SANS

HACKFEST SUMMIT 2023
WORKSHOP

BLOCKCHAIN SECURITY FOR BLUE TEAMS

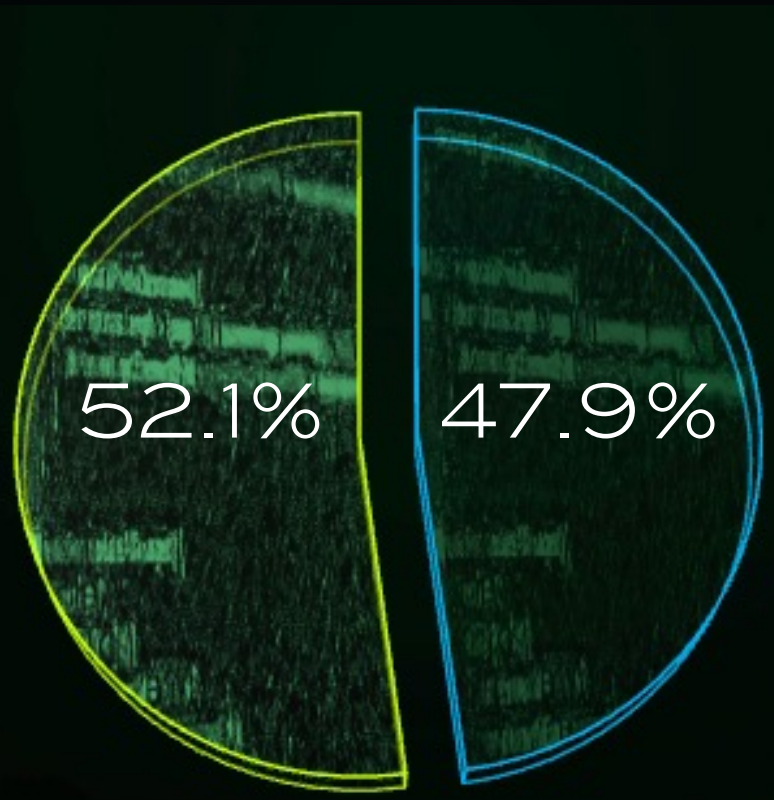
STEVEN WALBROEHL

CO-FOUNDER @ HALBORN -

AUTHOR & INSTRUCTOR || SEC554:
BLOCKCHAIN & SMART CONTRACT SECURITY

THE TOP 50 CYBER SECURITY INCIDENTS INVOLVING DIGITAL ASSETS.

ON-CHAIN INCIDENTS



OFF-CHAIN INCIDENTS

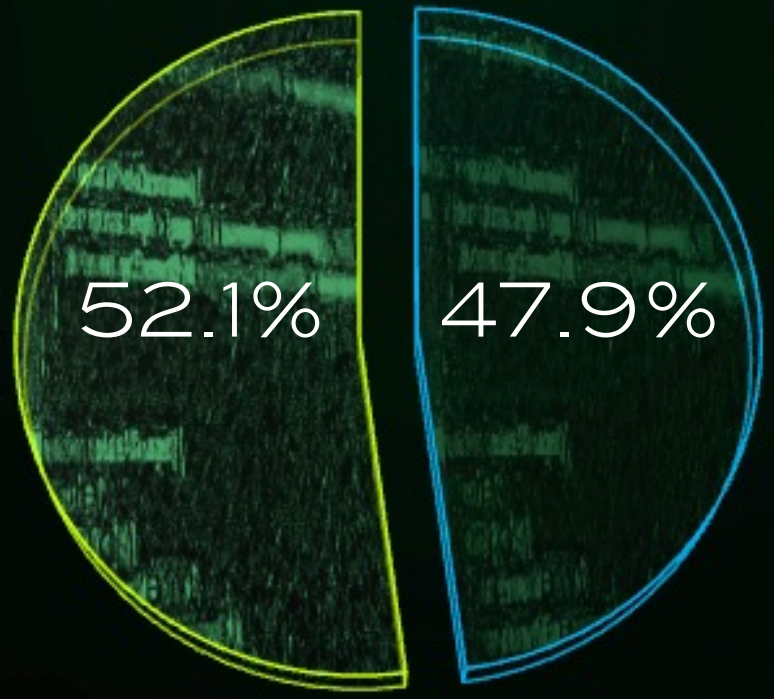
INCIDENT ROOT CAUSE

THE TOP 50 CYBER SECURITY INCIDENTS INVOLVING DIGITAL ASSETS.

ON-CHAIN INCIDENTS

11.9%
FINANCIAL OR ENVIRONMANTAL ATTACK

40.2%
SMART CONTRACT EXPLOITATION



INCIDENT ROOT CAUSE

OFF-CHAIN INCIDENTS

2.7%
TRADITIONAL WEB-2 INFRASTRUCTURE

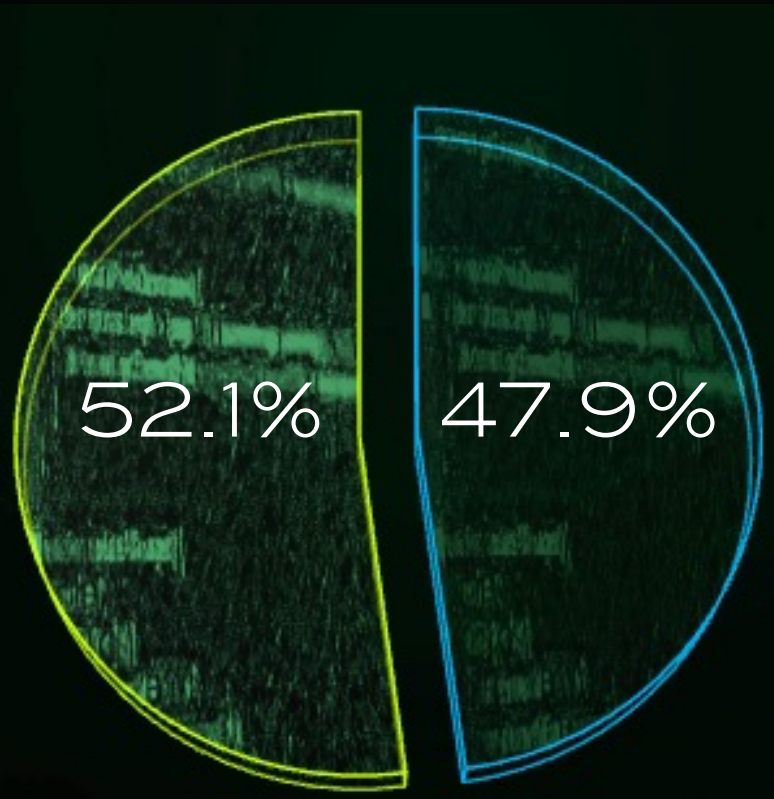
45.2%
PRIVATE KEY LOSS / THEFT

LETS FOCUS ON SOLUTIONS TO PROTECT ON-CHAIN CONTRACTS

ON-CHAIN INCIDENTS

11.9%
FINANCIAL OR ENVIRONMANTAL ATTACK

40.2%
SMART CONTRACT EXPLOITATION



INCIDENT ROOT CAUSE

OFF-CHAIN INCIDENTS

2.7%
TRADITIONAL WEB-2 INFRASTRUCTURE

45.2%
PRIVATE KEY LOSS / THEFT

DEFENSE IN DEPTH

A STRATEGY TO UTILIZE MULTIPLE LAYERS OF DIFFERENT SECURITY CONTROLS TO PROTECT ASSETS.

