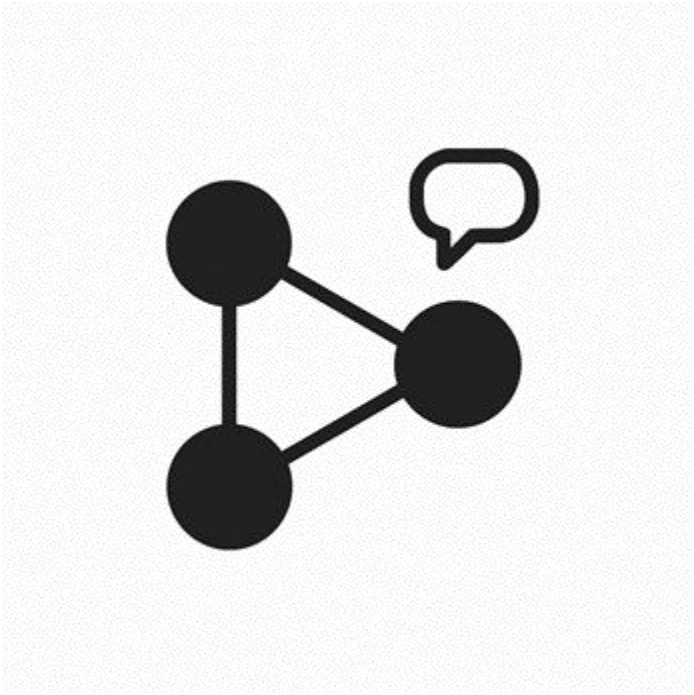
Diagrams that communicate

Using deployment diagrams to explain IT architecture and security to everybody

With Dr. Peter van Eijk



ClubCloudComputing

Your job

- IT architect
- Solutions architect
- Risk analyst
- Security architect
- Assessor?

Your position of power

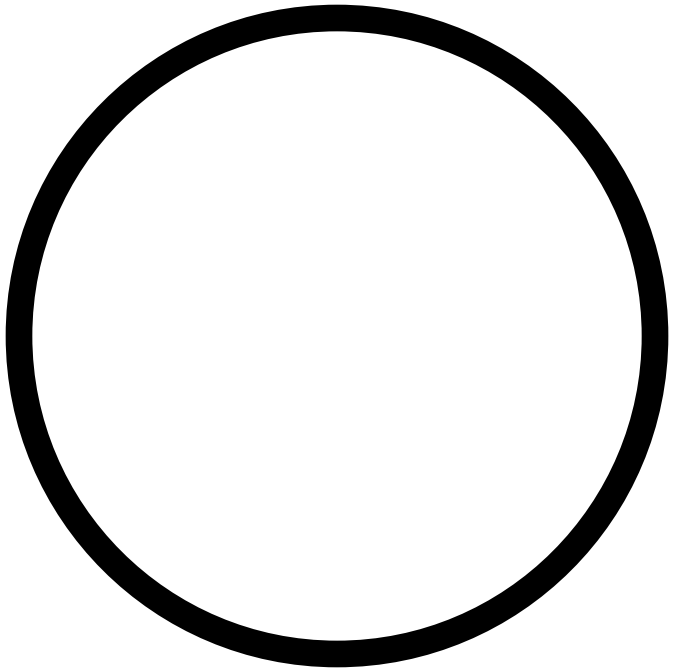
- You need to understand what other people are doing
- They need to understand what you want them to do
- Everybody needs to understand the risks and requirements
- Diagrams are a power tool

More: https://digitalinfrastructures.nl/book/power/positions-of-power-it/

Deployment diagrams

- Show allocation of IT function
- Including responsibilities for build, run, update, security, …

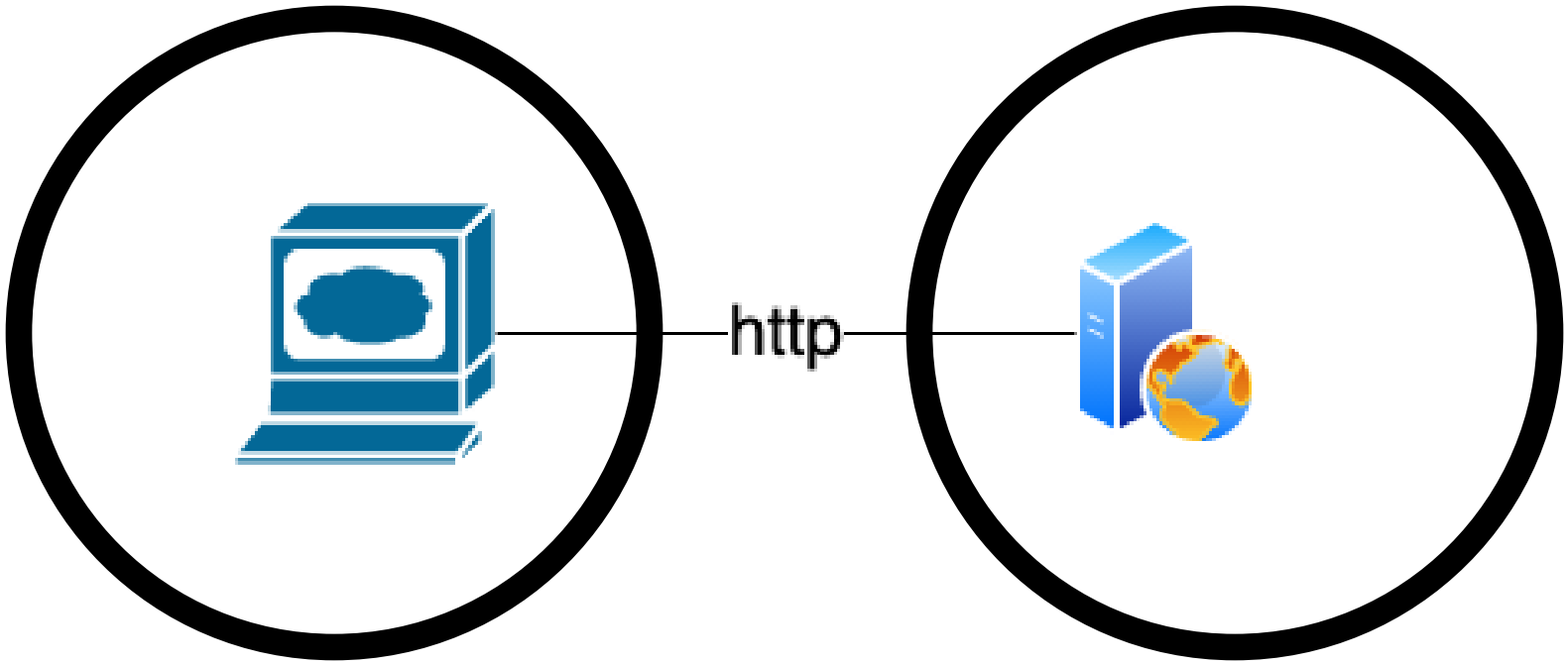
The basic symbols



Enclosure

Connection

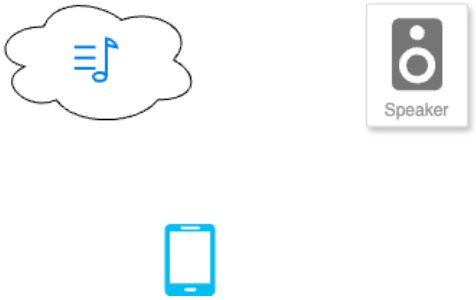
Webbrowsing



The circle represents: a domain of control, a control boundary, a demarcation of responsibilities.
If you control the boundary, you control the software and data that is inside.
Inside the boundary there is often an ***execution environment.***

