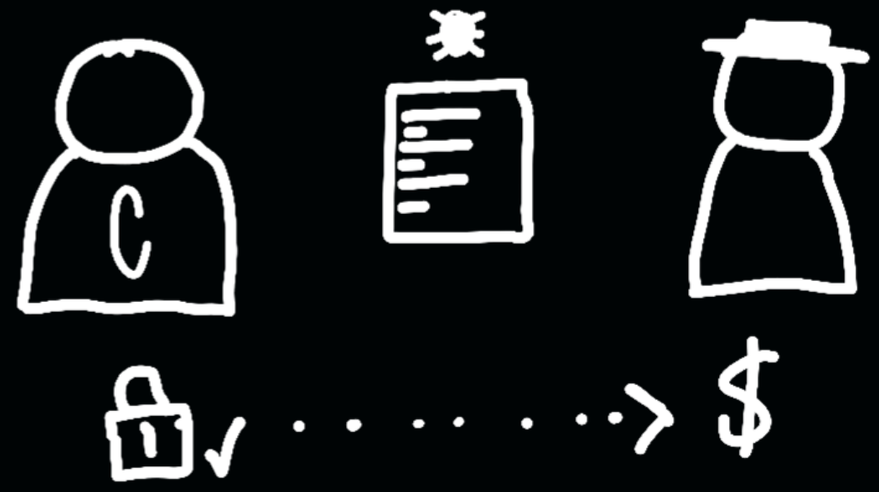
Eliminating Triage Intermediaries for Zero-day Exploits Using a Decentralized Payout Protocol

Clara Maine

TruCol | ***trucol.io***

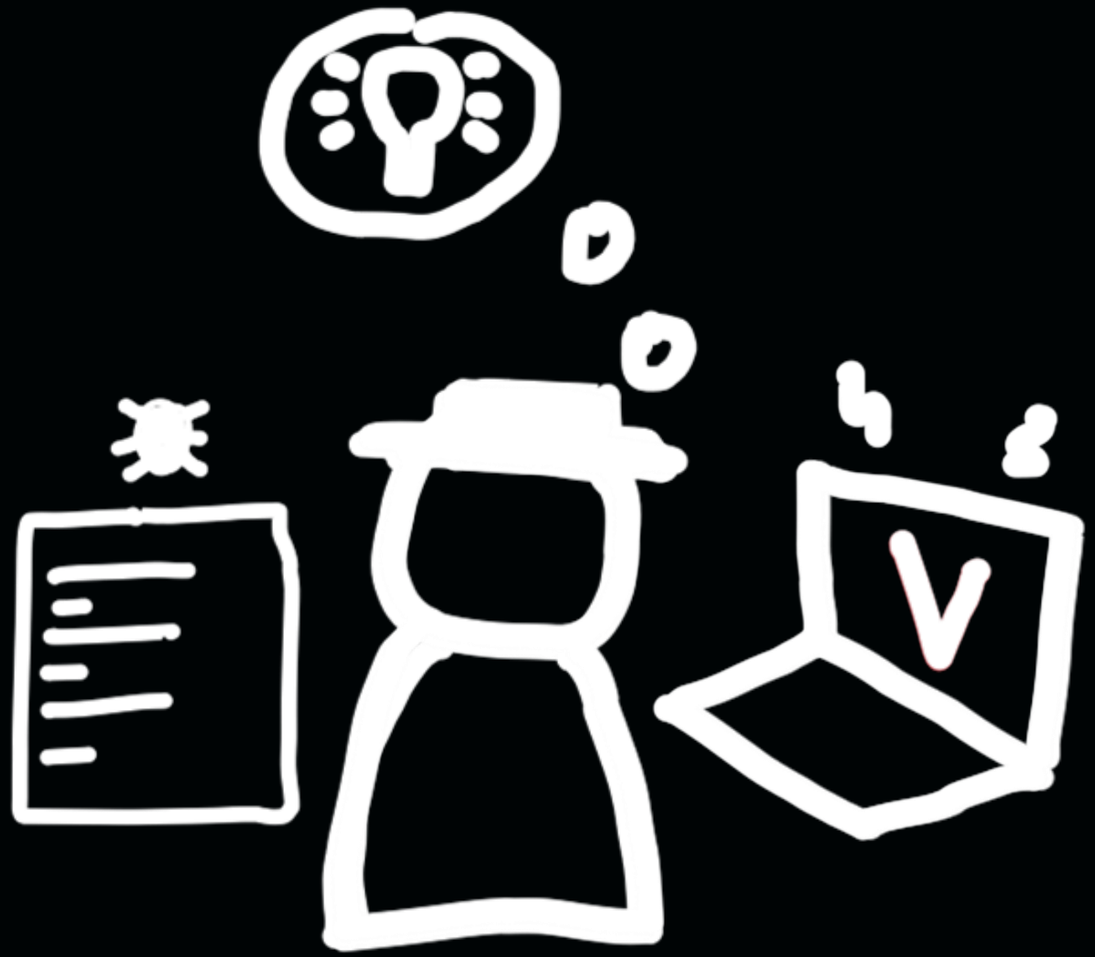
1. Cybersecurity vulnerabilities are increasingly relevant

1. **Bug Bounties** are an important tool for addressing these vulnerabilities in a mutually beneficial way
    - Hackers receive compensation for their work
    - Companies can harness a global network of intelligence to ensure secure software