SECURITY CONTROL CATEGORIES

DETECTIVE

- Smart Contract Monitoring
- Threat Intelligence
- Behavior Analytics

SECURITY CONTROL CATEGORIES

DETECTIVE

SMART CONTRACT MONITORING

THREAT INTELLIGENCE

BEHAVIOR ANALYTICS

DIRECTIVE

AUDITING

VULNERABILITY SCANNING

SECURE CODING

SECURITY CONTROL CATEGORIES

CORRECTIVE

- Pausing / Stopping Functions
- Cyber Insurance
- Incident Response Solutions

DETECTIVE

- Smart Contract Monitoring
- Threat Intelligence
- Behavior Analytics

DIRECTIVE

- Auditing
- Vulnerability Scanning
- Secure Coding

SECURITY CONTROL CATEGORIES

CORRECTIVE

Pausing / Stopping Functions

Cyber Insurance

Incident Response Solutions

AFTER INCIDENT
AFTER THREAT

DETECTIVE

Smart Contract Monitoring

Threat Intelligence

Behavior Analytics

DURING INCIDENT
AFTER THREAT

DIRECTIVE

Auditing

Vulnerability Scanning

Secure Coding

BEFORE INCIDENT
BEFORE THREAT

SMART CONTRACT SECURITY CURRENTLY HAS A GAP IN THE DEFENSE IN DEPTH STRATEGY

??

SMART CONTRACT SECURITY CURRENTLY HAS A GAP IN THE DEFENSE IN DEPTH STRATEGY

PREVENTIVE

BEFORE INCIDENT OCCURS ON ASSETS WITH ACTIVE THREATS

DIRECTIVE

VERIFY CONTROLS IMPLEMENTED THROUGH FORMAL TRAINING, REVIEW, SCANNING AND ALIGNMENT TO SECURITY BEST PRACTICES

- DIRECTIVE
- PREVENTIVE
- DETECTIVE
- CORRECTIVE

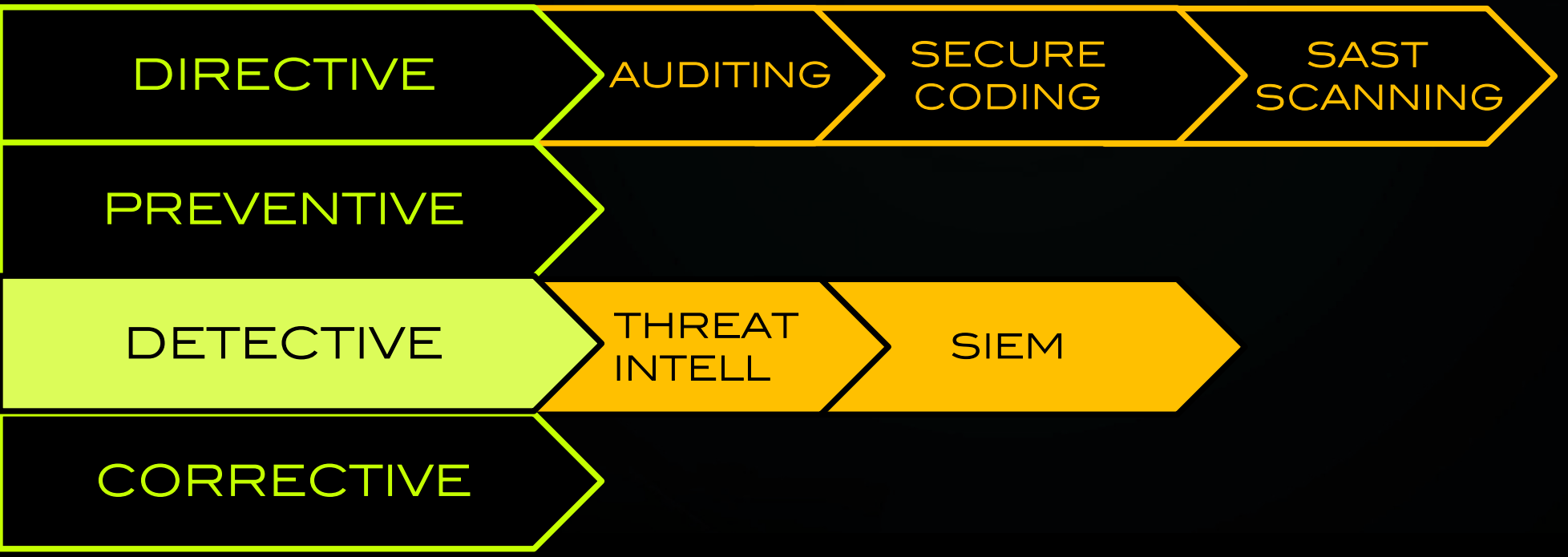
DIRECTIVE

SMART CONTRACT SECURITY SOLUTIONS

- AUDITING: THIRD PARTY WEB3 SECURITY TESTERS
- SECURE CODING: SEC554, BLOCKCHAIN COUNCIL
- VULNERABILITY SCANNING: SLITHER, CERTORA, MYTHRIL

DETECTIVE

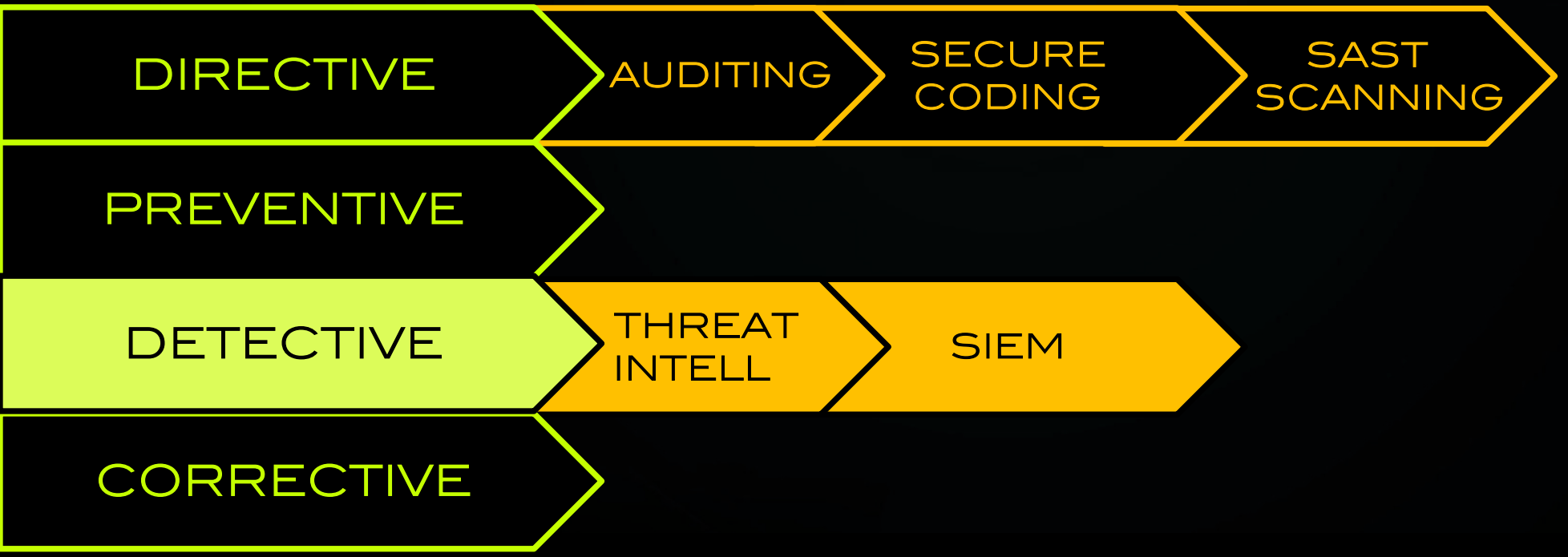
MONITOR, IDENTIFY, AND ALERT ON SECURITY ISSUES



