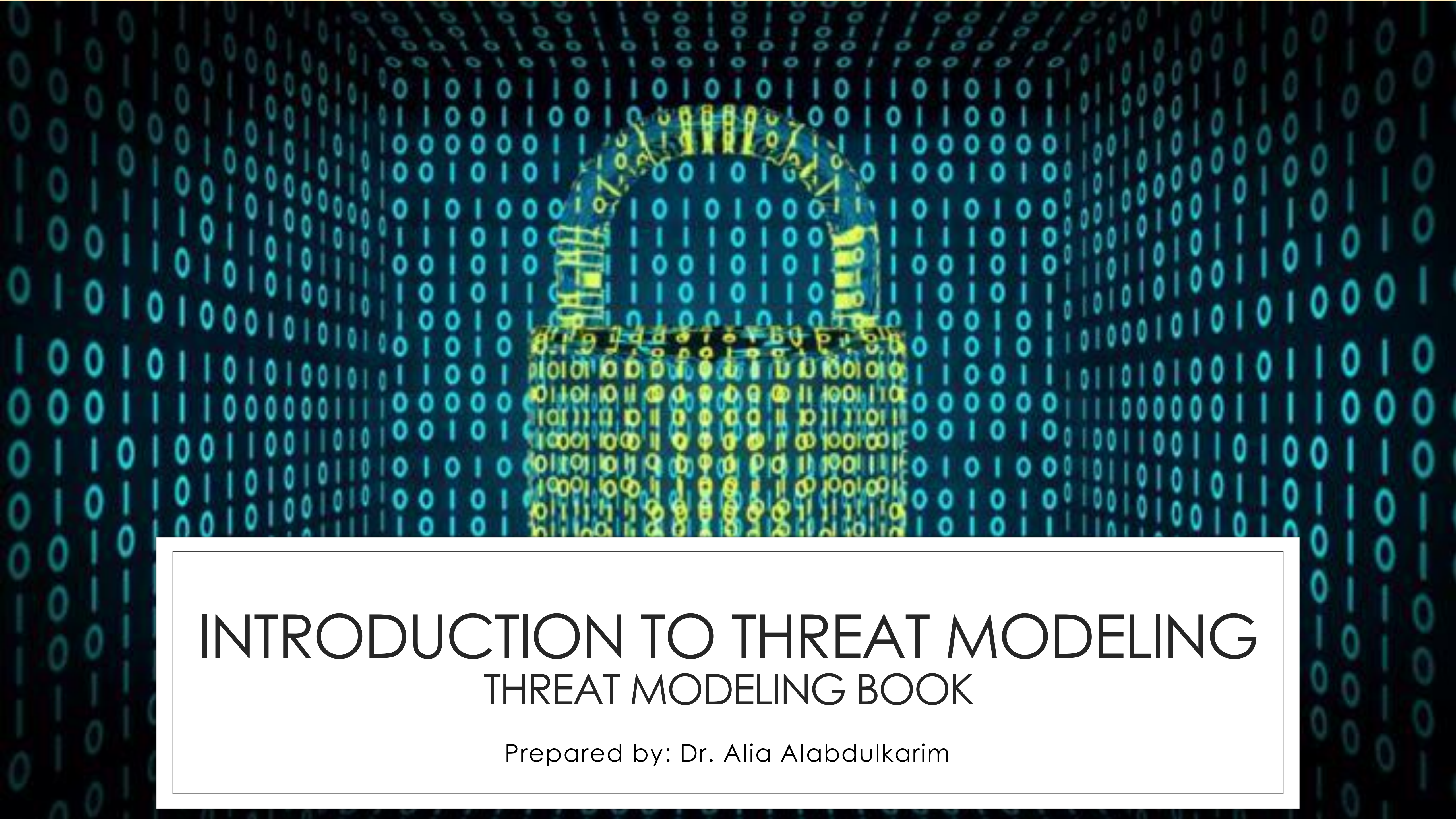


INTRODUCTION TO THREAT MODELING

THREAT MODELING BOOK

Prepared by: Dr. Alia Alabdulkarim

Threat Modeling

- Anyone can learn to threat model, and what’s more, everyone should.
- Threat modeling is about using models to **find security problems.**
- Using a model means abstracting away a lot of details to provide a look at a bigger picture, rather than the code itself.
- You model because:
  - It enables you to **find issues in things you haven’t built yet**
  - It enables you to **catch a problem before it starts**
  - It is a way to **anticipate the threats that could affect you**