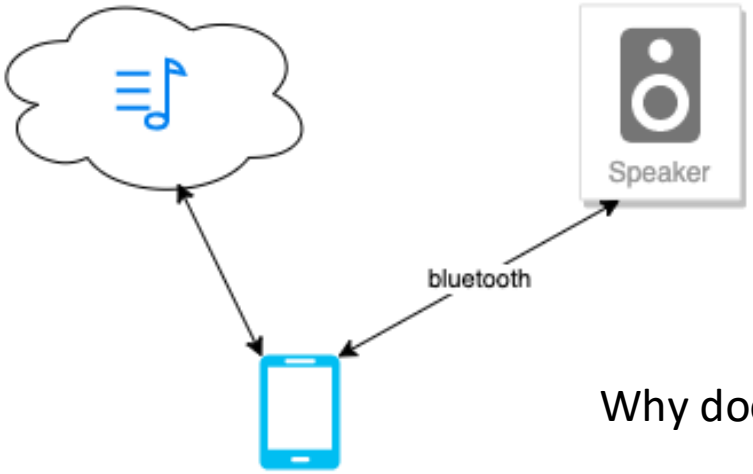
THESE DIAGRAMS LOOK VERY INTUITIVE, BUT YOU WANT TO CHECK THE DETAILS

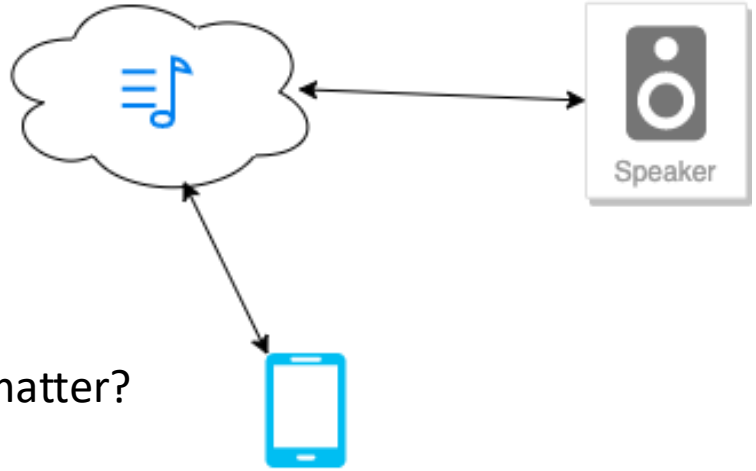
Home example



Your phone controls the music from the cloud (e.g. Spotify)

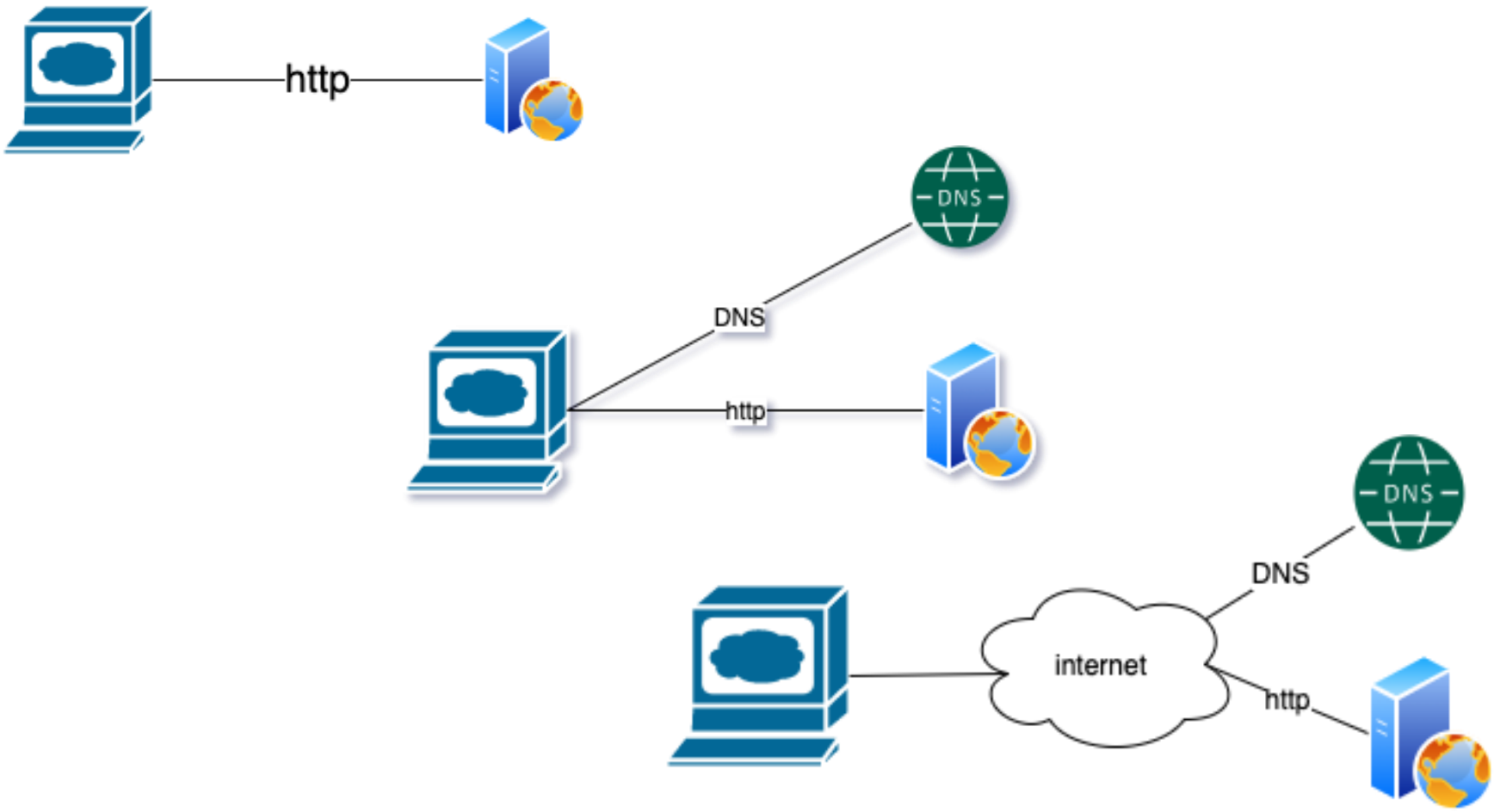
But how does the music flow?