DETECTIVE

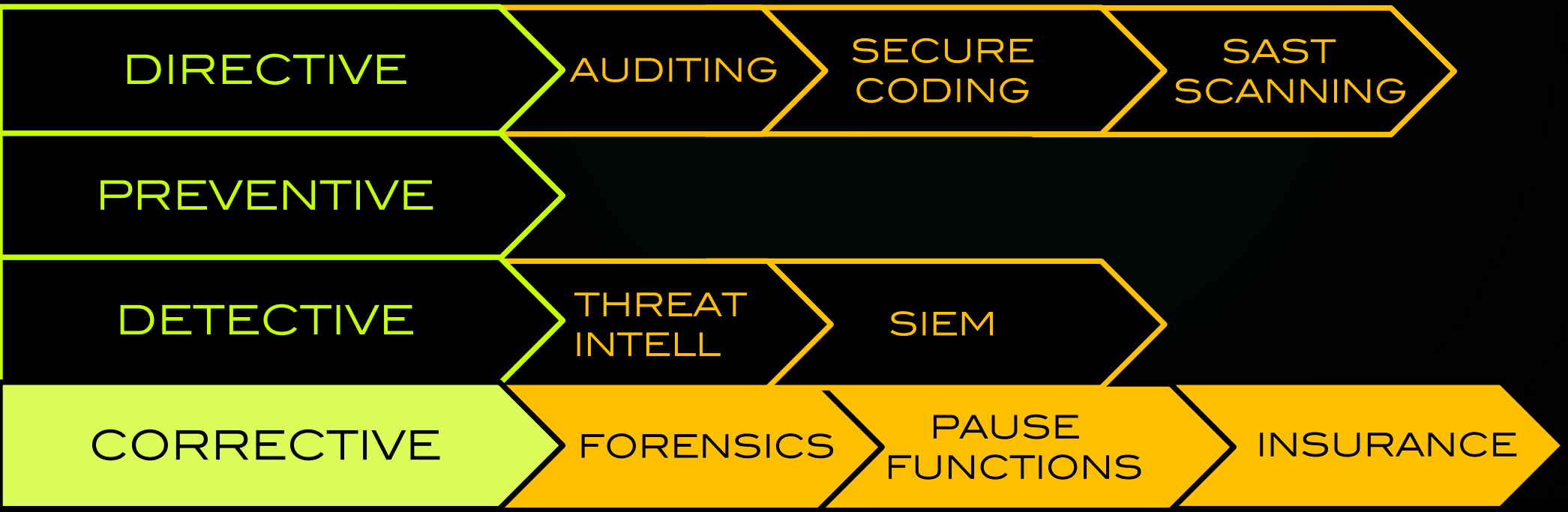
SMART CONTRACT SECURITY SOLUTIONS

- THREAT INTELL: CIPHERTRACE, ELLIPTIC
- SIEM: TENDERLY, FORTA, PHALCON.XYZ, BLOCKNATIVE



CORRECTIVE

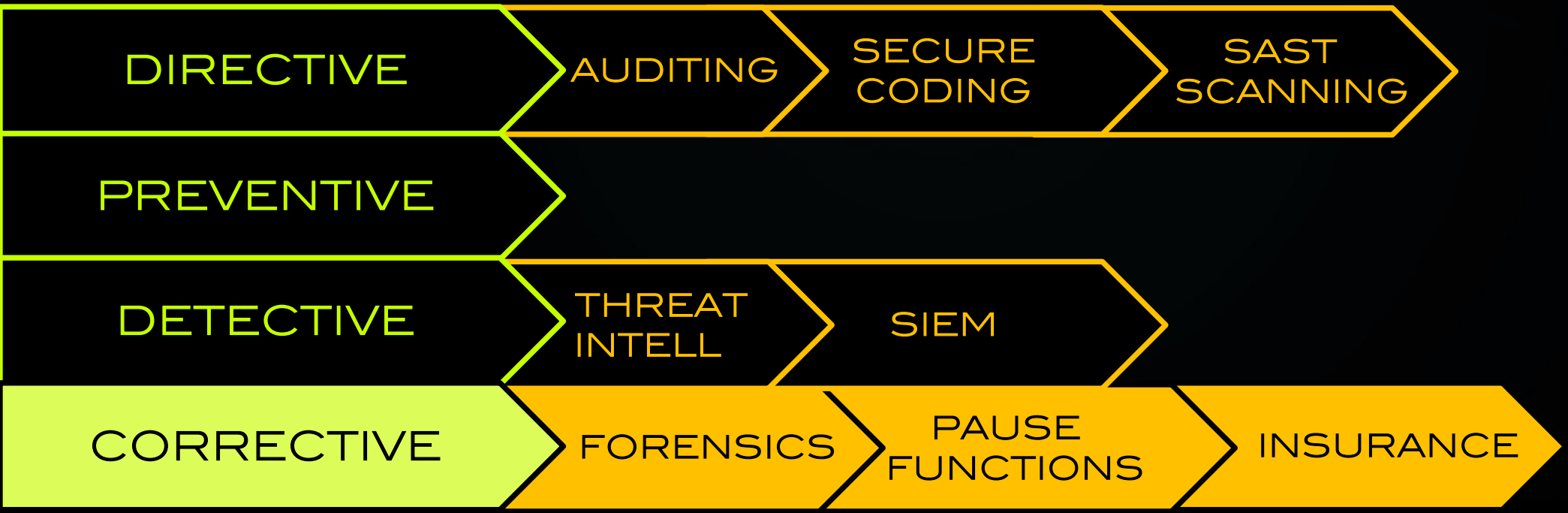
RESPOND AND RECOVER FROM VULNERABILITIES AND ENSURE THAT SIMILAR INCIDENTS ARE NOT REPEATED



CORRECTIVE

SMART CONTRACT SECURITY SOLUTIONS

- FORENSICS: CHAINALYSIS, KYC PARTNERS
- INSURANCE: VARIOUS PROVIDERS



LETS LOOK AT SOME TOOLS

- MONITORING: TENDERLY.CO
- FORENSICS: HTTPS://EXPLORER.PHALCON.XYZ/
- INVESTIGATION: LENS.ELEMENTUS.IO
- STATIC ANALYSIS: SLITHER
- MEMPOOL - BLOCKNATIVE

PREVENTIVE

WHY ARE PREVENTIVE SECURITY SOLUTIONS A CHALLENGE IN SMART CONTRACTS?

PREVENTIVE ??

PREVENTIVE

WHY ARE PREVENTIVE SECURITY SOLUTIONS A CHALLENGE IN SMART CONTRACTS?

CENTRALIZED vs **DECENTRALIZED**

- CENSORSHIP AND CONTROL
- CUSTODIAL RISKS
- RESPONSIBILITY TO OTHERS
- LACK OF TRANSPARENCY
- AVAILABILITY / LIABILITY COSTS

- LACK OF CONTROL
- END USER RISKS
- SELF-RESPONSIBLITY
- MALICIOUS GOVERNANCE
- REACTION TIME INCREASE

PREVENTIVE

CENTRALIZED vs DECENTRALIZED

How can we create TRUST in a TRUSTLESS environment to protect assets?