Cybersecurity Triage: Status-Quo

Exploit Identification

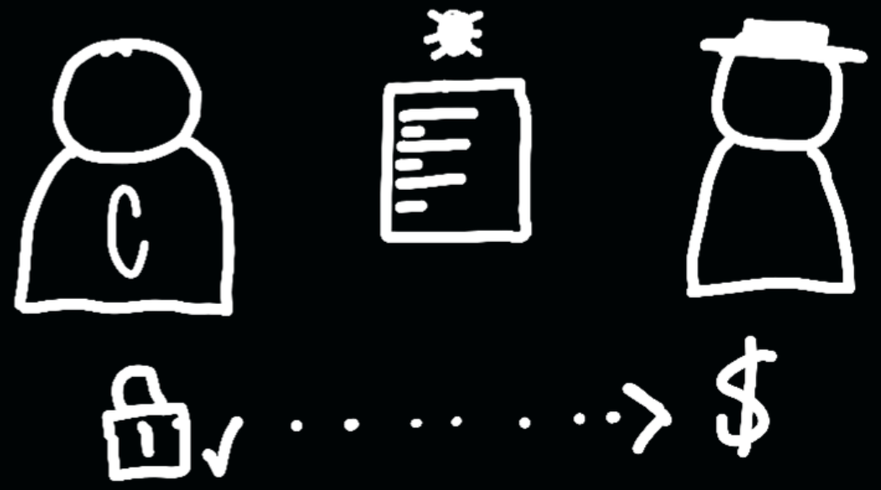


Barriers to Direct Disclosure

1. Legality

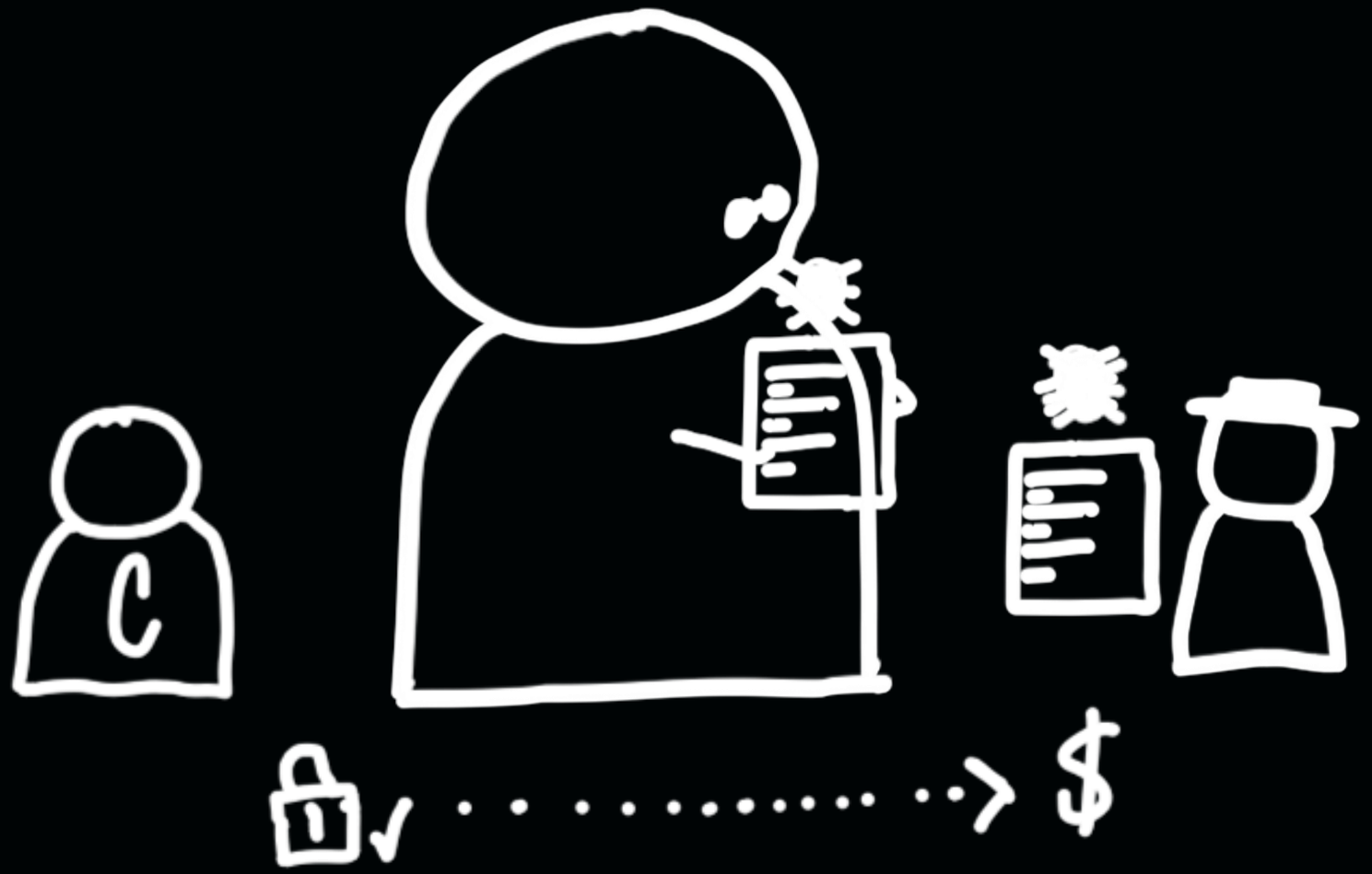
1. Difficulty in convincing owners of severity and negotiating a fair price

1. Small firms and users with no budget or technical skills to deal with the exploit



Cybersecurity Triage: Status-Quo

Triage

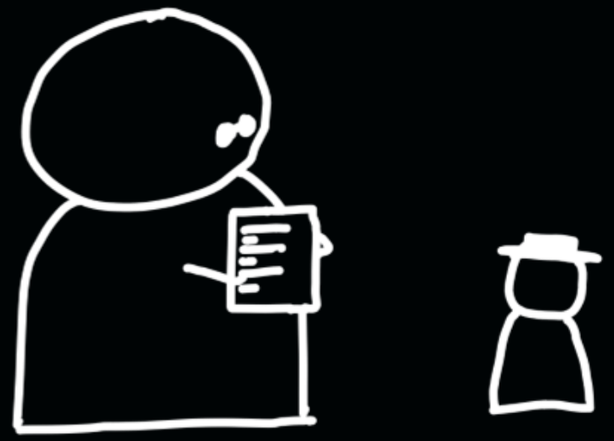


Independent exploit evaluation

Negotiating payout

Triage Intermediaries are Costly

The current process is very taxing on ***time*** and ***money***.



