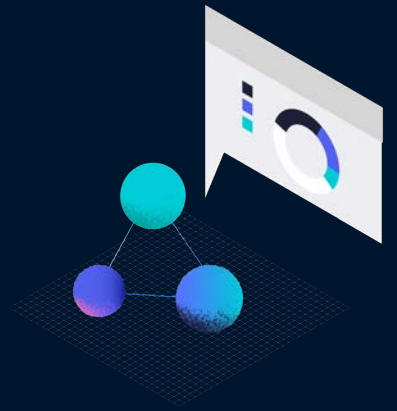
Cape Privacy

Protecting Sensitive Data and Machine Learning Models

Introduction



Introduction

- Confidential Computing
- Cape enables Confidential Computing
- Demos

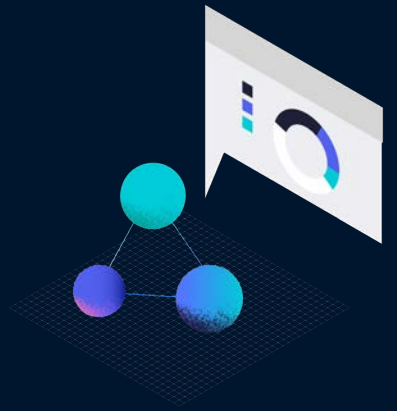
Who am I?!

- Working at Cape ~5 years
- Working on confidential computing
- Mostly backend developer with some frontend/sdk experience

Feedback

- You can pick up PyCape right now and try Cape out
- We’re looking for feedback on all aspects
- We’ll provide links to documentation, getting started guides and a link to join our discord so you can easily try Cape out and get help when needed!

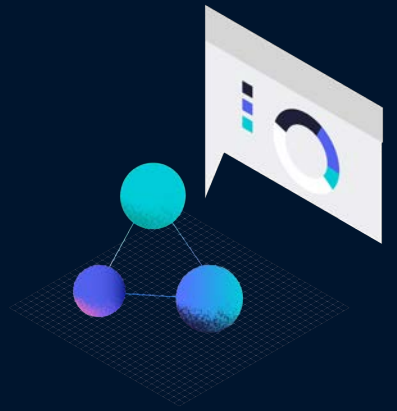
Confidential Computing



Confidential Computing

- What is confidential computing?
- Broad set of technologies (FHE, MPC, Enclaves) protecting data while in use
- Can complement each other
- Enclaves are what underlie Cape’s product

Primitives



Primitives

- Encryption & Key Management

- Common for developers to consider but still overly complicated

Decisions, Decisions

- AES vs. RSA
- If using the AES, the mode GCM, CBC
- Security, depending on what is chosen the security could be better or worse.
- Efficiency, depending on methods chosen one way could be more efficient than the other
- How to pack all the required data before sending it along. (i.e. tag, nonce)

What?

```
def _rsa_encrypt(inputs: bytes, public_key: rsa.RSAPublicKey) -> bytes:
    return public_key.encrypt(
        inputs,
        padding=padding.OAEP(
            mgf=padding.MGF1(algorithm=hashes.SHA256()),
            algorithm=hashes.SHA256(),
            label=None,
        ),
    )
```

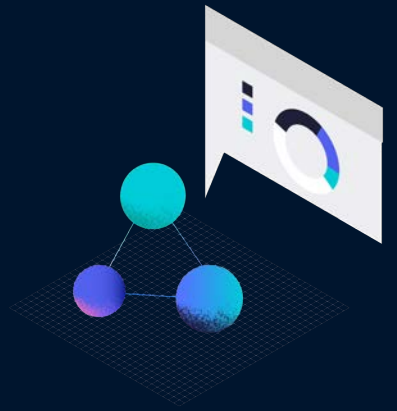
Library to help

- Can look up good defaults
- Would still take some time to fully understand
- Library which has the good default already decided would be ideal, library could then be a trusted source for documentation while also allowing you to configure the options if needed

Key Management

- Decisions to be made with key management as well
- Depending on the cloud provider could be many different products to choose from
- AWS Example: KMS & Secrets Manager
- Ideally would be able to simplify without all the choice

Nitro Enclaves



Nitro Enclaves

- Cape built upon AWS Nitro Enclaves
- Allows users to deploy code which can only be run in locked down container
- Attestations confirms that the code running in enclave is what you expect

Flexible

- Turn any container in an Enclave Image File
- EIFs are what are deployed to the encalve
- Contains the file system of the OS that was inside the container

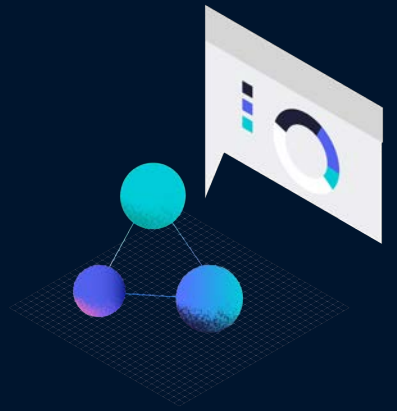
EIF Metadata

- Signature
- Platform Configuration Registers (PCRs)
- Creation time and similar information

Attestation

- Prove what the enclave is
- Enclave sends attestation document during communication
- Contains PCRs to prove what is running inside the enclave
- Signed by root AWS certificate which must be verified

Overview of Cape



Cape

- Working on confidential computing for over 4 years
- Helps protect their data and their user’s data
- We provide three main entry points into our system

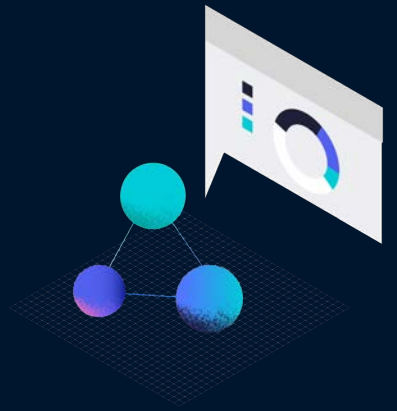
Cape's Verbs

- Encrypt
- Deploy, deploy python functions to Cape
- Run, run python functions with your encrypted data

SDKs

- Cape provides many SDKs for interacting with the platform
- Python, Javascript (browser, nodejs), Java
- CLI tool written in Golang
- More coming

PyCape & Cape Functions



PyCape

- Written in Python

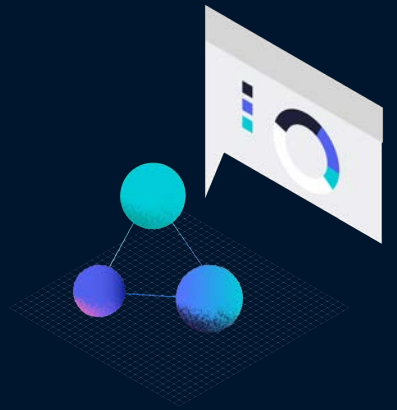
- Implements core functionality, encrypting, deploying and running functions

Cape Functions

- Written in python
- Utilities for packaging script and dependencies together
- Resulting directory is what is uploaded to Cape

```
def cape_handler(input_bytes: bytes):
    # do something
    return "output"
```

Its Demo Time



Its Demo Time!

Most of the code and instructions for this demo can be found here:

https://github.com/capeprivacy/image-classification-onnx

Links

https://docs.capeprivacy.com/getting-started/

https://discord.gg/nQW7YxUYjh

Thank You

Contact Us:

capeprivacy.com