!!:SERAPH

A PREVENTIVE SECURITY SOLUTION FOR SMART CONTRACTS

HALBORN

SERAPH

POLICY ENGINE

Create custom policies on individual functions.

INTRUSION PREVENTION

Mem-Pool to Prevent On-Chain Incidents from Occuring

FUNCTION FIREWALL

Automated decision logic.

SEGREGATION OF DUTIES

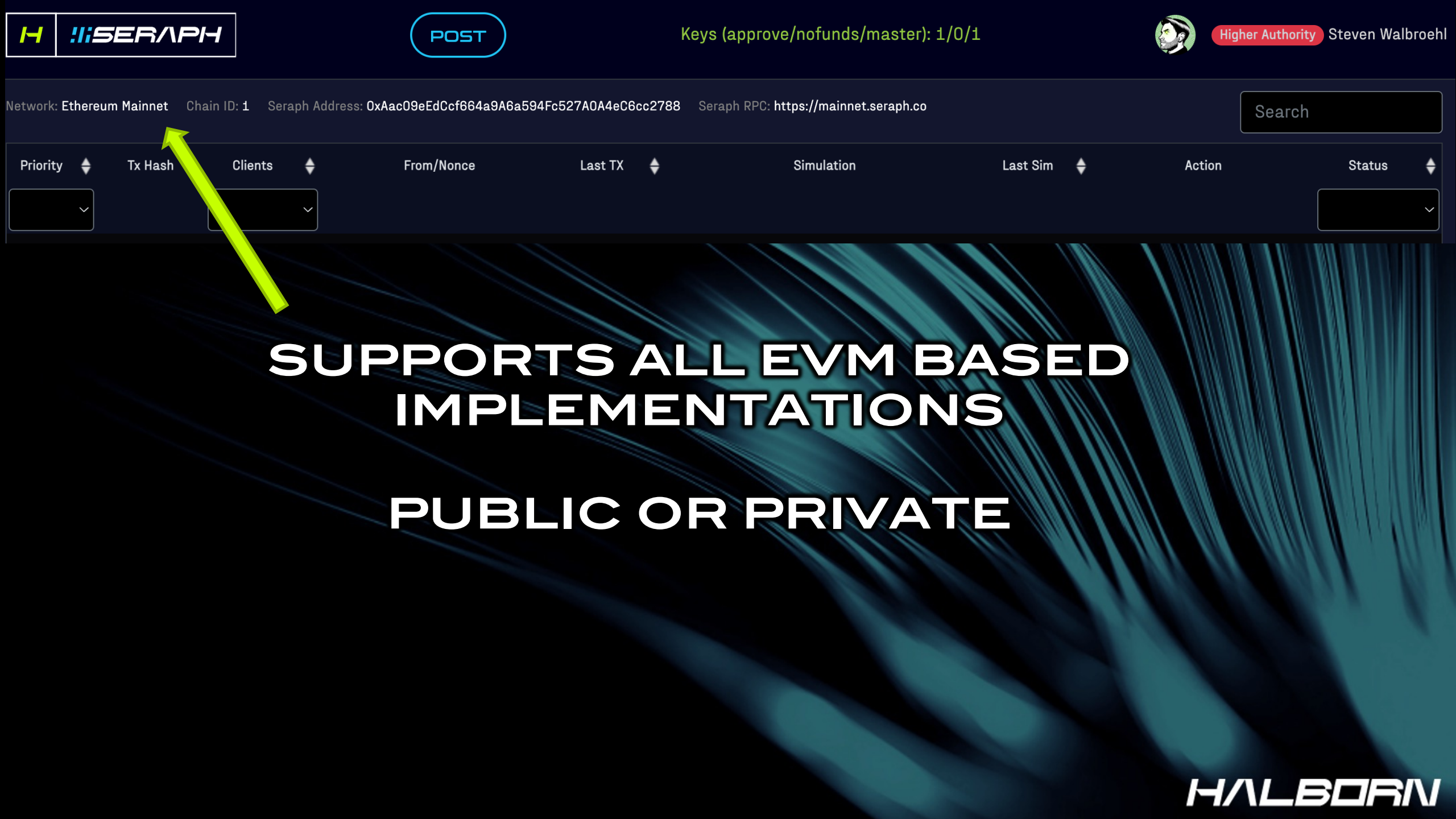
Notorize based on authorization scope.

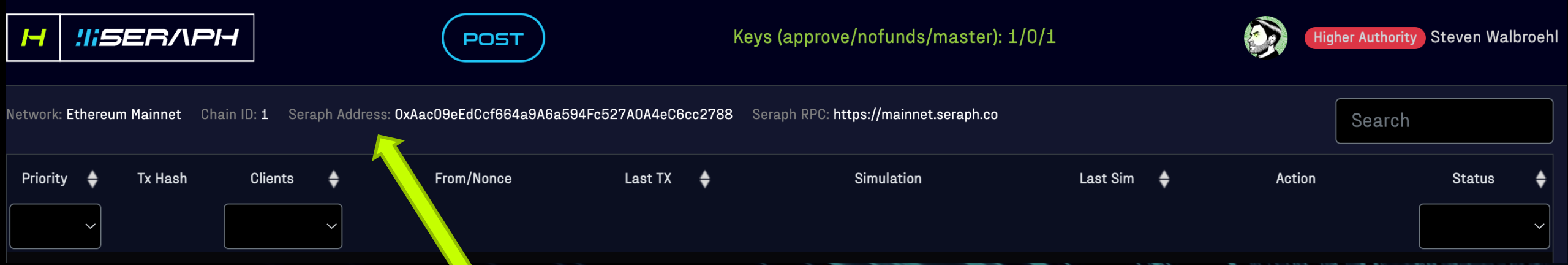
TRANSACTION AUDIT

Simulate events before they get confirmed on-chain

NON-CUSTODIAL

Keep your private keys private.


SERAPH
POST
Keys (approve/nofunds/master): 1/0/1
Higher Authority
Steven Walbroehl
Network: Ethereum Mainnet
Chain ID: 1
Seraph Address: 0xAac09eEdCcf664a9A6a594Fc527A0A4eC6cc2788
Seraph RPC: https://mainnet.seraph.co
Search
Priority
Tx Hash
Clients
From/Nonce
Last TX
Simulation
Last Sim
Action
Status
SUPPORTS ALL EVM BASED IMPLEMENTATIONS
PUBLIC OR PRIVATE
HALBORN

