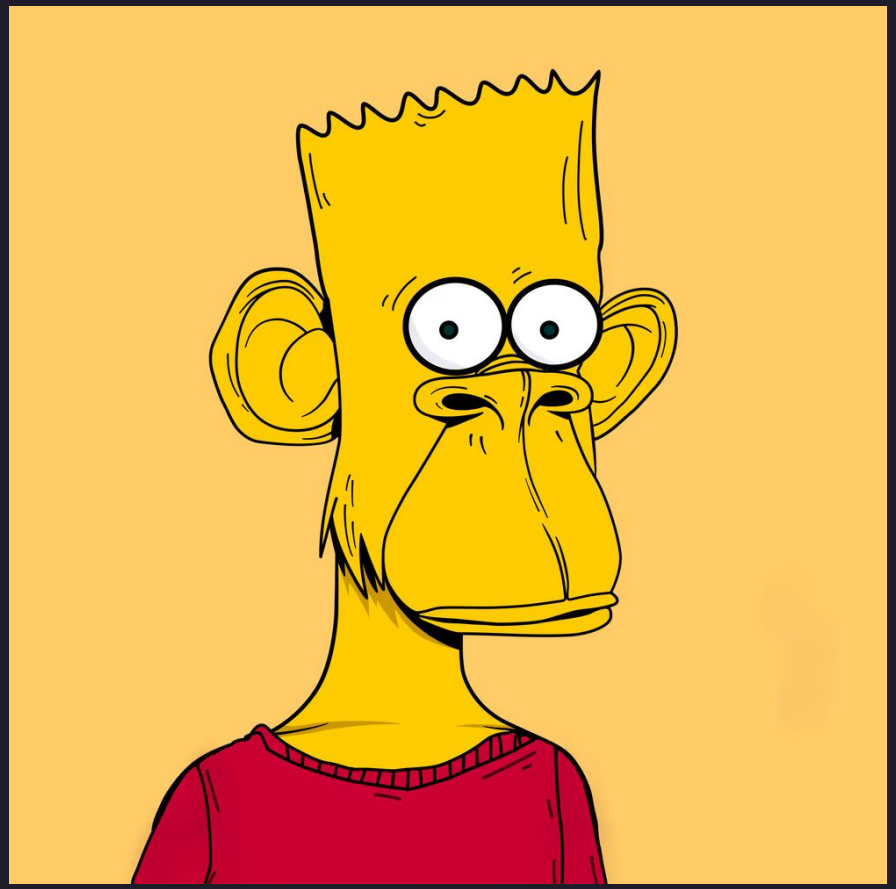

Functional Assets
System Diagram (existing)
Operational Security
Component (Entity, Process, Interactor, Data Store)
Threats & Controls
Description / Use Cases / Doc Refs
Technical Assets


Functional Assets
System Diagram (existing)
Operational Security
Component (Entity, Process, Interactor, Data Store)
Threats & Controls
Description / Use Cases / Doc Refs
Sequence / Data-flow Diagrams
Technical Assets


Functional Assets
System Diagram (existing)
Operational Security
Component (Entity, Process, Interactor, Data Store)
Threats & Controls
Description / Use Cases / Doc Refs
Technical Assets
Sequence / Data-flow Diagrams


Functional Assets
System Diagram (existing)
Operational Security
Component (Entity, Process, Interactor, Data Store)
Threats & Controls
Description / Use Cases / Doc Refs
Technical Assets
Sequence / Data-flow Diagrams

Value



Resist

Some temptations to resist if you try to add this to your own process:

- Numbering - One way to capture information is to assign it individual identifiers e.g. component 1.3, asset 3.6

  This can lead to crazy situations where you end up with something like:

Component	Asset	Threat	Control
1.5	2.6	3.7	4.19

  Which makes absolutely no sense.

  Certainly capture information in a structured way, but don't sacrifice comprehension for structure.

- No Reviews - It's still possible for devs to arrange a consistent threat model that actually has an incomplete scope. This can usually be done by removing components, which is not what we want!

  Review by someone from the Security team (or a Security Champion) is essential, and common sense needs to be applied.

  This hasn't happened very often to me, perhaps only a couple of times.

Speed

Depending on the number of Components, Assets, Threats and Controls, driving consistency in a threat model can be a time consuming task.