Intermediaries must understand the exploit and verify its severity

They must assess the impact value of the exploit to the company

Cybersecurity Triage: Automated

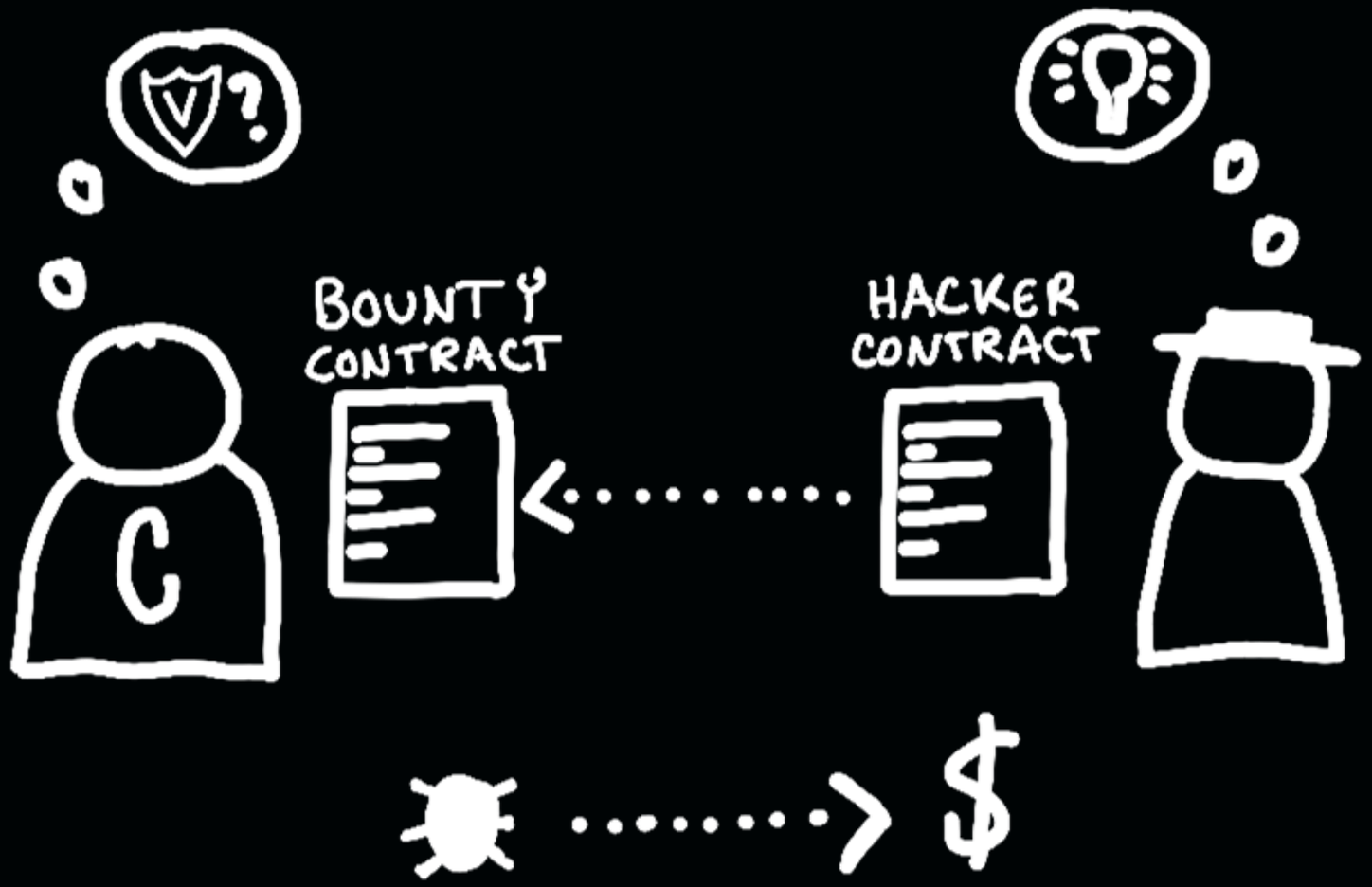
If direct vulnerability disclosure is not feasible

and triage intermediaries are expensive

Could there be another way?