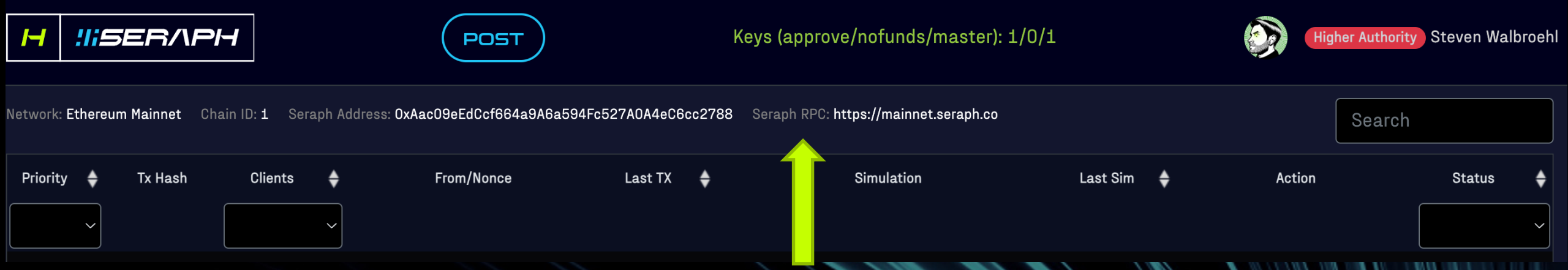


ON-CHAIN SOLUTION

SERAPH IS A SMART CONTRACT THAT EXISTS ON DLT

- SUPPORTS ALL EVM BASED IMPLEMENTATIONS

HALBORN



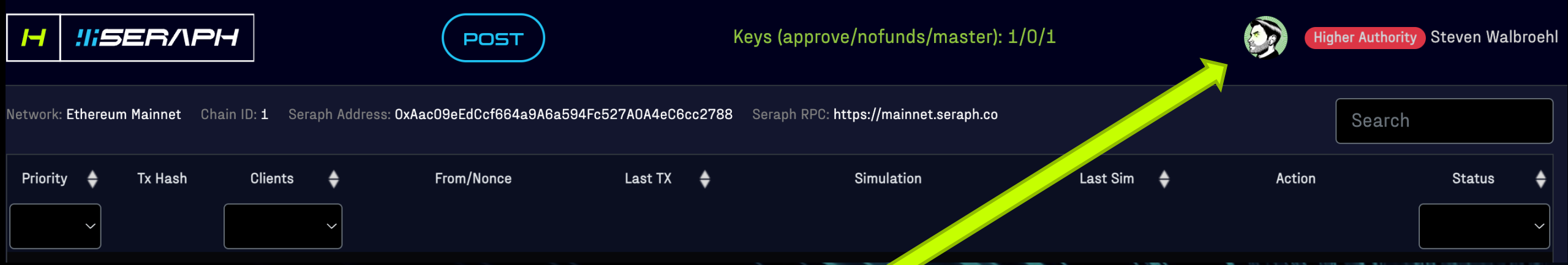
SERAPH ENTERPRISE
HOST YOUR OWN RPC ENDPOINT

OR

SERAPH AS A SERVICE
CONNECT TO THE MANAGED RPC ENDPOINT

- SUPPORTS ALL EVM BASED IMPLEMENTATIONS
- ON-CHAIN SOLUTION

HALBORN



SIGN IN AND ASSIGN ROLES WITH YOUR EXISTING IAM.

(O365 / GSUITE / OID / SAML / OKTA ETC..)

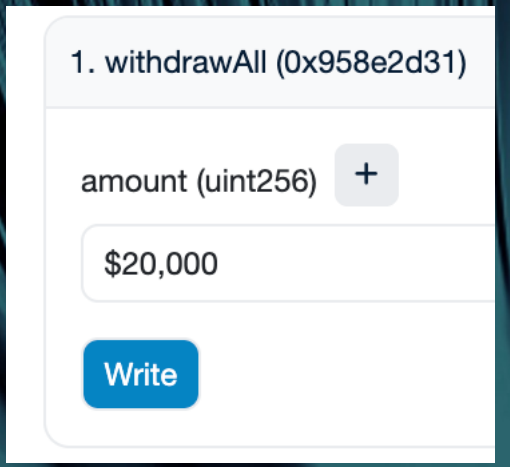
NO PRIVATE KEYS REQUIRED

- SUPPORTS ALL EVM BASED IMPLEMENTATIONS
- ON-CHAIN SOLUTION
- SELF-HOST OR SAAS VERSIONS

HALBORN

A SMART CONTRACT FUNCTION PROTECTED WITH SERAPH IS EXECUTED

`function withdrawAll withSeraph{$20,000 }`



HALBORN

H | SERAPH

POST

Keys (approve/nofunds/master): 1/0/1

Higher Authority Steven Walbroehl

Network: Ethereum Mainnet Chain ID: 1 Seraph Address: 0xAac09eEdCcf664a9A6a594Fc527A0A4eC6cc2788 Seraph RPC: https://mainnet.seraph.co

Search

Priority	Tx Hash	Clients	From/Nonce	Last TX	Simulation	Last Sim	Action	Status

CLIENT NAME	CALLER	FUNCTION	DATA	MSG.VALUE	Action	Status
TestClient	0x8a9fd70bf449e62e89dc12c24aa9fd99978bfabf	withdrawAll	[{"name":"amount","value":"20000"}]	0x0	Approve Reject	Pending Approval

waiting for confirmation

AN ALERT IS SENT TO THE APPROVER TO REVIEW THE TRANSACTION

HALBORN

SERAPH | POST | Keys (approve/nofunds/master): 1/0/1 | Higher Authority Steven Walbroehl

Network: Ethereum Mainnet Chain ID: 1 Seraph Address: 0xAac09eEdCcf664a9A6a594Fc527A0A4eC6cc2788 Seraph RPC: https://mainnet.seraph.co

Search

Priority	Tx Hash	Clients	From/Nonce	Last TX	Simulation	Last Sim	Action	Status

CLIENT NAME	CALLER	FUNCTION	DATA	MSG.VALUE	Action	Status
TestClient	0x8a9fd70bf449e62e89dc12c24aa9fd99978bfabf	withdrawAll	[{"name":"amount","value":"20000"}]	0x0	Approve Reject	Rejected

waiting for confirmation

THE APPROVER LOGS IN TO SERAPH WITH THEIR GMAIL.

AFTER REVIEWING THE TRANSACTION AND CHECKING THE POLICY, A DECISION IS MADE TO REJECT.

HALBORN

Network: Ethereum Mainnet Chain ID: 1 Seraph Address: 0xAac09eEdCcf664a9A6a594Fc527A0A4eC6cc2788 Seraph RPC: https://mainnet.seraph.co

Search

Priority	Tx Hash	Clients	From/Nonce	Last TX	Simulation	Last Sim	Action	Status

SERAPH UPDATES ITS ONCHAIN STATE WITH THE REJECT DATA

SERAPH

Tx Hash: 0x357f7b7579c823edf313ed4723429986fecb1f3facda3d782ed2fe42638083bb

Approve Tx Hash:

Priority: 1

Simulation: link

TX INFO

approveTx:

▶ tx:

status: sent

timestamp: 1691347617510

priority: 1

THE SMART CONTRACT CALLED THEN CHECKS THE APPROVAL STATE IN SERAPH ON-CHAIN

HALBORN

H | SERAPH

POST

Keys (approve/nofunds/master): 1/0/1

Higher Authority Steven Walbroehl

Network: Ethereum Mainnet Chain ID: 1 Seraph Address: 0xAac09eEdCcf664a9A6a594Fc527A0A4eC6cc2788 Seraph RPC: https://mainnet.seraph.co

Search

Priority	Tx Hash	Clients	From/Nonce	Last TX	Simulation	Last Sim	Action	Status

0x357f7b7579c823edf313ed4723429986fecb1f3facda3d782ed2fe42638083bb

Fail with error 'Seraph: Transaction not approved'

9473498 1 Block Confirmation

23 secs ago (Aug-06-2023 06:58:36 PM +UTC)