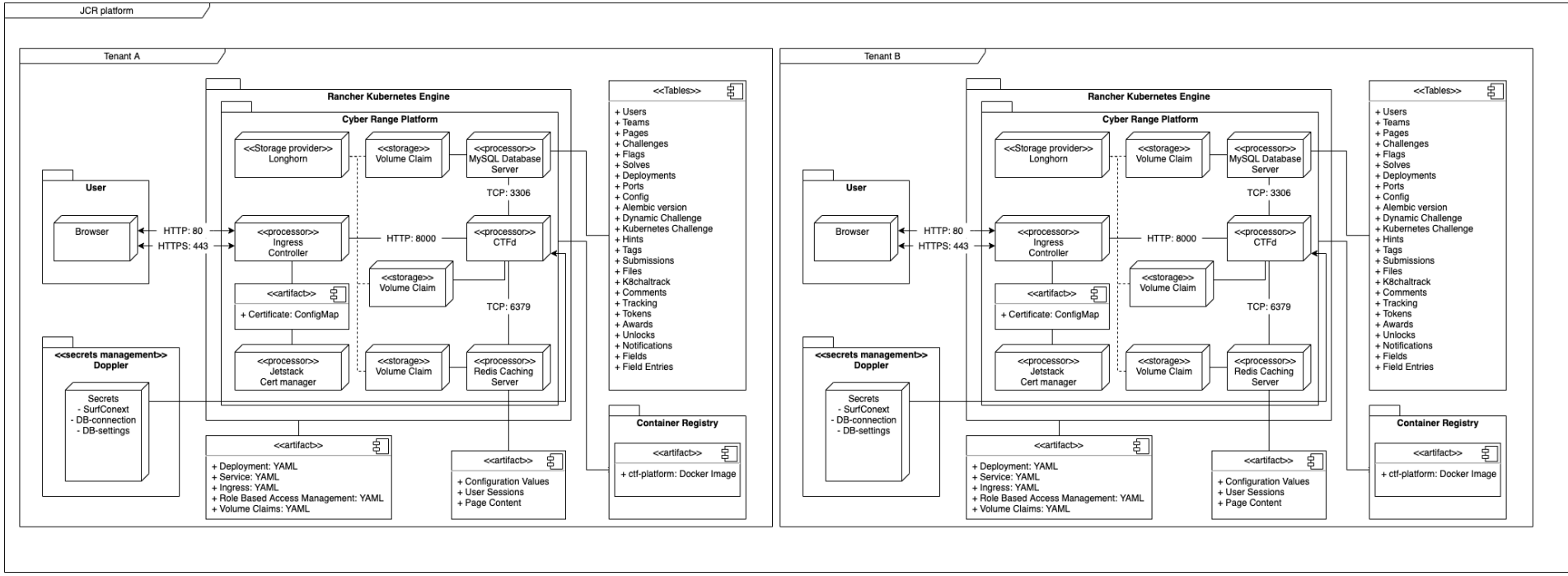


Why does it matter?

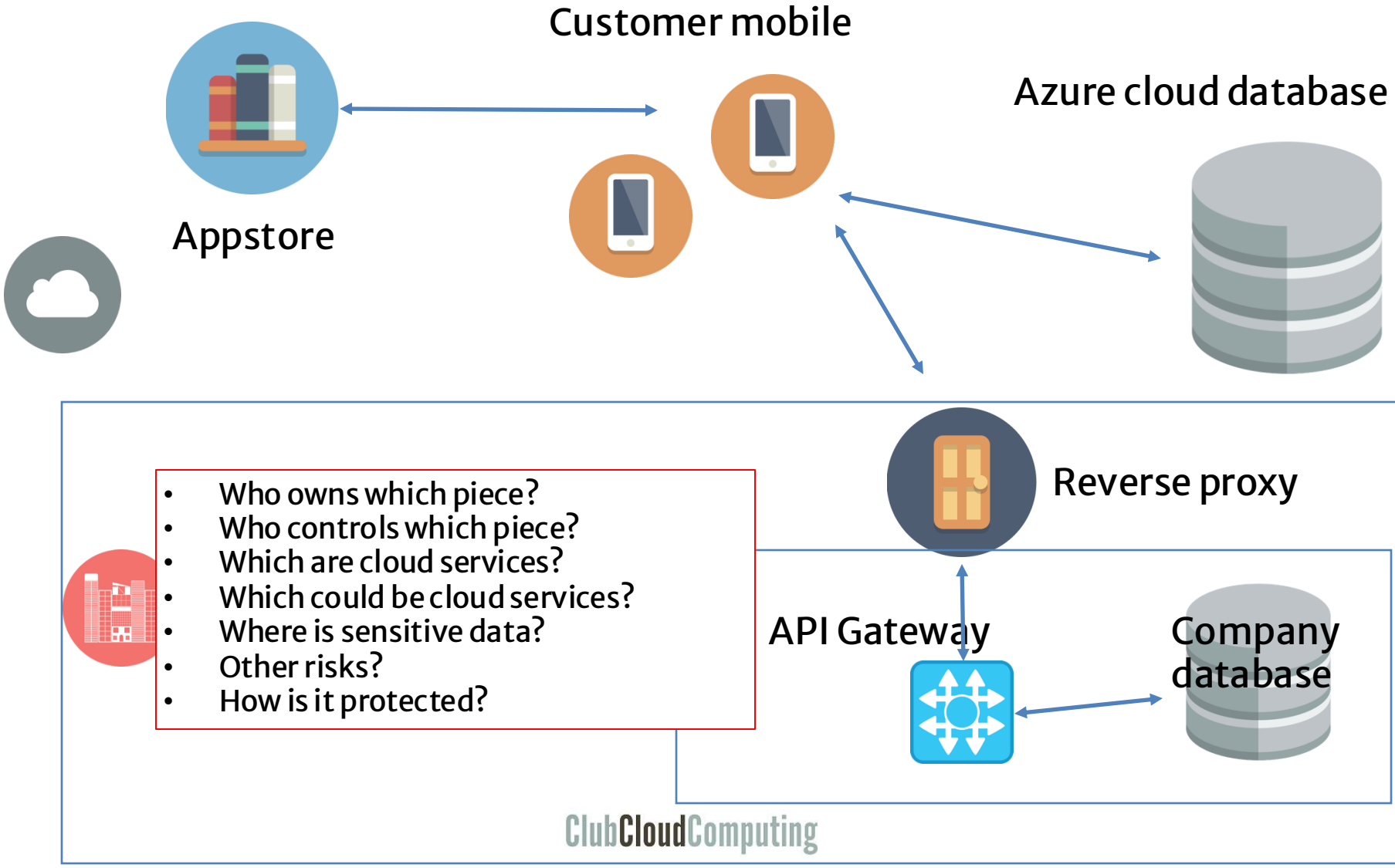
You can refine as you wish



It can get too refined ...