Cybersecurity Triage: Automated

The TruSec Protocol



The TruCol Team

- 2020: Wanted to set programming bounties, encountered middlepersons
- Jan 2021: Assembled student team from Delft and Radboud university.
- Feb 2021: Competed at ETHDenver, developed TruCol protocol
- July 2021: Presented at Ethereum conference in Paris

Akke Toeter
Co-founder

Victoria Bosch
Co-Founder

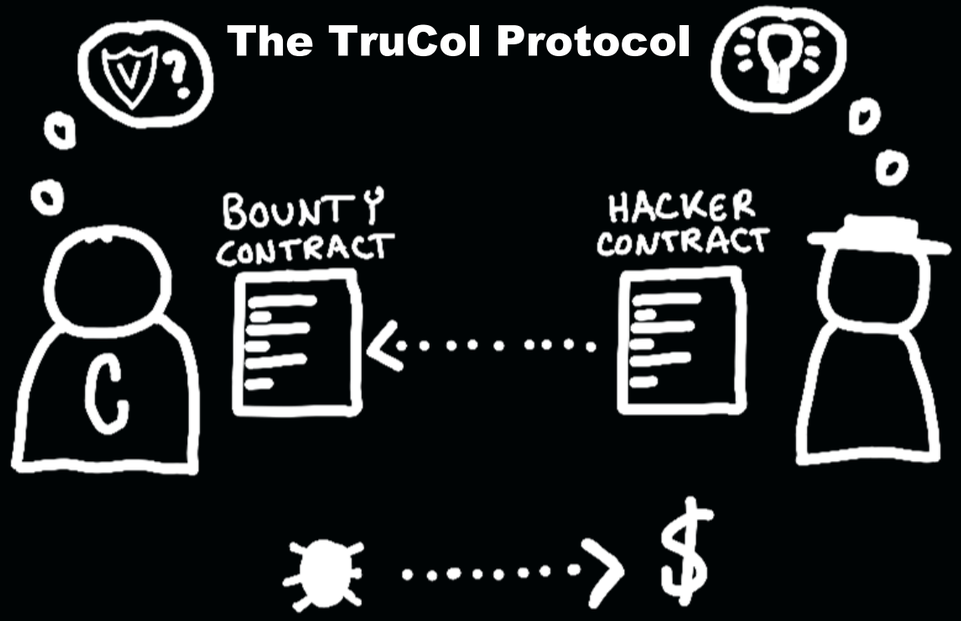
Clara Maine
Developer

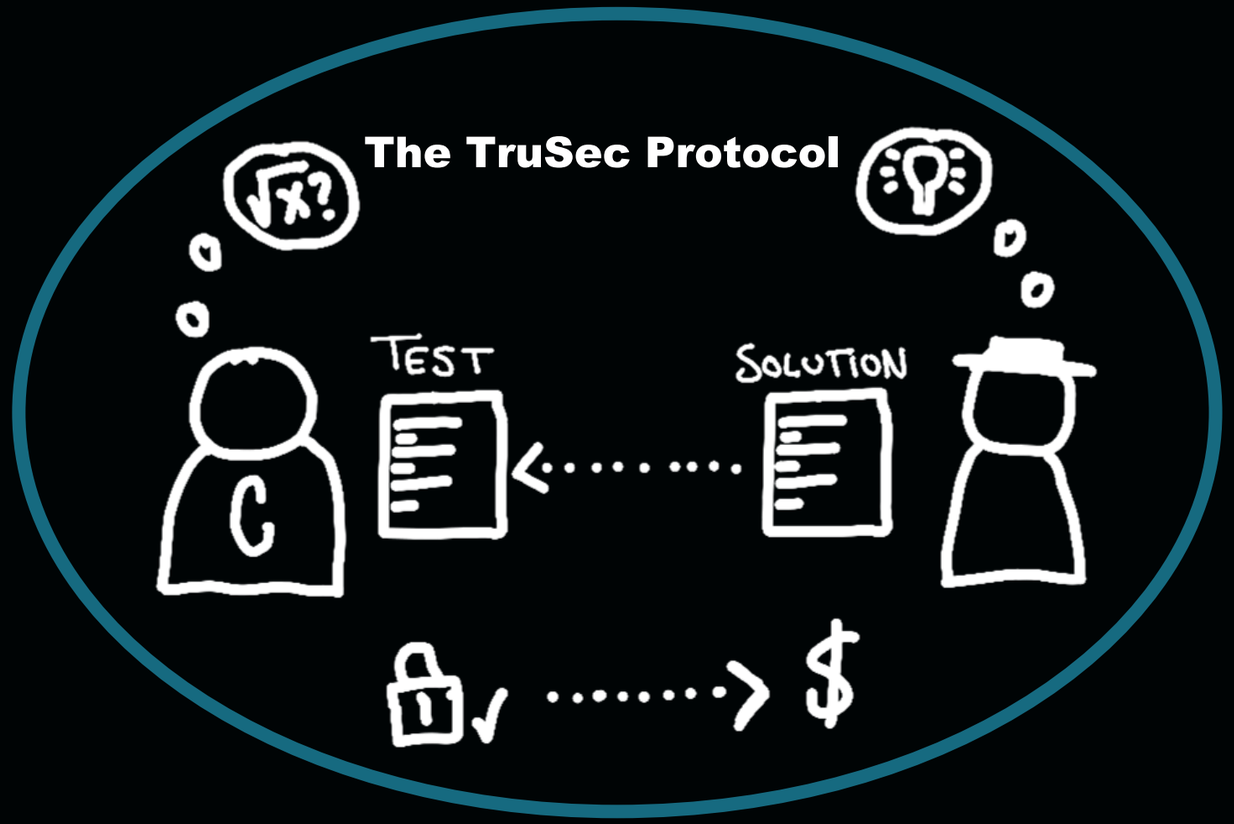
Rashim Charles