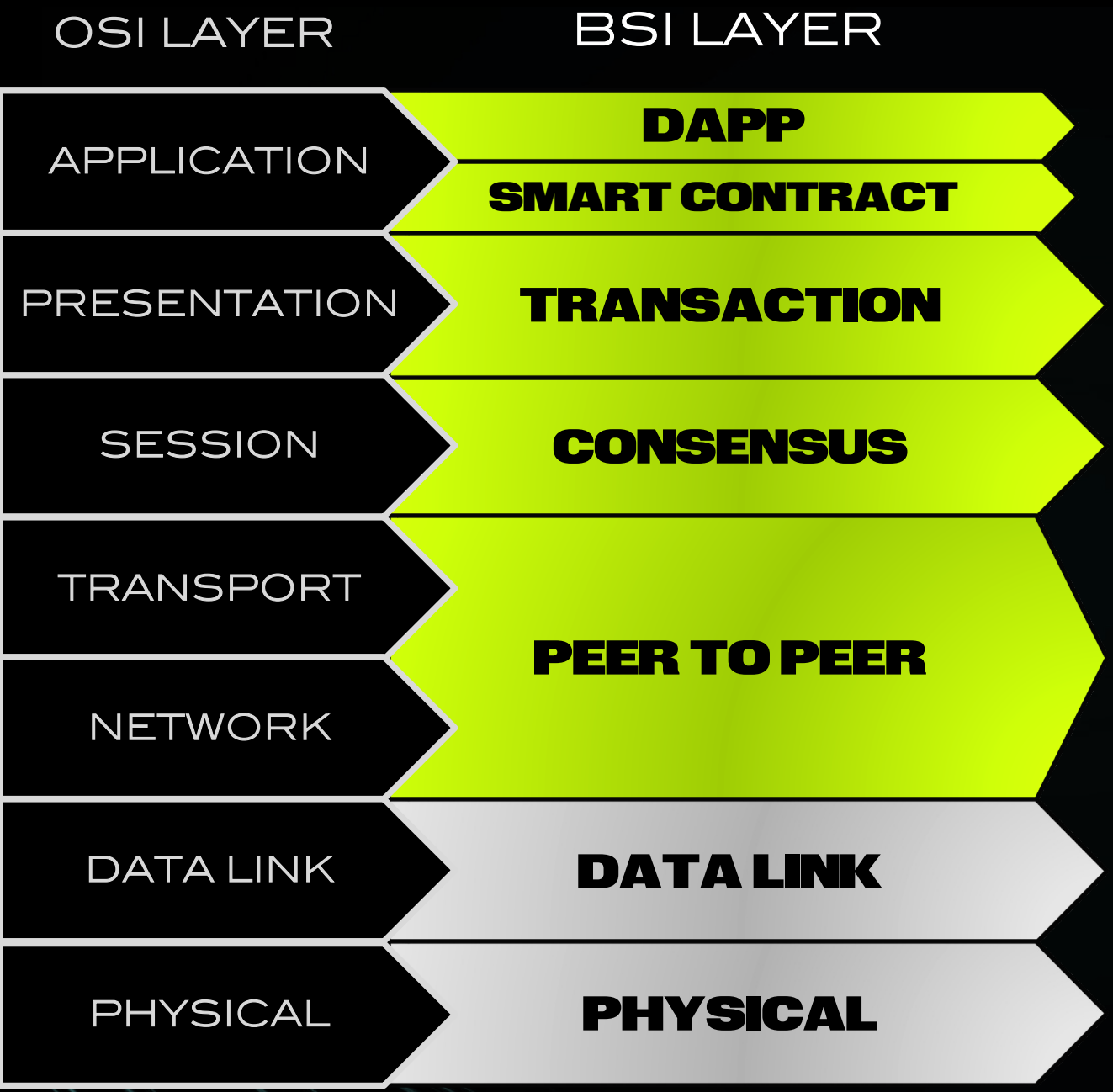
0x8a9Fd70bF449e62e89dc12C24AA9Fd99978BFabF

0x2D100984Bfe1796A2af81e32a44782b4A3129393

Warning! Error encountered during contract execution [**execution reverted**]

MALICIOUS FUNCTION CALL IS PREVENTED FROM EXECUTING

HALBORN



THE BLOCKCHAIN SYSTEMS INTERCONNECTION MODEL

BSI LAYER

- DAPP
- SMART CONTRACT
- TRANSACTION
- CONSENSUS
- PEER TO PEER
- DATA LINK
- PHYSICAL

THE BSI MODEL

A CONCEPTUAL MODEL THAT PROVIDES A STANDARD FROM OF REFERENCE FOR BLOCKCHAIN BASED DLT SYSTEMS

BSI LAYER

- DAPP
- SMART CONTRACT
- TRANSACTION
- CONSENSUS
- PEER TO PEER
- DATA LINK
- PHYSICAL

PHYSICAL + DATA LINK

THE SAME PHYSICAL MEDIUM STANDARDS AND INTERFACES FOR SENDING AND RECIEVING DATA

BSI LAYER

- DAPP
- SMART CONTRACT
- TRANSACTION
- CONSENSUS
- **PEER TO PEER**
- DATA LINK
- PHYSICAL

PEER TO PEER

Messages and data is broadcast on a distributed network of equally privileged nodes that partition tasks and workloads between its peers on the system.

BSI LAYER

- DAPP
- SMART CONTRACT
- TRANSACTION
- **CONSENSUS**
- PEER TO PEER
- DATA LINK
- PHYSICAL

CONSENSUS

THE METHOD USED TO ACHIEVE AGREEMENT, TRUST, AND SECURITY ON THE STATE OF THE NETWORK.

PROOF OF WORK

PROOF OF STAKE

BSI LAYER

- DAPP
- SMART CONTRACT
- **TRANSACTION**
- CONSENSUS
- PEER TO PEER
- DATA LINK
- PHYSICAL

TRANSACTION

THE CRYPTOGRAPHICLY SIGNED MESSAGE SENT TO THE NETWORK TO UPDATE OR CONFIRM DATA

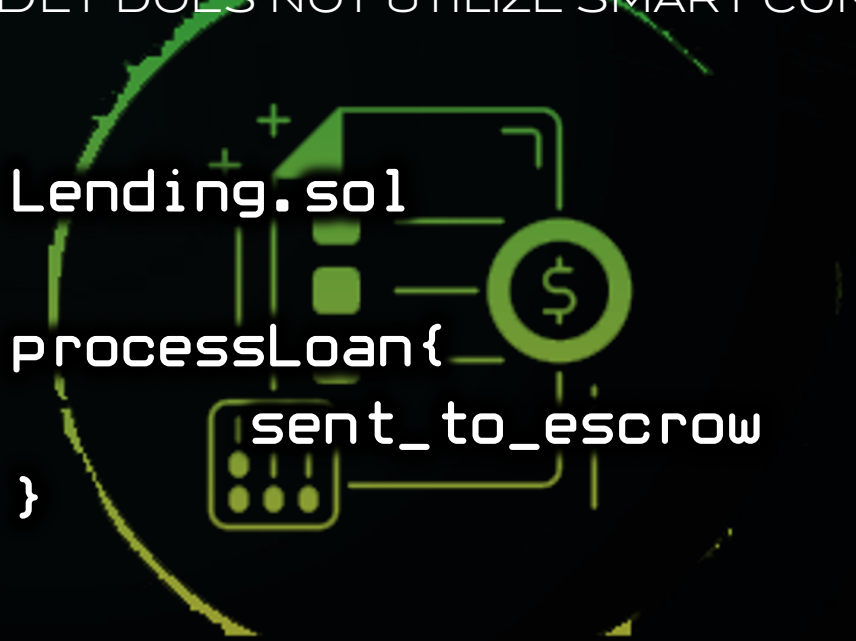
BSI LAYER

- DAPP
- SMART CONTRACT*
- TRANSACTION
- CONSENSUS
- PEER TO PEER
- DATA LINK
- PHYSICAL

SMART CONTRACT

SELF-EXECUTING CODE THAT RUN ACTIONS BASED ON PRE-DETERMINED CONDITIONS & LOGIC

*(SOME DLT DOES NOT UTILIZE SMART CONTRACTS)

