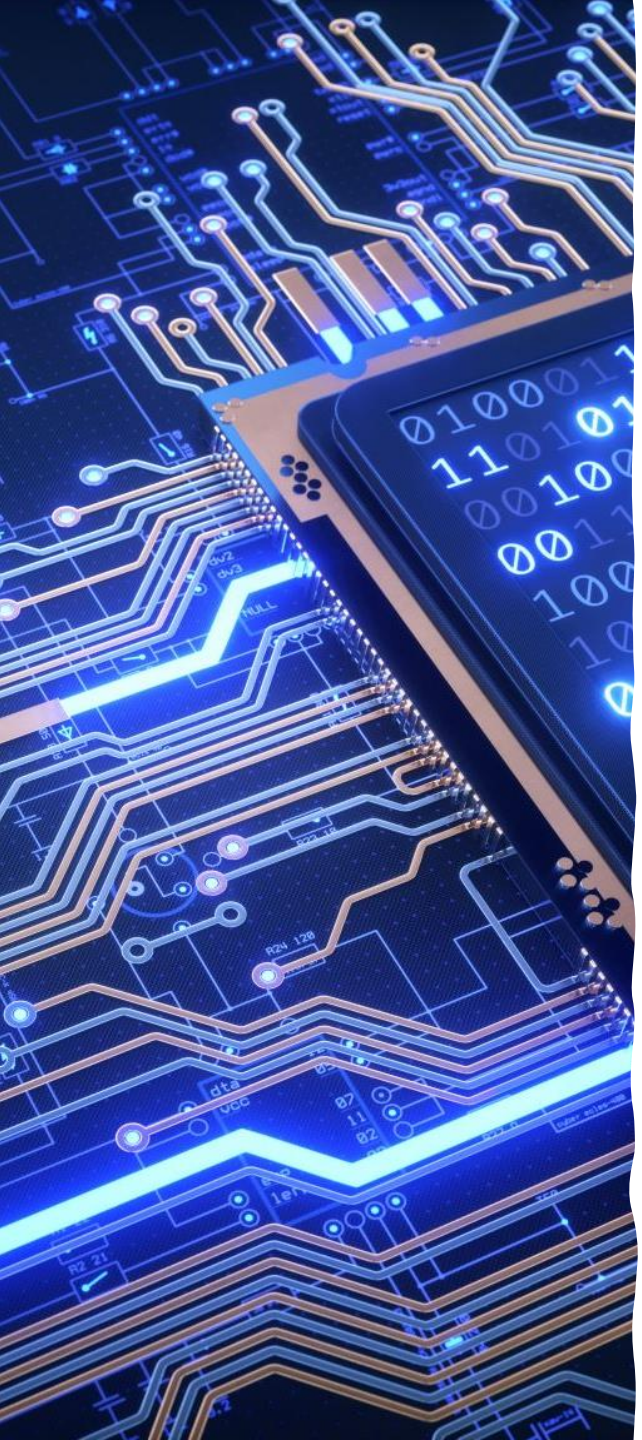


IoT-Cloud and Blockchain

Sha256

Phillip G. Bradford

University of Connecticut, Stamford CT. **USA**

phillip.bradford@uconn.edu

Outline

Motivation

Simple Python SHA256

Code

Running

Motivation

Quickly distinguish large files

Fingerprints

Determine whether large files have changed

Message digest hash functions

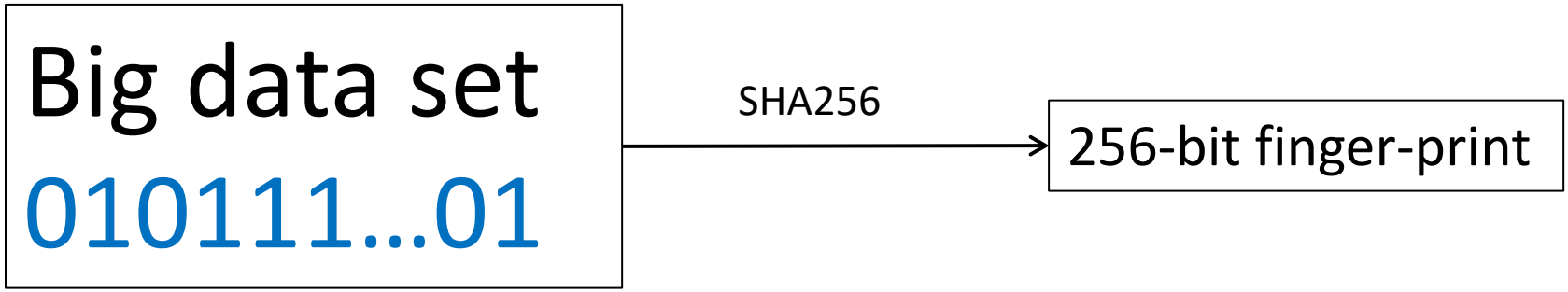
Take very large files and produce 256-bit fingerprints

Pigeonhole principle

practically foiled by size $2^{256}+1$ is *very* large

Sha256 has no known collisions

SHA256 standard



SHA-256: Secure Hash (Digest) Algorithm

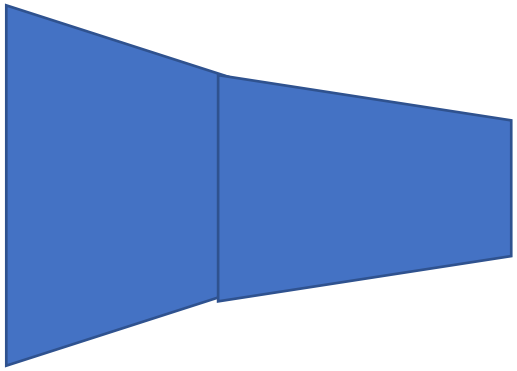
- Designed by NSA and standardized by NIST (1994-2001)

SHA-256 Hashing

- One-way
- Digital finger-prints
- Digest takes up to 2^{64}-1 bit input and outputs 256-bit output

Our simple Blocks

DATA



Ce50f7d76ff4d7…

Fingerprint output

Preparation

Python3

pip3 install pycryptodome

https://github.com/wonder-phil/BlockchainRPiExtras.git

Sha256

```
import datetime
import hashlib

class ExampleHash:

    def __init__(self):
        self.hashFunction = hashlib.new('sha256')

    def compHash(self, data):
        myBytes = data.encode()
        self.hashFunction.update(myBytes)
```