Business Dev

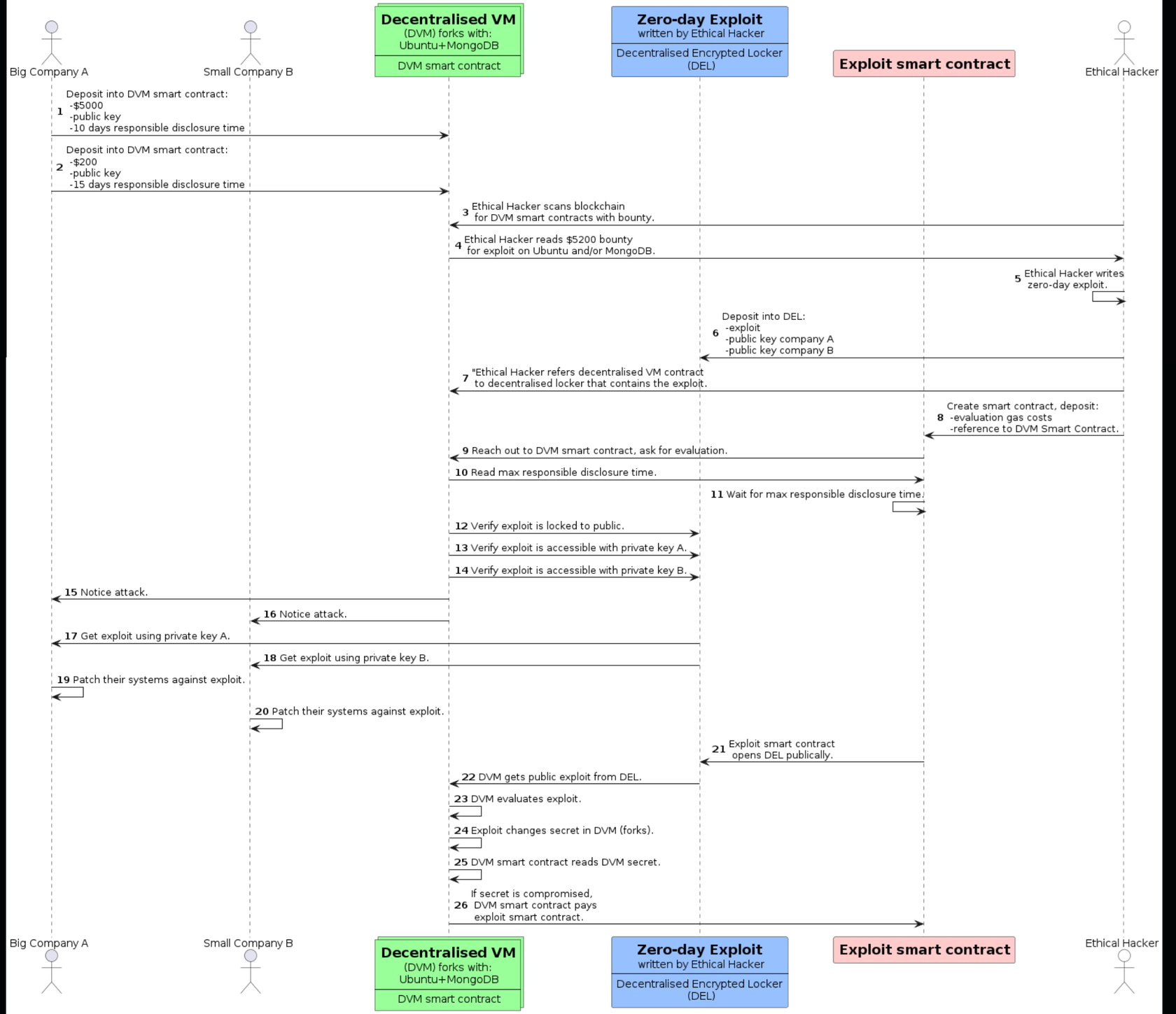
Chihab Amghane
Developer

Marc Droog
Co-founder

TruCol → TruSec

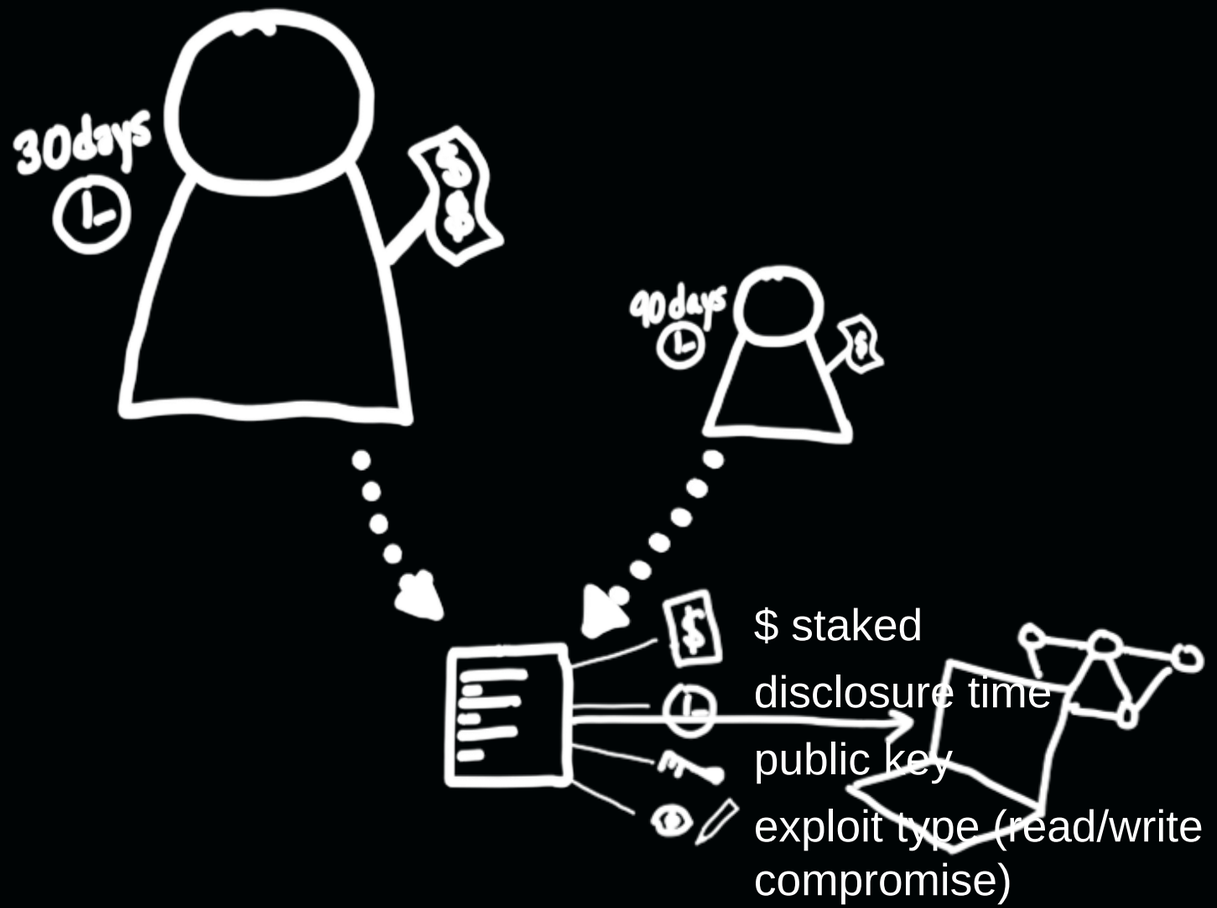






The TruSec Protocol

- Companies pool bounties into smart contract
- Smart contract controls decentralised virtual machine (DVM)
- DVM contains software stack and write secret



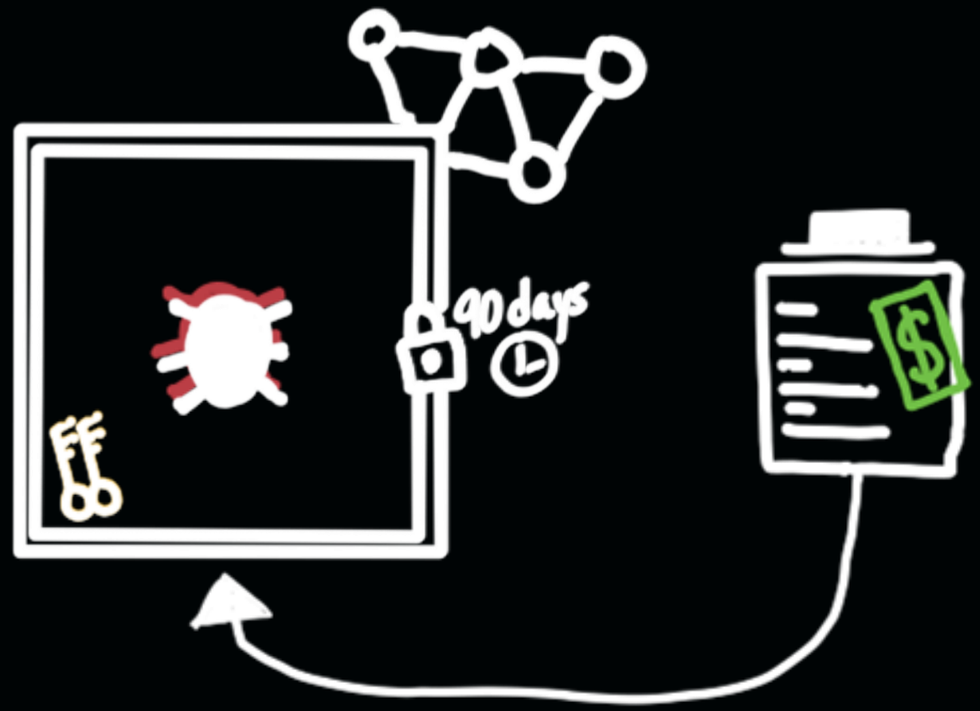
Staking the security of software stacks

The TruSec Protocol

- Hackers see:
  - amount staked
  - when they will get the reward
  - which software stack
- Develop zero-day exploit
  - Put it in decentralised locker
  - Encode zero day exploit with stakeholder public keys

The TruSec Protocol

- Hackers see:
  - amount staked
  - when they will get the reward
  - which software stack
- Develop zero-day exploit
  - Put it in decentralised locker
  - Encode zero day exploit with stakeholder public keys
- Write smart-contract:
  - Pay gas fees for DVM evaluation
  - Specify decentralised locker location