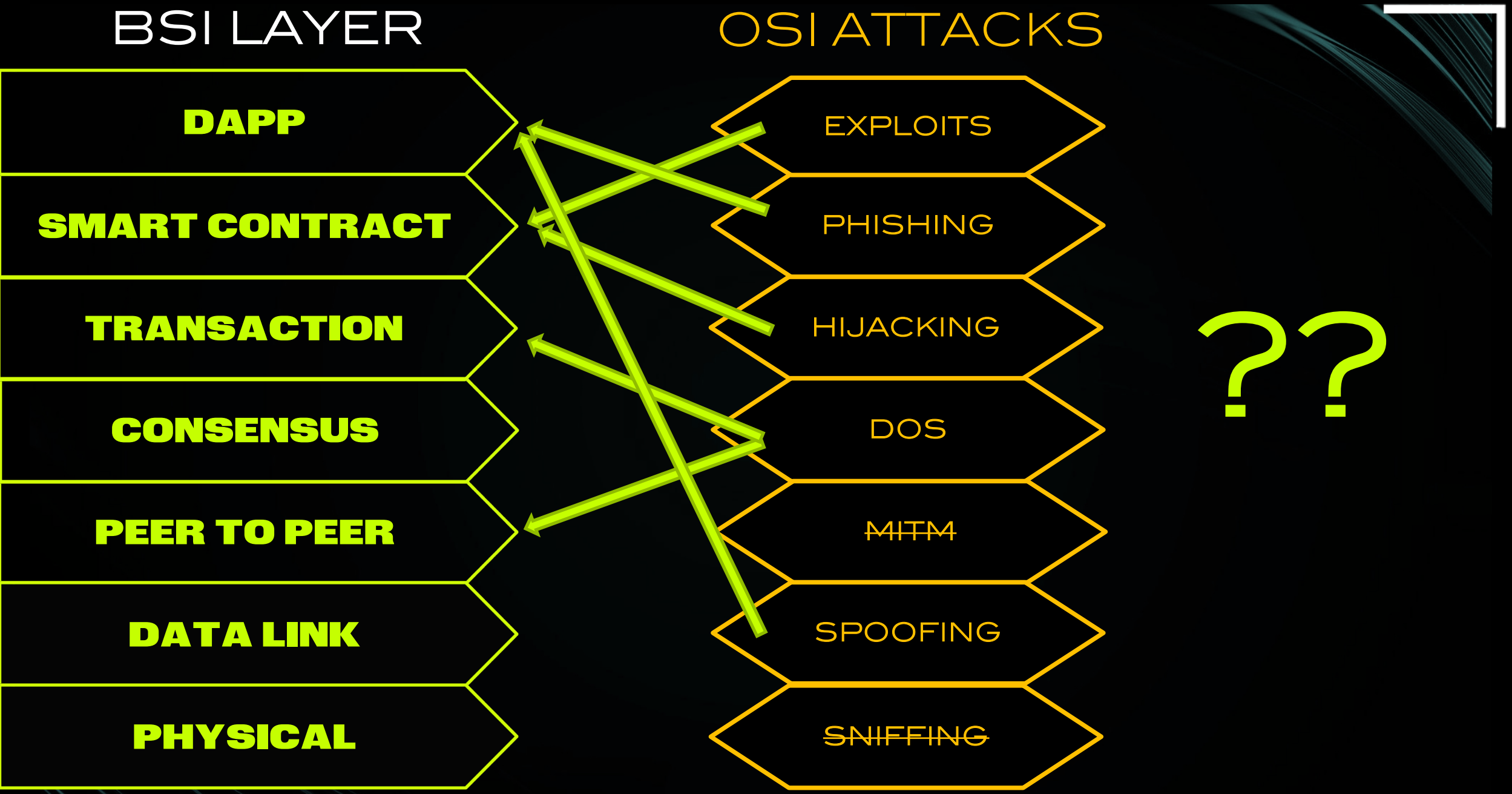


BSI LAYER

- **DAPP**
- **SMART CONTRACT**
- **TRANSACTION**
- **CONSENSUS**
- **PEER TO PEER**
- **DATA LINK**
- **PHYSICAL**

DAPP

DECENTRALIZED APPLICATIONS THAT ALLOW A USER TO INTERACT WITH BLOCKCHAIN BY USING A PRIVATE KEY


BSI LAYER
OSI ATTACKS
DAPP
SMART CONTRACT
TRANSACTION
CONSENSUS
PEER TO PEER
DATA LINK
PHYSICAL
EXPLOITS
PHISHING
HIJACKING
DOS
~~MITM~~
SPOOFING
~~SNIFFING~~
??

THE ATTACK SURFACE HAS CHANGED

??

WE NEED NEW WAYS TO QUANTIFY RISK

QUANTIFYING RISK EXAMPLES

RESOURCES SHOULD TO USED TO MITIGATE THE RIGHT RISKS

CONSENSUS RISK

- Distributed Systems are less likely to have availability outages.
- There are far less reasons for an attacker to compromise the data in transit. (i.e. on the Wire via sniffing/or MitM attacks)
- 51% / 34% attacks are more likely on systems with low participants, and can be expensive and cost-prohibitive to launch.

SMART CONTRACT RISK

- There are no “OWASP Top 10” vulnerabilities in Smart Contracts or DAPPs.
- We need tools that can identify new categories of vulnerabilities.
- Financial Logic via DeFi can now be attacked with far greater impact can ever before via environmental exploits.
- Code auditing pre-deployment is extremely important due to immutability of DLT.

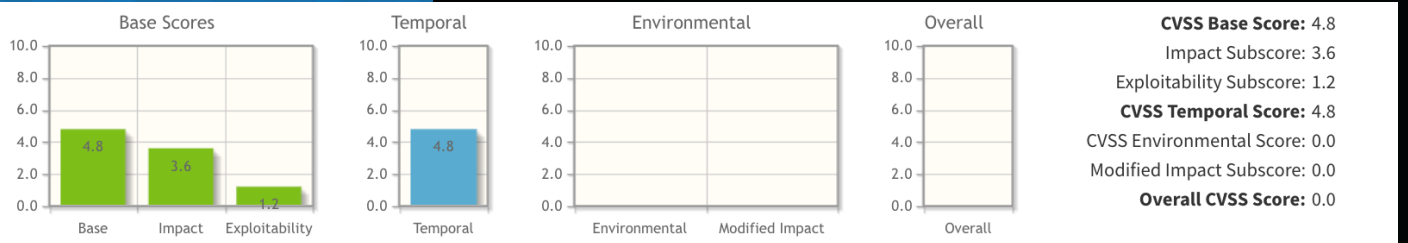
USER RISKS

- Self Custody of the Private Keys make end users and custodial platforms the primary target.
- Like with Phishing, misdirection is used very often due to the “open” nature of distributed networks.
- A private key if lost or forgotten is unrecoverable.
- DLT with a public shows all records and actions of every participant/account/wallet.