Learning to Threat Model

4 key questions:

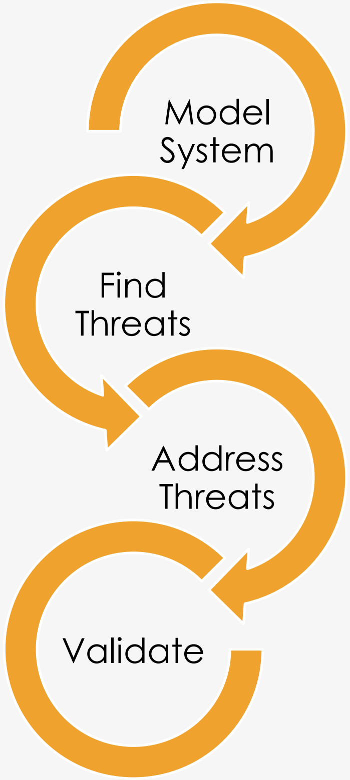
Q1. What are you building?

Q2. What can go wrong?

Q3. What should you do about those things that can go wrong?

Q4. Did you do a decent job of analysis?

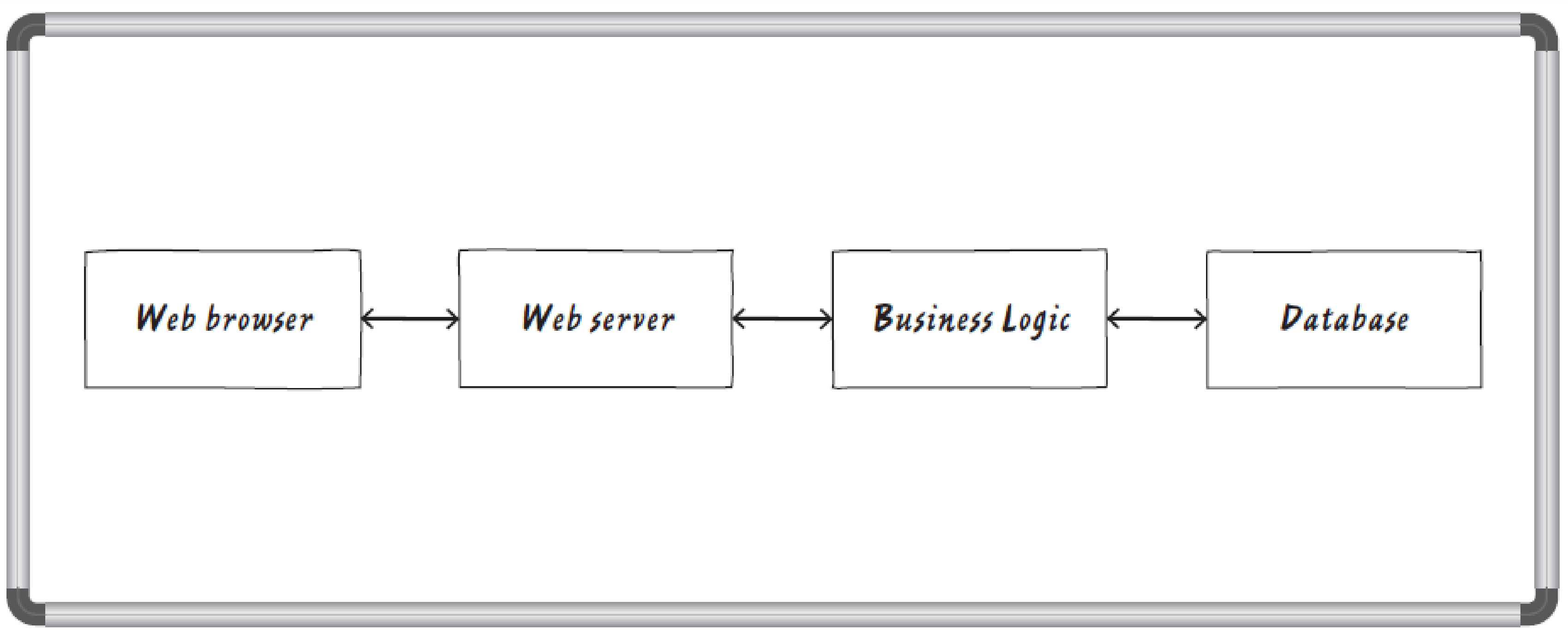
Those questions lead to 4 major activities involved in threat modeling:



Q1. What Are You Building?