Hybrid cloud example



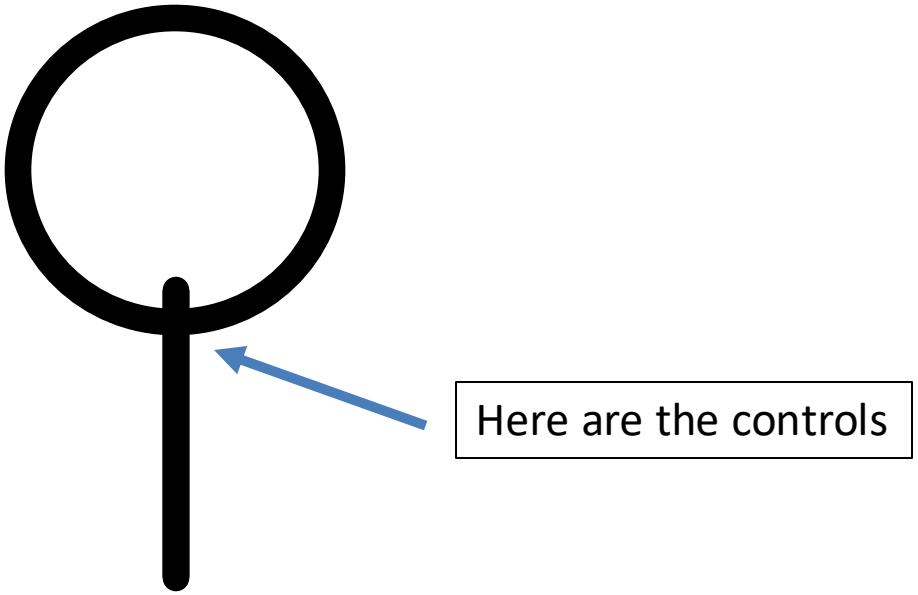
Inside the control boundary

- Data
  - Including configuration and secrets
- Software
- AI models
- …

This is a starting point for analysis

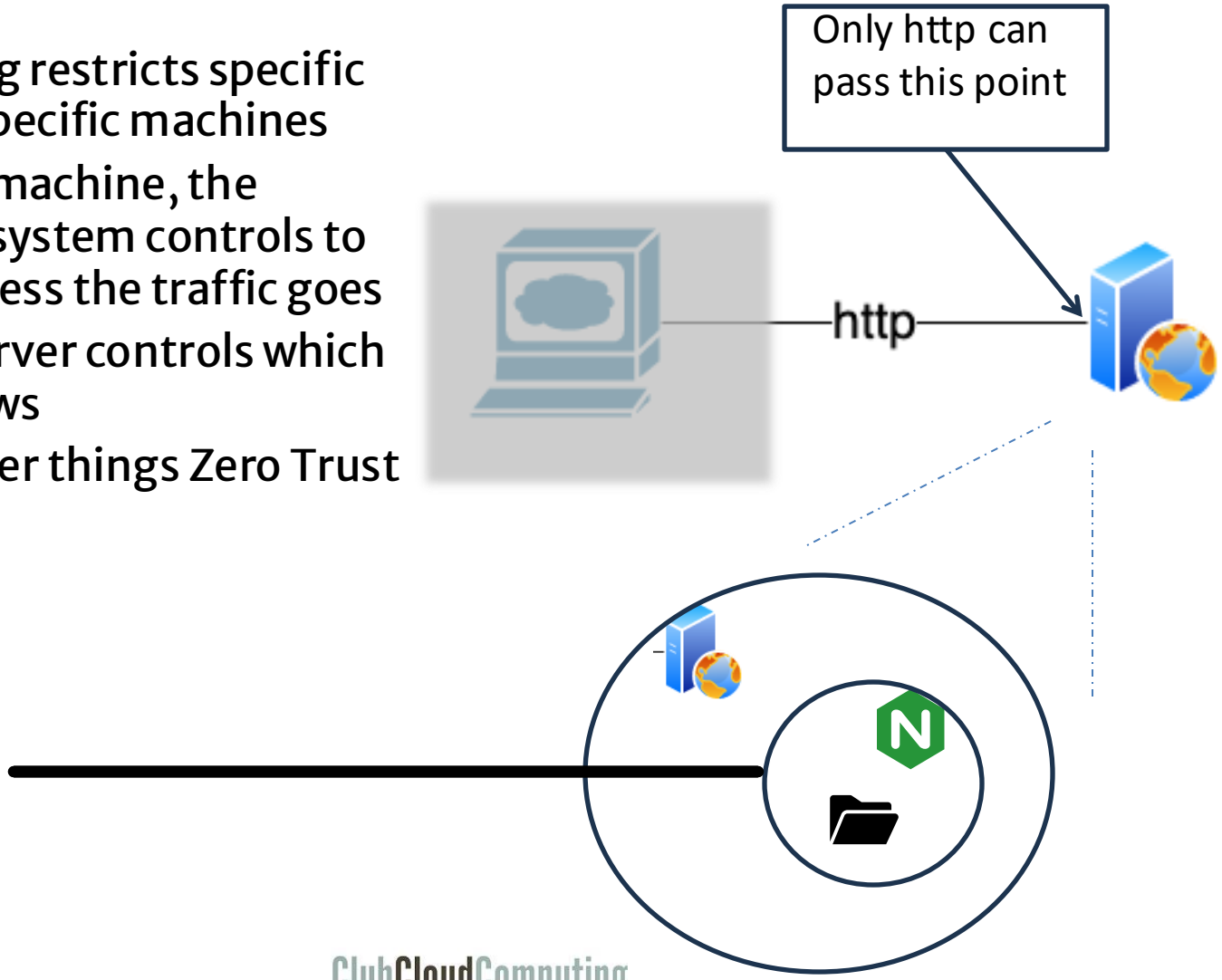
The boundary controls

- “The box”
- Control boundary
- Perimeter
- Policy
- Network segment
- …



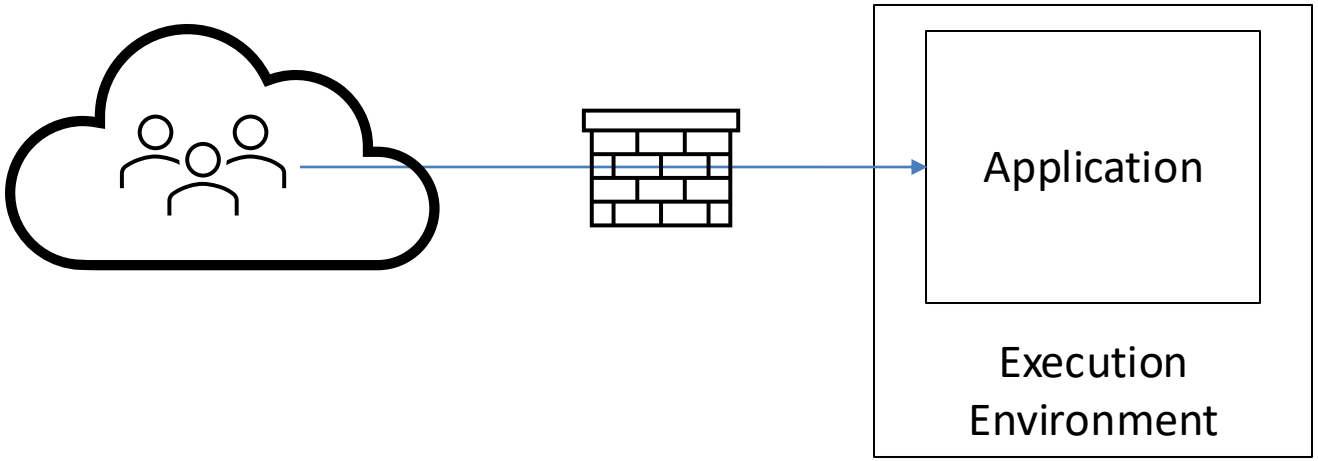
Example controls

- Networking restricts specific traffic to specific machines
- Inside the machine, the operating system controls to which process the traffic goes
- The webserver controls which files it shows
- And all other things Zero Trust