Tactics:

- Control the scope (i.e. size) of threat models so consistency isn't difficult to achieve
  - "Nothing smaller than a repo, nothing larger than a dev team"
- Consistency stops at 'out-of-scope' components
- Pre-populate threat models for different types of solutions e.g. microservice, web app etc.
- Allow flagrant copying of existing threat models
- Tooling can help identify inconsistencies
  - But your current tooling might not do this
  - I wrote my own tooling to help
- Update threat models when design change happens (or at least annually) e.g.
  - new Components
  - new Functional Assets

Result

- My high assurance threat models passed all audits (for medical devices)
- Other consumers of the threat models
  - Internally - the threat models became some of the best documentation for how our systems worked
    - Accurate documentation is a by-product of consistent documentation
  - Investor due-diligence - “some of the best threat models they had seen”
  - Occasionally shared with customers, who were also happy with them

Good

“Did we do a good job?” from the 4-question framework:

- Often more a threat model program question
- Can we answer this question for a particular threat model?
- Part of a reasonable definition of “good” should include “consistent”
- “Consistency” is an objective property of the threat model, so

⇒ I would say if your threat model is “consistent” it is objectively “good” in some sense

Consistency also can help define the ‘definition of done’ for a threat model.

Stretch goal - What are other objective properties we want in threat models?

- I can think of “coverage” i.e. we have threats covering all Components and Assets
- Are there others?

Takeaways

Takeaways

- Because people make mistakes, the scope of your threat model may be missing things
- To get assurance nothing is missing, we need to be able to detect mistakes
- Consistency is a property of threat models we can use to detect mistakes
- Consistency can be achieved by:
  - capturing system information independently in different parts of your threat model
  - actively cross-referencing that information in those different parts of your threat model
- Ideally find a tool that can help spot inconsistency, otherwise include consistency as an activity when reviewing threat models
- Value:
  - Helps ensure your scope is not missing things (and hence not missing threats)
  - Helps produce accurate and consumable threat models
  - Helps to answer “Did we do a good job?”
  - Helps to define a ‘definition of done’ for a threat model

References

- Threat Model Template (http://tinyurl.com/threat-model-template) example (a Google Doc), shows
  - Overall layout
  - Different tables used
  - Columns in each table capturing information
- threatware.readthedocs.io for
  - A tool to automate some consistency and coverage checking for the template
  - More information on the approach to Threat Modelling that I use
  - Detailed documentation on how to populate the Threat Modelling Template

The template, documentation and tooling are all open source - take what you need.

Questions?

Discussion

Question 1:

How have others incorporated consistency (or achieved assurance) in your threat modelling process? What challenges did you face?

Question 2:

Are your threat models audited? How does that change your approach?

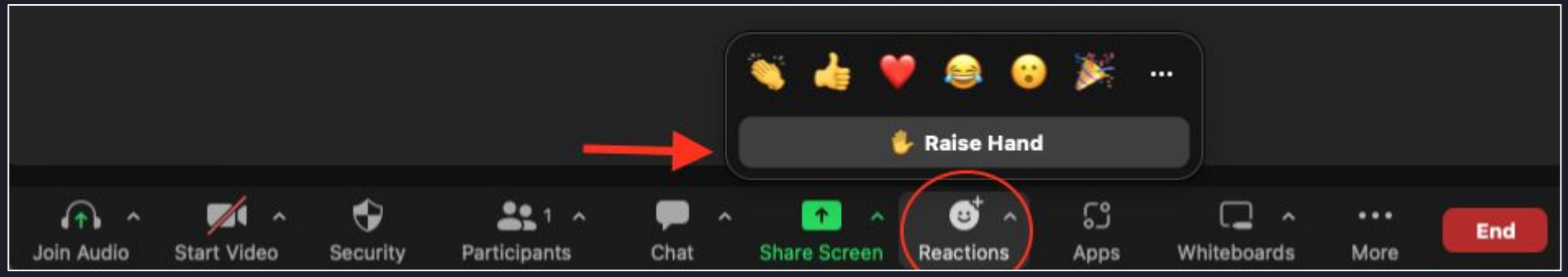
Question 3:

What other ways can we ensure we “did a good job” with our threat models?

Question 4:

Are there other “objective properties” of threat models we should be aware of?

Use “✋ Raise Hand” feature to let the hosts know you have something to share.



Next Meetup

Topic: Show Me the Money: Open FAIR and the ROI for Threat Modeling

Date: 11AM-noon ET, Thursday, Feb 22, 2024

Register: threatmodelingconnect.com/events