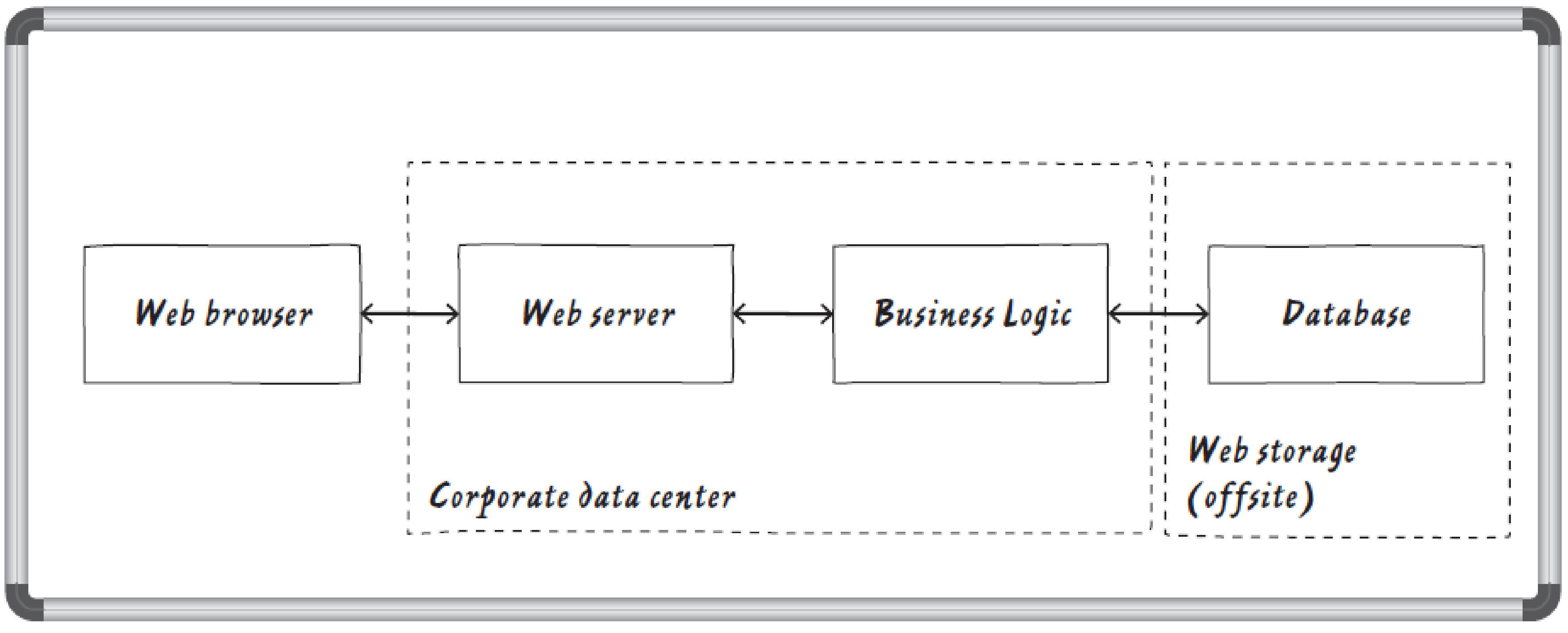
🔒Web application? Mobile Application? Desktop Application… etc.