Retrofitting Zero Trust

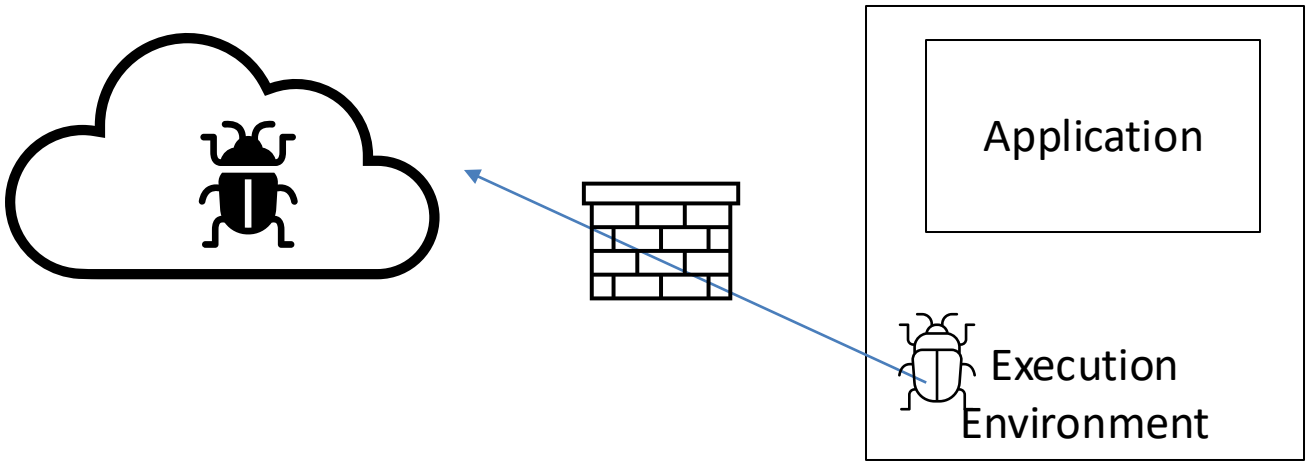
Note, each control boundary can be a PEP (Policy Enforcement Point)



Additional controls - exfil

- Who? 4-eyes for large downloads?
- When? Time of day
- Where? Source IP filtering?
- Why? Only allow specific users

Note, each control boundary can be a PEP (Policy Enforcement Point)



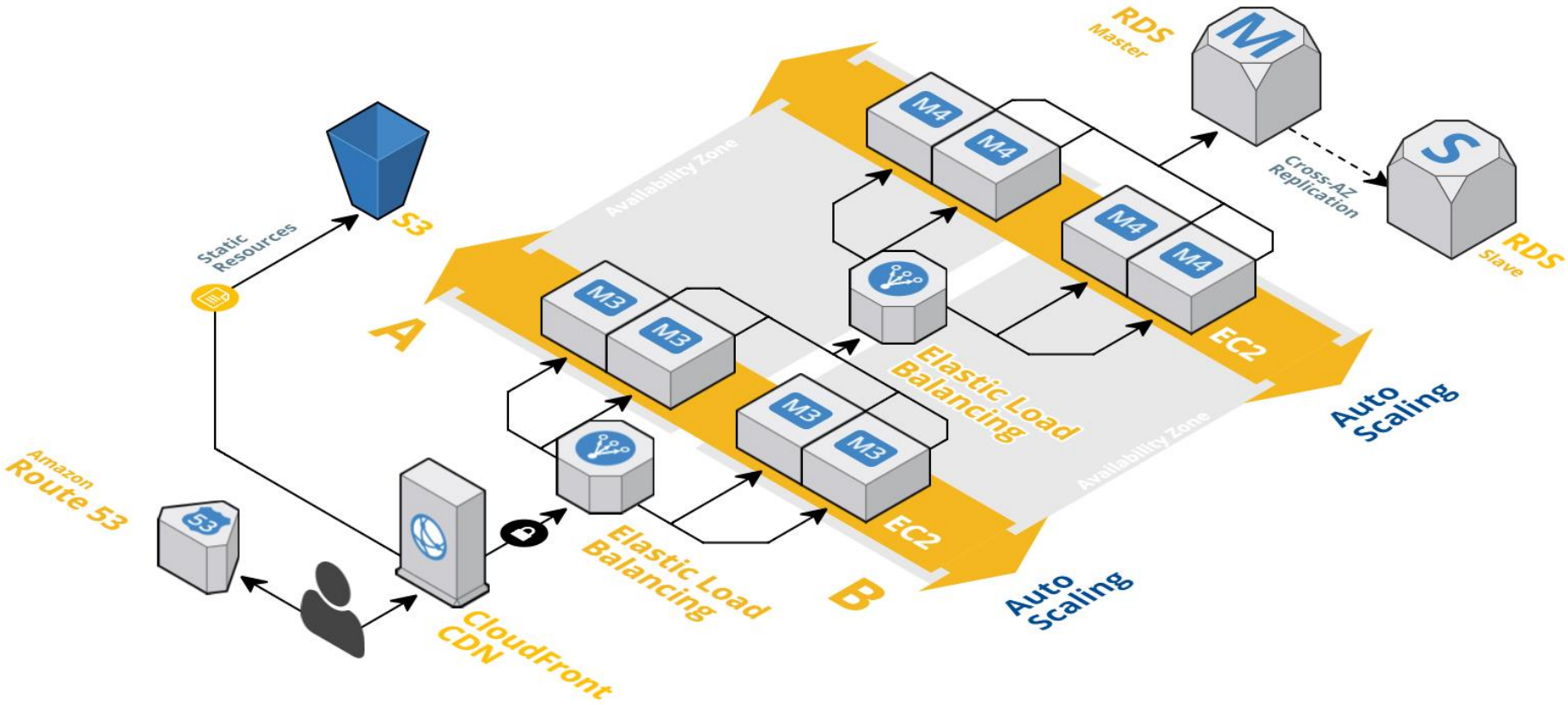
Additional controls – reverse allow

- Who? To which server?
- When? Note: logging and monitoring happens all the time
- Where? In the firewall
- Why? Exfil of sensitive data