The TruSec Protocol

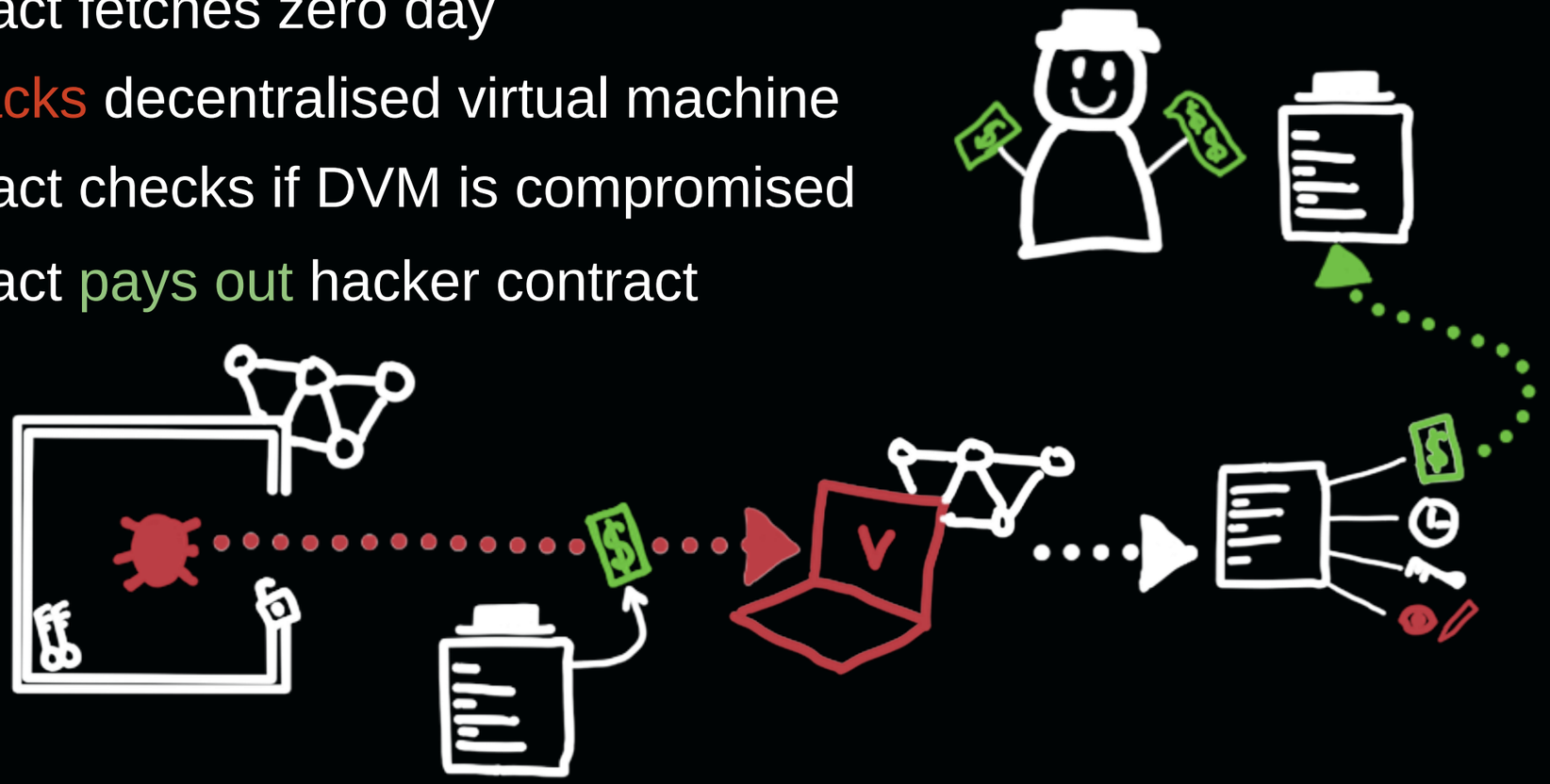
Before disclosure time:

1. Bounty contract checks locker is still locked
2. Companies use private key to get zero day exploit
3. Companies patch their systems

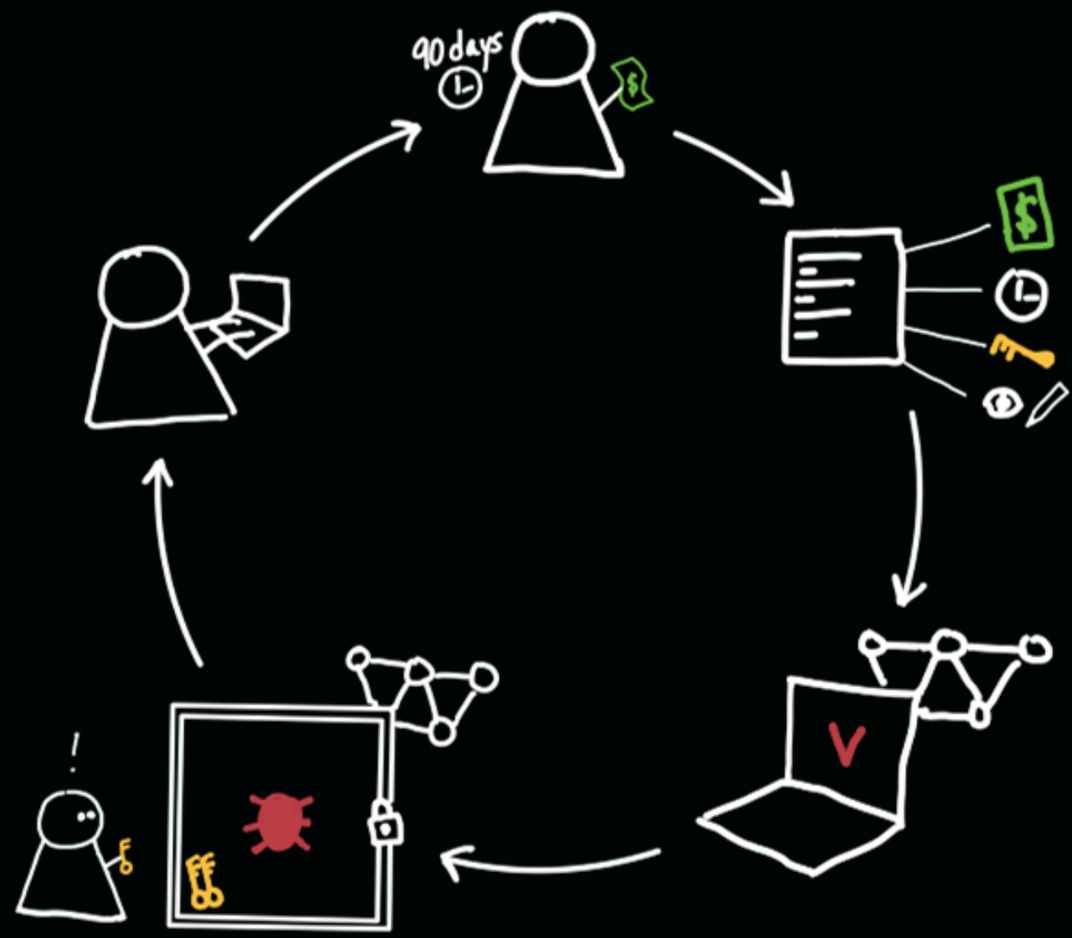
The TruSec Protocol

After disclosure time:

1. Locker opens
2. Bounty contract fetches zero day
3. Zero day attacks decentralised virtual machine
4. Bounty contract checks if DVM is compromised
5. Bounty contract pays out hacker contract



A New Cycle Begins



Companies can re-allocate new stakes on their patched devices

Protocol Scope

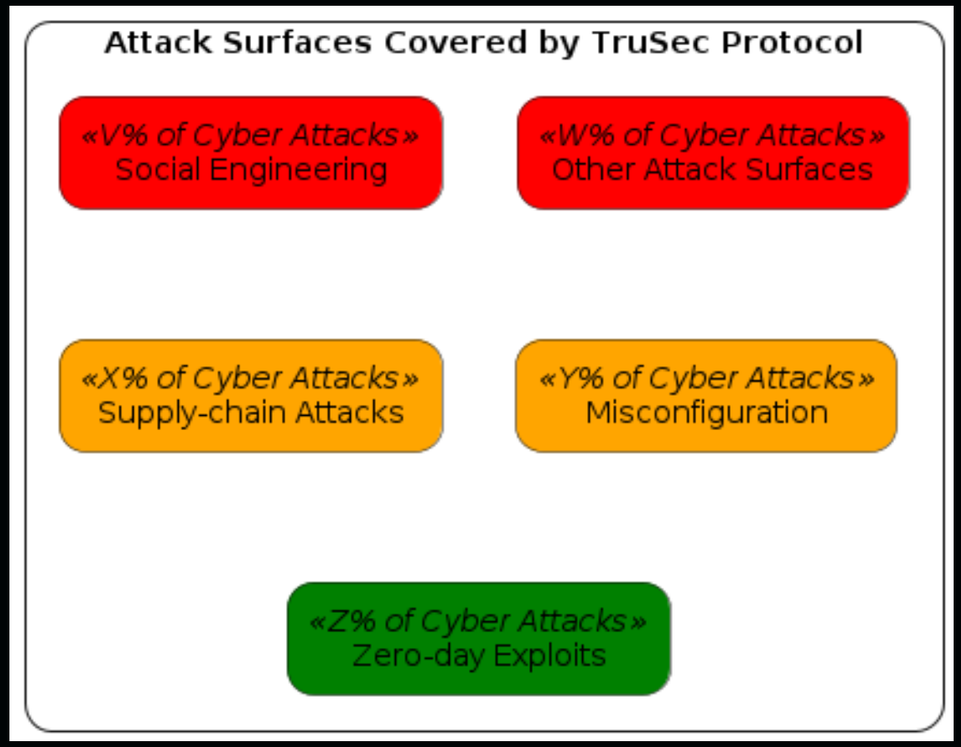
Designed for zero day exploits

Applicable to:

- supply-chain attacks
  - misconfiguration

Does not cover:

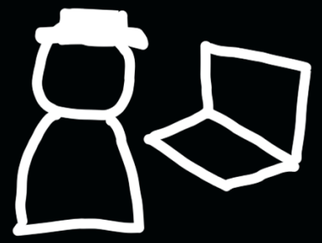
- Social Engineering
- Other Attack Surfaces



Limitations

- All-or-nothing, the bounty contract must be correct
- Protocol only applies to deterministically verifiable zero-day exploits
- DVMs are costly to operate
- Allows over-emphasising quantitative security level

To Summarize



Hackers:

- Get automatic payout without disputes
- Know how much they earn up front
- Know when they will get the money

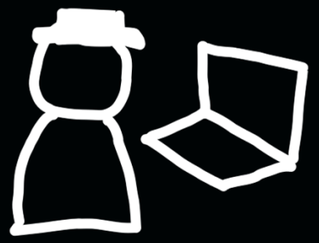
Companies:

- Show customers “how secure” their OS software is against zero-day exploits in terms of $ staked.
- Gain transparency in zero-day exploit market

Decision makers:

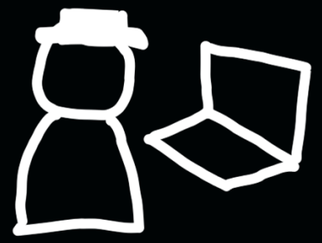
- Can allocate cyber-security resources more efficiently.

Roadmap



- Get community advice and input
- Attract expertise on decentralised virtual machines
- Select implementation platforms
- Raise funds to build MWE
- Get community feedback
- Adjust implementation
- Security Audit
- Gradually raise bounties until:
  - hacks on protocol move to software hacks

Moving Forward



Companies:

- What is important to you?
  - Compliance, exploit types, configuration staking, etc.?
- Advise us on decentralised virtual machine development

Devs:

- Join the BUIDL at github.com/TruSec

Get in touch!

- trucol@protonmail.com

Acknowledgements

Tony Smith - F-Secure

Leon Botros - IRMA

Wouter - Ethical Hacker

Lidia Giuliano - Blackhat Speaker Coach