🔒The type of your application mandates the remaining activities of threat modeling



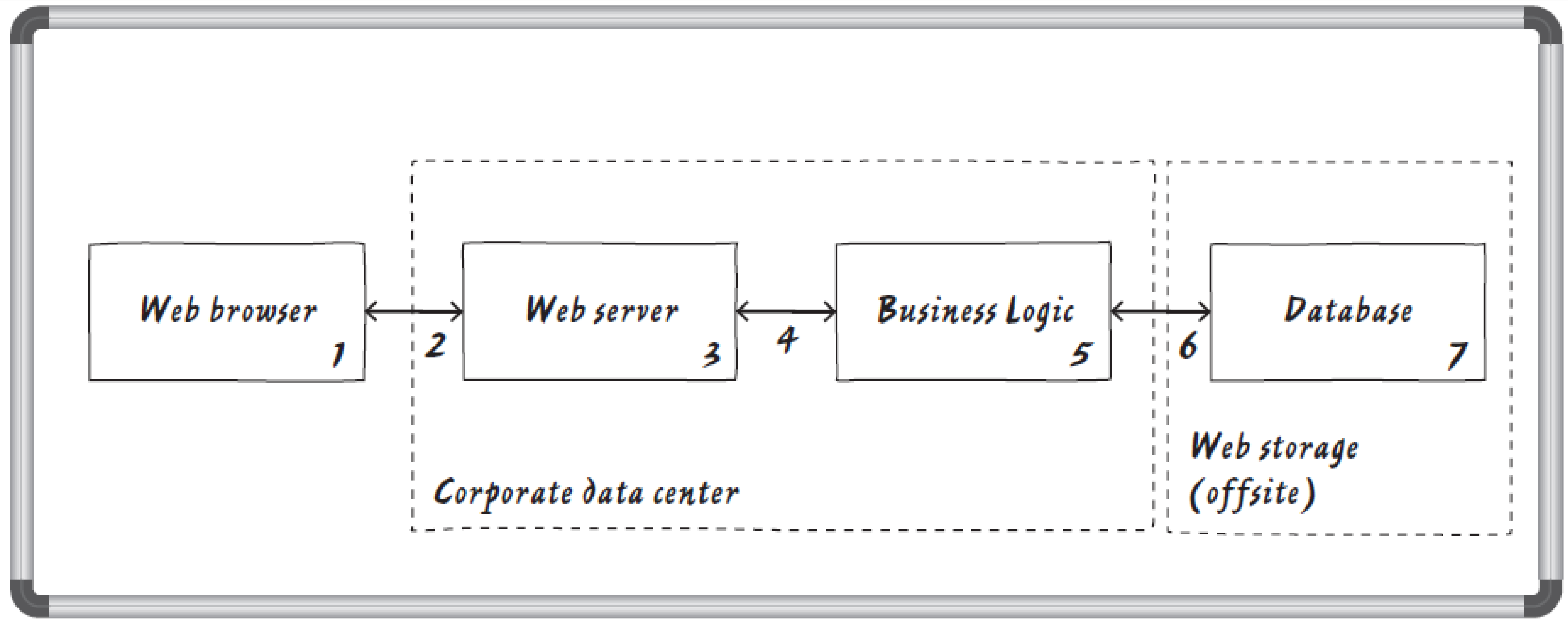
Q1. What Are You Building?