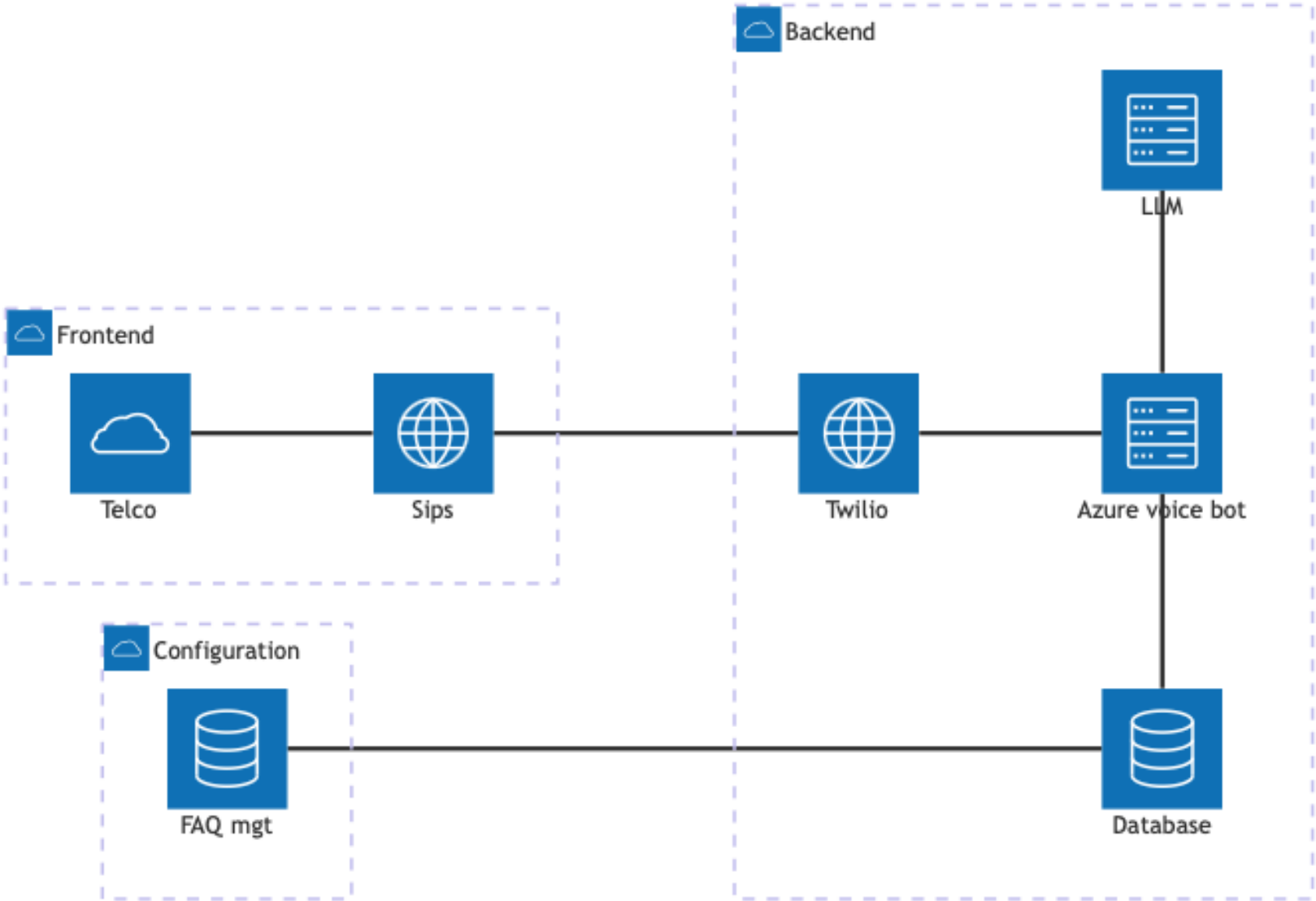
Examples of Execution Environments

- Containers, functions as a service such as AWS Lambda, AWS S3 storage buckets, SaaS/PaaS providers, Antivirus agents, firewalls, routers, switches, and so on
- There is data inside, and code
- Tip: ask how the *code* gets deployed inside (it is an attack vector)

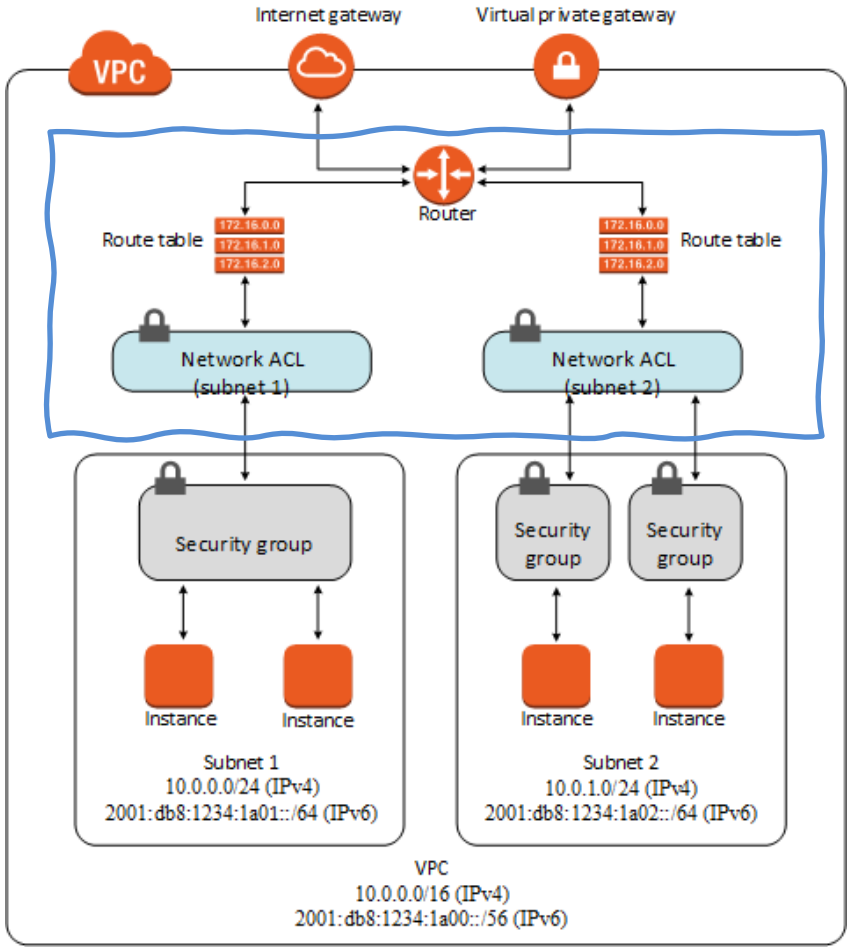
Example AWS design drawn by cloudcraft.co



AI voice bot example

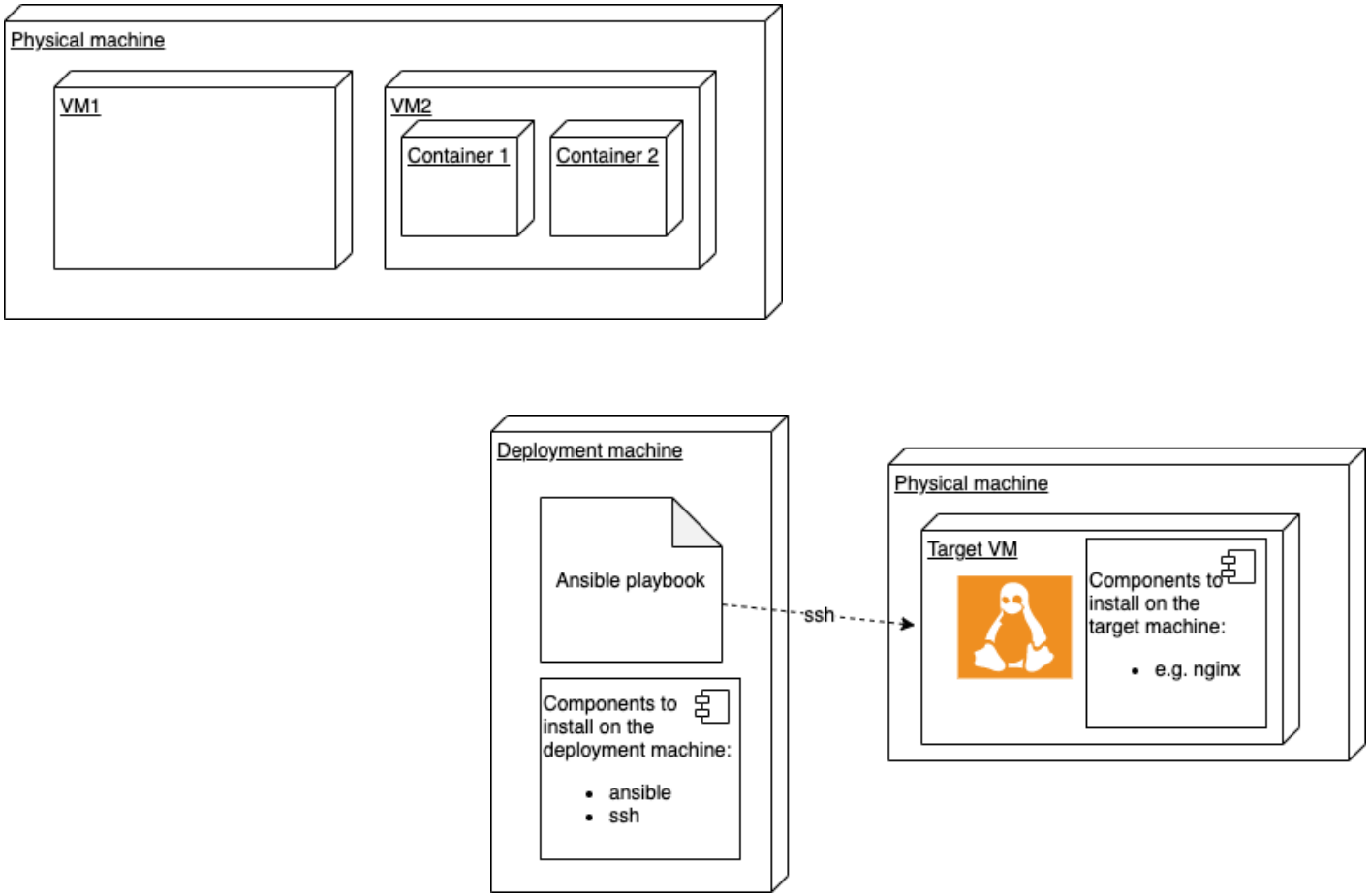


What is wrong here?