🔒Adding boundaries

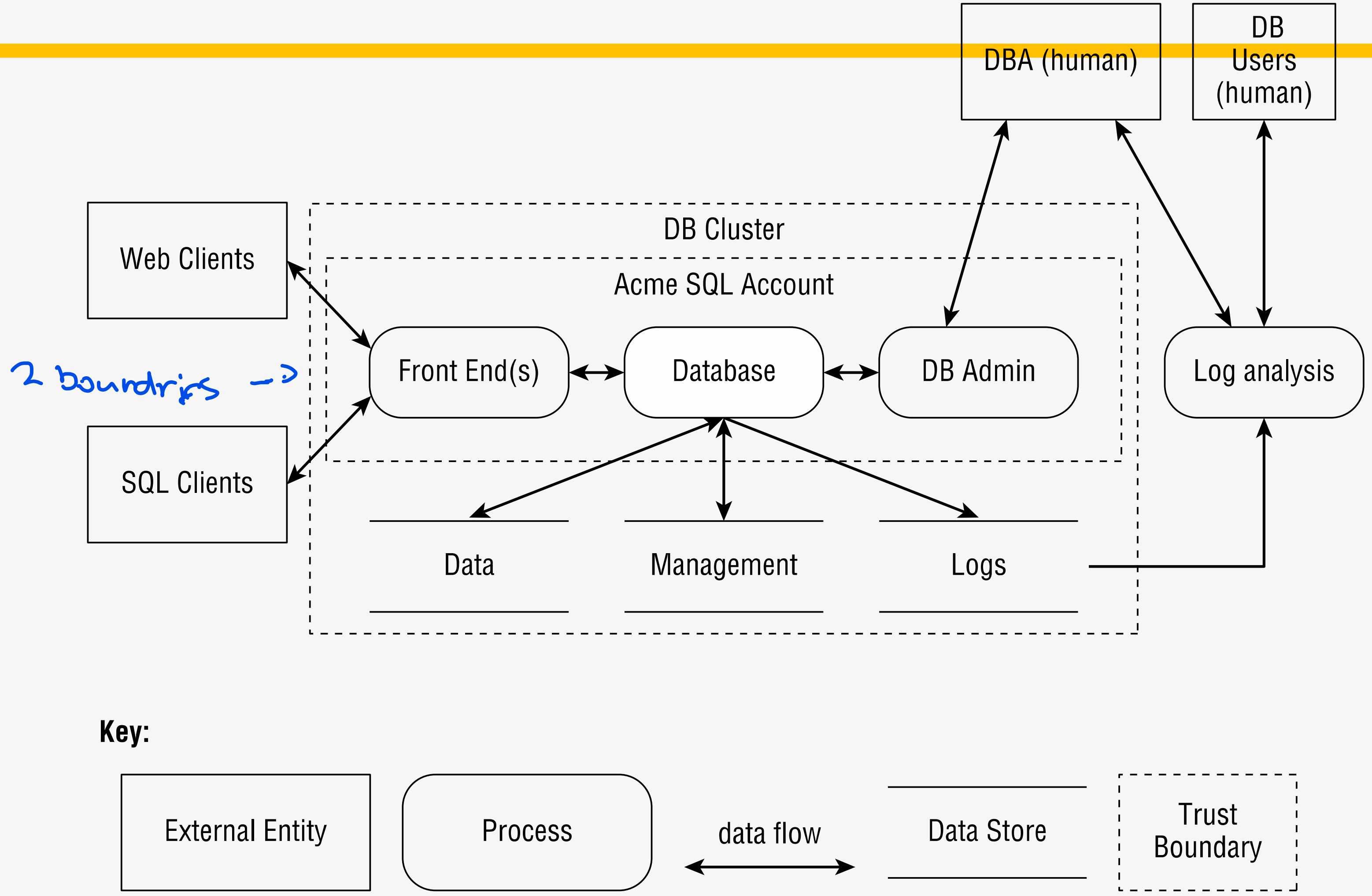


Q1. What Are You Building?

🔒Adding numbers to the diagram



Q1. What Are You Building?



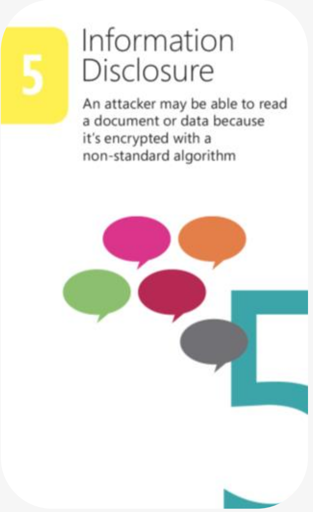
Q2. What Can Go Wrong?

- 🔒Start looking for what can wrong using the diagram
- 🔒Think about what could go wrong:
  - 🔑How do you know that the web browser is being used by the person you expect?
  - 🔑What happens if someone modifies data in the database?
  - 🔑Is it OK for information to move from one box to the next without being encrypted?
- 🔒Example of methods that can be used to find threats:
  - 🔑Elevation of Privilege (EoP) game
  - 🔑STRIDE (**S**poofing, **T**ampering, **R**epudiation, **I**nformation disclosure, **D**enial of service, **E**levation of privilege) – **Chapter 3**
  - 🔑Attack trees – **Chapter 4**

Q2. What Can Go Wrong?

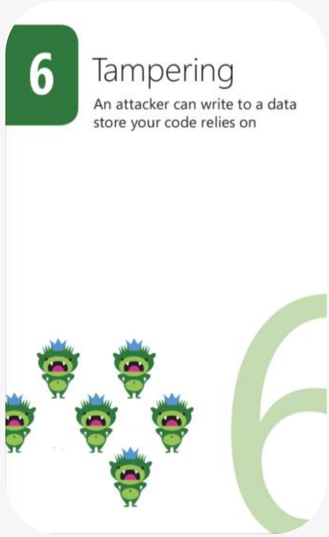
Elevation of Privilege (EoP) game

- 🔒Elevation of Privilege is a serious card game designed to **help you identify threats**.
- 🔒Each card has a number in the upper left, and an example of a threat as the main text on the card.
- 🔒Each round works like so:
  - Each player plays one card, starting with the person leading the round, and then moving clockwise.
  - To play a card, read it aloud, and try to determine if it affects the system you have diagrammed.
  - When each player has played a card, the player who has played the highest card wins the round. That player leads the next round.
  - When all the cards have been played, the game ends and the person with the most points wins.



5 Information Disclosure

An attacker may be able to read a document or data because it's encrypted with a non-standard algorithm



6 Tampering

An attacker can write to a data store your code relies on

Q2. What Can Go Wrong?

Tips for Identifying Threats

- **Start with external entities:** نبدأ من برا
  - Always maintain a structure or an organization..
- **Never ignore a threat because it’s not what you’re looking for right now:**
  - You might come up with some threats while looking at other categories
  - Write them down and come back to them
- **Focus on feasible threats:**
  - “Someone might insert a back door at the chip factory,”
  - “Someone might hire our janitorial staff to plug in a hardware key logger and steal all our passwords.” عمال نظافة
  - Real possibilities but not very likely compared to other more common attacks

Q3. What should you do about those things that can go wrong?