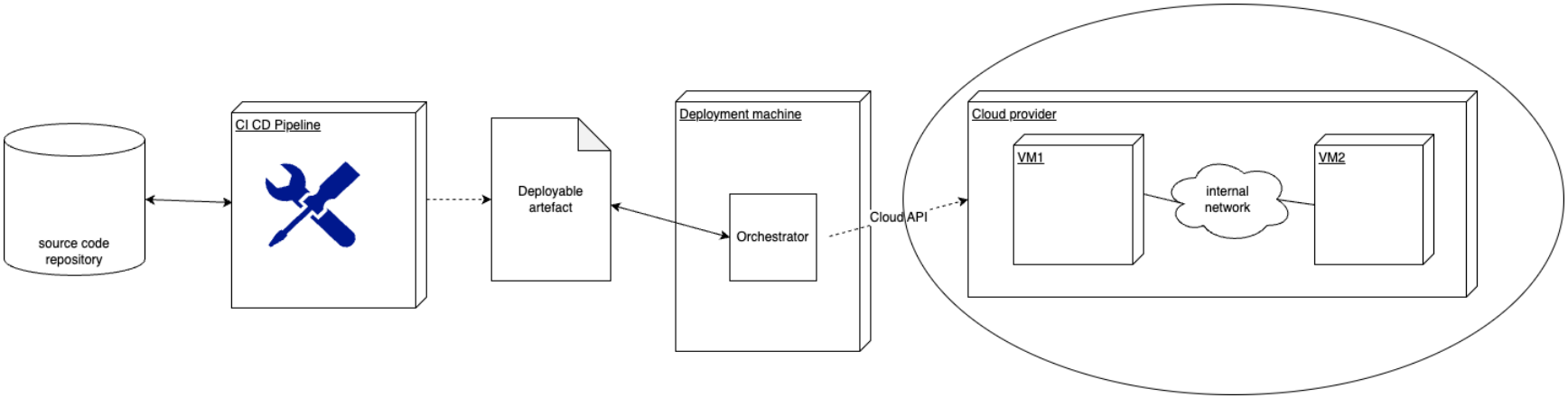


- Various configuration items are really inside the router environment
- A similar thing holds for the security groups (it is not an execution environment). In fact it is more approriate to call the subnet the security group.