Addressing Each Threat

- The next step is to go through the lists and address each threat

- **Four possible actions:**

  تخفيف

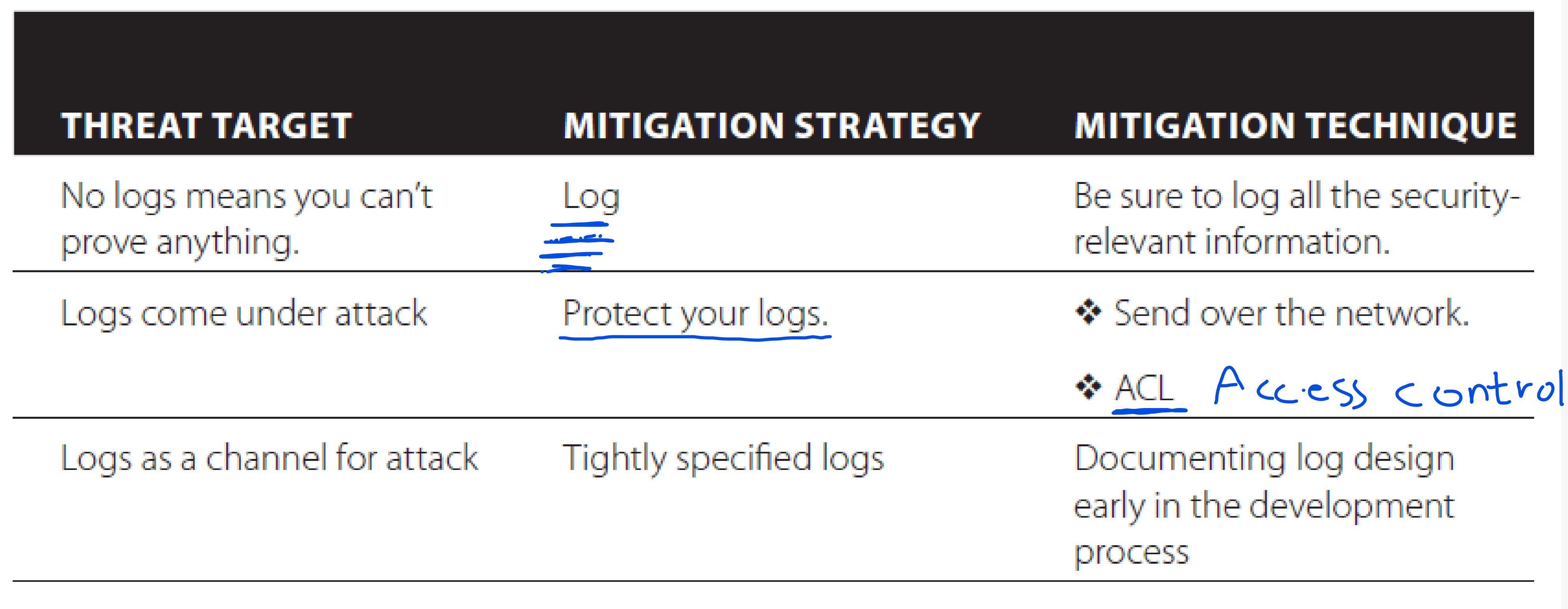
  معالجه سريعه تخفف

  - **Mitigating threats:** is about doing things to make it harder to take advantage of a threat.
  - **Eliminating threats:** is almost always achieved by eliminating features. => إزالتها تماماً
  - **Transferring threats**: is about letting someone or something else handle the risk. نخلي شخص => يتحمل المسؤوليه
  - **Accepting the risk:** when an unlikely threat requires an expensive solution.

- Mitigation is generally the easiest and the best for your customers

Q3. What should you do about those things that can go wrong?

Example: Addressing Repudiation ← إنكار



THREAT TARGET	MITIGATION STRATEGY	MITIGATION TECHNIQUE
No logs means you can't prove anything.	Log	Be sure to log all the security-relevant information.
Logs come under attack	Protect your logs.	❖ Send over the network. ❖ ACL Access control list
Logs as a channel for attack	Tightly specified logs	Documenting log design early in the development process

موأهِل در
يفحص

Q4. Did you do a decent job of analysis?

Checking Your Work

- 🔒Validation is the last thing you do
- 🔒Consists of few tasks:
  - Checking the model ✓ ⇒ التأكد من التغير الأخيرة مطابقة - للمودل أبلكيشن OR is it comblete?
    - ➢Updating the diagram
    - ➢Diagram details
  - Checking each threat ✓ ⇒
  - Checking your tests ✓

Q4. Did you do a decent job of analysis?

Checking the Model

- Final model must match what you built
  - Otherwise how do you know that you found the right threats
- Arrange for a meeting to answer the questions:
  - Is this complete??
  - Is it accurate??
  - Does it cover all the security decisions we made??
  - Can I start the next version with this diagram without any changes?
- If all answers are "yes" → sufficient I'm done
- At least one no → you need to update ✓

Q4. Did you do a decent job of analysis?

Checking Each Threat

- 🔒Two ways:
  - 🔑Checking you correctly addressed each threat you found ✓
    - ➢Did you do something with each threat?
    - ➢You don't want to drop anything
    - ➢Take time in taking meeting minutes to document all bugs
  - 🔑Asking if you found all the threats you should find✓

Q4. Did you do a decent job of analysis?

Checking Your Tests

- Ensure you have built a good test to detect the problem
    - Manual
    - Automated
- Some will be easy, other will be tricky

Case Study (Appendix E)

Acme’s Operational Network (Reading assignment)

The systems that make up the operational network are as follows:

- **Desktop and mobile:** are the end-user systems that everyone in the company uses.
- **E-mail and intranet:** are an Exchange server and a set of internal wikis and blog servers.
- **Development servers:** includes the local source-control repository, along with bug tracking, build, and test servers.
- **Production:** This is where products are made using a just-in-time approach. It includes an operations network that is full of machine tools and other equipment that is finicky and hard to keep operational, never mind secure.

Case Study (Appendix E)

Acme’s Operational Network (Reading assignment)

The systems that make up the operational network are as follows:

- 🔒**Directory:** This is an Active Directory server, which is used for account management across most of the systems at Acme.
- 🔒**HR Management:** This is a personnel database, time-card system for hourly employees, and related services.
- 🔒**Website/Sales/CRM:** This is the website through which orders are placed. The website runs at an IaaS cloud provider. It has a direct connection to the production shop. The website is locally built and managed with a variety of dependencies.
- 🔒**Payroll:** This is an outsourced payroll company.

Case Study (Appendix E)

Acme's Operational Network (Reading assignment)

Q1. What are you building?

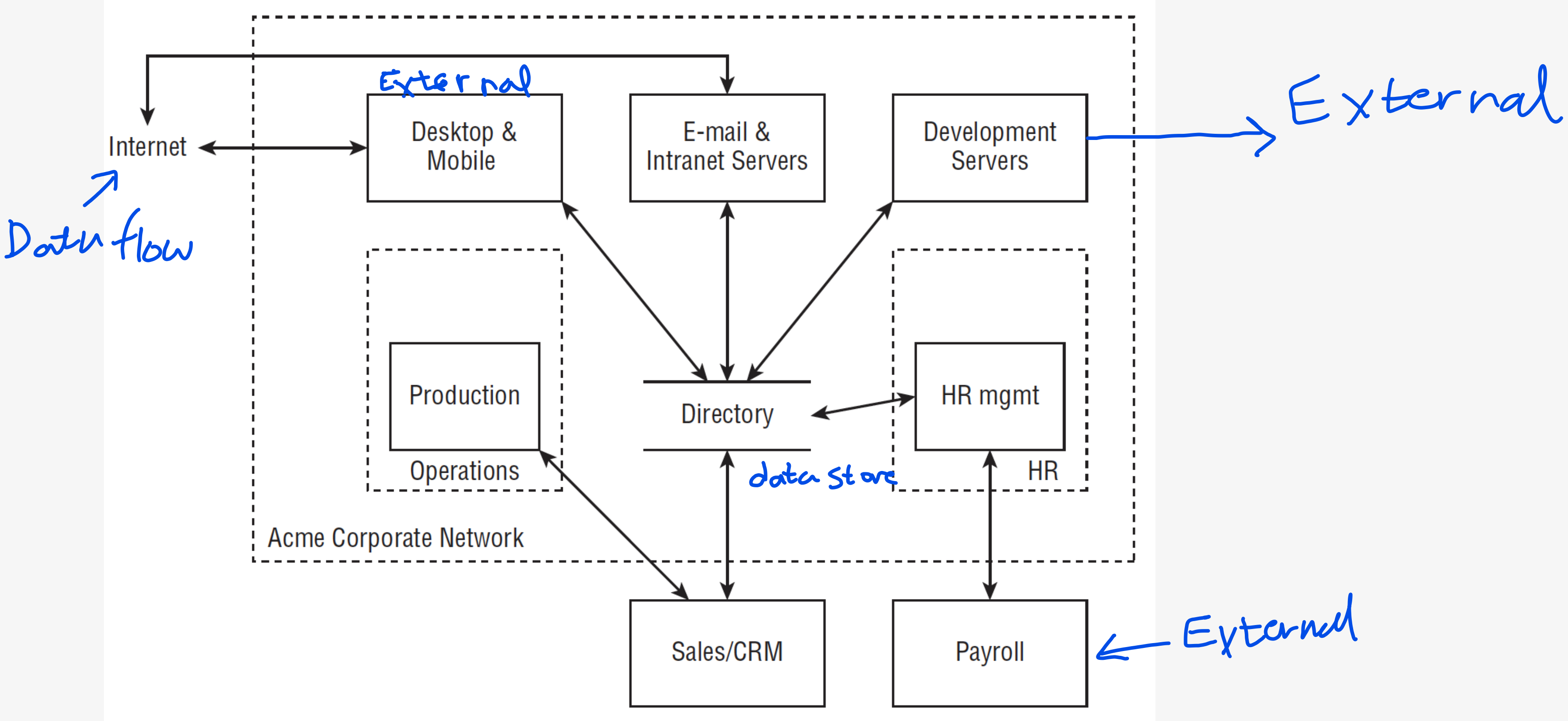


Figure E-2: Acme's operational business network

Case Study (Appendix E)

Acme's Operational Network (Reading assignment)

Q1. What are you building?

Q2. What can go wrong? (use STRIDE– Chapter 3)

Q3. What should you do about those things that can go wrong? (use STRIDE– Chapter 3)

Q4. Did you do a decent job of analysis?

In summary, Acme has used STRIDE threat modeling and a model of their operational network to identify many threats. Again, they have moved from a vague sense of unease to a well justified set of concerns, which they can work through. From here, they'd need to decide on a prioritization scheme for those concerns, or consider additional security requirements, depending on their unique needs.

References

- Threat Modeling
  - Chapter 1: Dive In and Threat Model
  - Appendix E: Case Studies
- Extra references