Virtualization



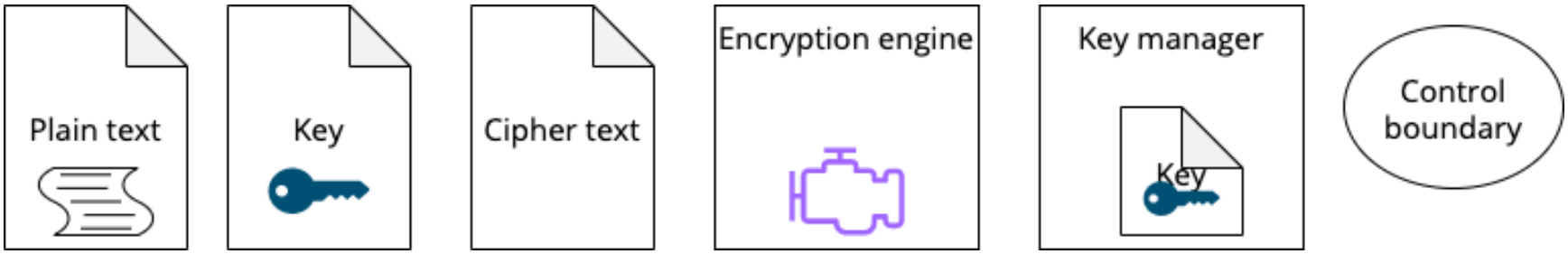
Continuous delivery and DevOps

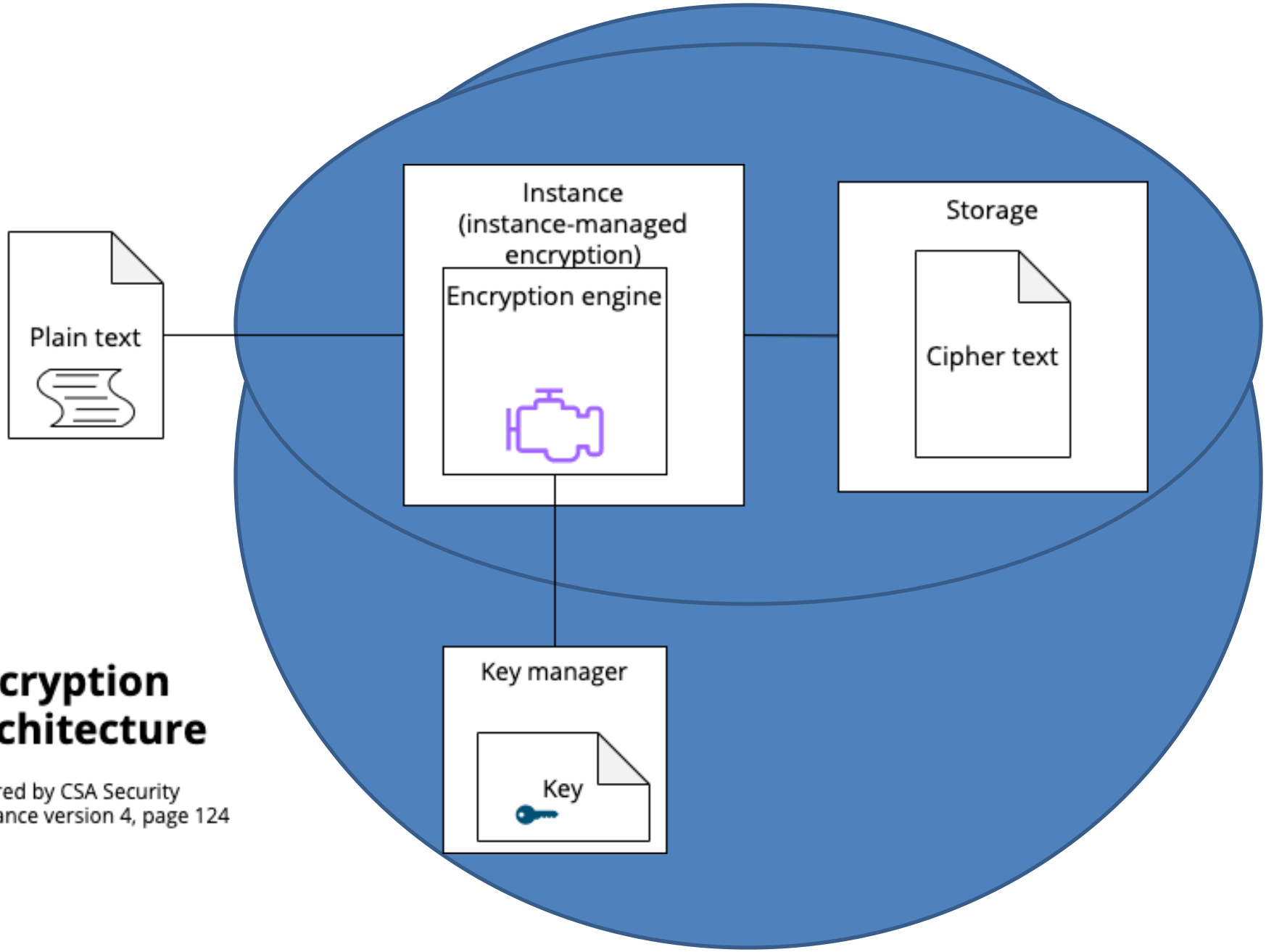


Cloud Encryption Architectures

Described using deployment diagrams

Deployment diagram symbols used here





Encryption Architecture

Inspired by CSA Security Guidance version 4, page 124

Drawing tools

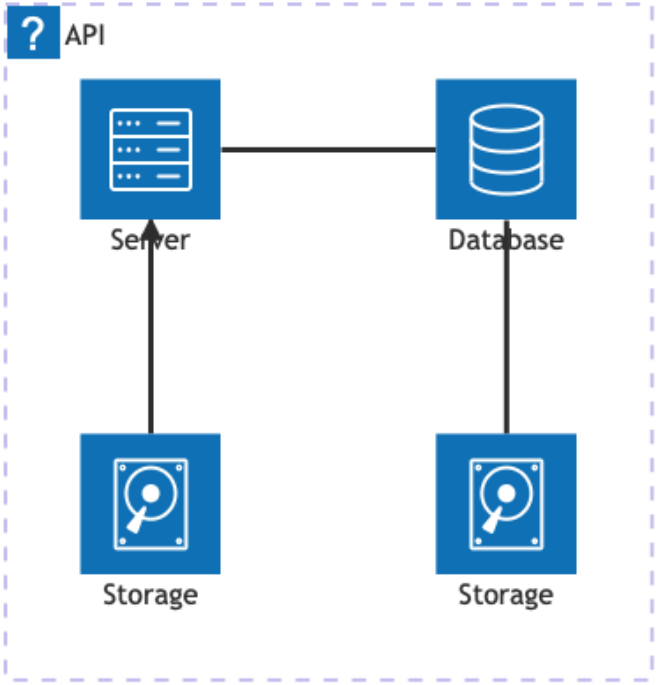
- Sharpie
- Powerpoint
- Mermaid
- Drawio
- D2
- dotuml for VSCode
- Claude: what type of mermaid diagram would be most useful for representing this image?
- And of course, your more professional architectural tools (UML, Archimate, etc).

Mermaid – scripted diagrams

```
architecture-beta
group api(Storage)[API]

service db(database)[Database] in api
service disk1(disk)[Storage] in api
service disk2(disk)[Storage] in api
service server(server)[Server] in api

db:L -- R:server
disk1:T --> B:server
disk2:T -- B:db
```

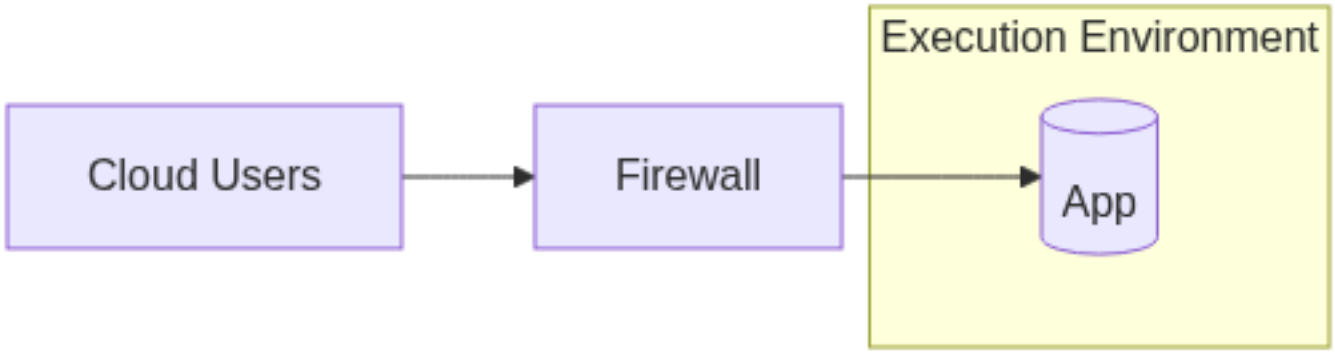


https://mermaid.live/

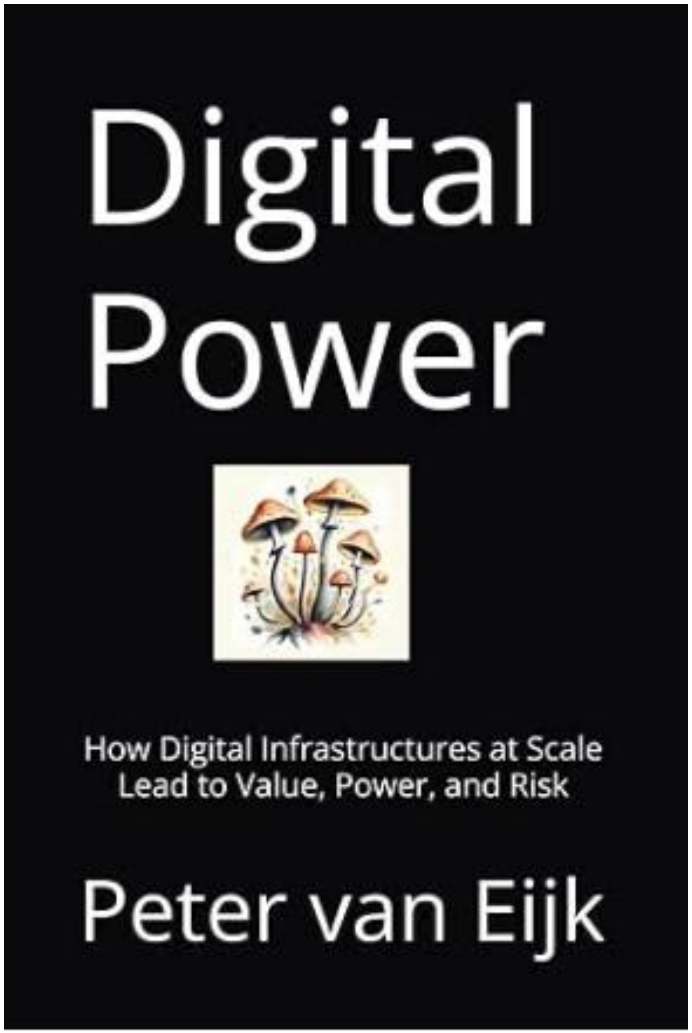
Mermaid

```
flowchart LR
  subgraph exec["Execution Environment"]
    app[App]
  end

  users[Cloud Users]
  fw[Firewall]
  users --> fw
  fw --> app
```



Shameless plug



PETER VAN EIJK
P@D1G.NL

YouTube channel: ClubCloudComputing

Thank you!

ClubCloudComputing

Feedback

- Thanks for attending!