NIST - CVSS CALCULATOR

AN EXAMPLE OF HOW WE RE-INVENT RISK QUANTIFICATION

NATIONAL VULNERABILITY DATABASE NVD



CVSS Base Score: 4.8
Impact Subscore: 3.6
Exploitability Subscore: 1.2
CVSS Temporal Score: 4.8
CVSS Environmental Score: 0.0
Modified Impact Subscore: 0.0
Overall CVSS Score: 0.0

Show Equations

CVSS v3.1 Vector

AV:N/AC:H/PR:L/UI:R/S:U/C:H/I:N/A:N/E:X/RL:X/RC:C/CR:X/IR:X/AR:X/MAV:N/MAC:X/MPR:N/MUI:X/MS:C/MC:N/MI:X/MA:X

Base Score Metrics

Exploitability Metrics

Attack Vector (AV)*

Network (AV:N) | Adjacent Network (AV:A) | Local (AV:L) | Physical (AV:P)

Attack Complexity (AC)*

Low (AC:L) | High (AC:H)

Privileges Required (PR)*

None (PR:N) | Low (PR:L) | High (PR:H)

User Interaction (UI)*

None (UI:N) | Required (UI:R)

Scope (S)*

Unchanged (S:U) | Changed (S:C)

Impact Metrics

Confidentiality Impact (C)*

None (C:N) | Low (C:L) | High (C:H)

Integrity Impact (I)*

None (I:N) | Low (I:L) | High (I:H)

Availability Impact (A)*

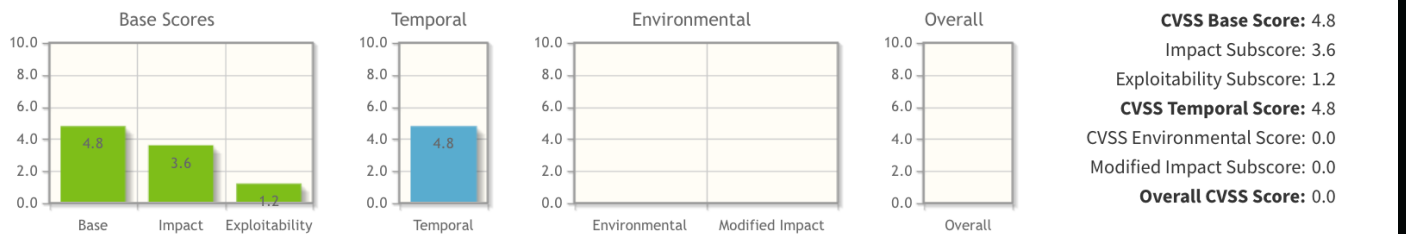
None (A:N) | Low (A:L) | High (A:H)

THE COMMON VULNERABILITY SCORING (CVSS) CALCULATOR IS A WIDELY USED TOOL FOR MEASURING THE CRITICALITY OF A VULNERABILITY.

NIST - CVSS CALCULATOR

AN EXAMPLE OF HOW WE RE-INVENT RISK QUANTIFICATION

NIST NATIONAL VULNERABILITY DATABASE NVD



CVSS Base Score: 4.8
Impact Subscore: 3.6
Exploitability Subscore: 1.2
CVSS Temporal Score: 4.8
CVSS Environmental Score: 0.0
Modified Impact Subscore: 0.0
Overall CVSS Score: 0.0

Show Equations

CVSS v3.1 Vector

AV:N/AC:H/PR:L/UI:R/S:U/C:H/I:N/A:N/E:X/RL:X/RC:C/CR:X/IR:X/AR:X/MAV:N/MAC:X/MPR:N/MUI:X/MS:C/MC:N/MI:X/MA:X

Base Score Metrics

Exploitability Metrics

Attack Vector (AV)*
Network (AV:N) | Adjacent Network (AV:A) | Local (AV:L) | Physical (AV:P)

Attack Complexity (AC)*
Low (AC:L) | High (AC:H)

Privileges Required (PR)*
None (PR:N) | Low (PR:L) | High (PR:H)

User Interaction (UI)*
None (UI:N) | Required (UI:R)

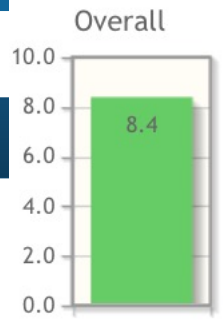
Scope (S)*
Unchanged (S:U) | Changed (S:C)

Impact Metrics

Confidentiality Impact (C)*
None (C:N) | Low (C:L) | High (C:H)

Integrity Impact (I)*
None (I:N) | Low (I:L) | High (I:H)

Availability Impact (A)*
None (A:N) | Low (A:L) | High (A:H)



CVSS Base Score: 8.3
Impact Subscore: 6.0
Exploitability Subscore: 1.6
CVSS Temporal Score: 8.3
CVSS Environmental Score: 8.4
Modified Impact Subscore: 6.1
Overall CVSS Score: 8.4

AV:N/AC:H/PR:N/UI:R/S:C/C:H/I:H/A:H/E:H/RL:U/RC:C/CR:H/IR:X/AR:H/MAV:N/MAC:X/MPR:N/MUI:X/MS:C/MC:H/MI:H/MA:H

AFTER ENTERING IN EXPLOITABILITY AND IMPACT VALUES, THE CVSS CALCULATOR WILL OUTPUT A VECTOR SCORE IN THE RANGE OF 1 TO 10, WITH 10 BEING A CRITICAL RISK

HALBORN - BVSS CALCULATOR

AN EXAMPLE OF HOW WE RE-INVENT RISK QUANTIFICATION

HALBORN

// BLOCKCHAIN VULNERABILITY SCORING SYSTEM

BVSS CALCULATOR

- **HALBORN HAS CREATED A BLOCKCHAIN VULNERABILITY SCORING SYSTEM TO CALCULATE RISK LEVEL FOR DEFI AND SMART CONTRACTS**
- **USED IN OUR SMART CONTRACT AUDITS AND ASSESSMENTS TO QUANTIFY RISK LEVEL**
- **INCLUDES IMPACT METRICS LIKE: DEPOSIT AMOUNTS and YIELD**
- **AND CONSIDERS OTHER FACTORS LIKE ATTACK COST (i.e. GAS REQUIREMENTS)**

IMPACT METRICS

Confidentiality

NONE	LOW	MEDIUM	HIGH	CRITICAL

No data is affected

Availability

NONE	LOW	MEDIUM	HIGH	CRITICAL

No features are affected

Integrity

NONE	LOW	MEDIUM	HIGH	CRITICAL

No data is affected

Deposit

NONE	LOW	MEDIUM	HIGH	CRITICAL

No funds are affected

Yield

NONE	LOW	MEDIUM	HIGH	CRITICAL

No yield is affected

EXPLOITABILITY METRICS

Attack Origin

ARBITRARY	SPECIFIC

Access to a privileged account is required

Attack Cost

LOW	MEDIUM	HIGH

The cost is comparable or greater to the benefits of triggering the bug

Attack Complexity

LOW	MEDIUM	HIGH

COEFFICIENTS

Reversibility

NONE	PARTIAL	FULL

Some consequences can be reversed or the cost of reversing is significant

Scope

UNCHANGED	CHANGED

Third-party systems or users are affected

HALBORN - BVSS CALCULATOR

A FREE TOOL IN OUR UPCOMING SOLUTIONS CENTER

H SECURITY SOLUTIONS CENTER BETA

// BLOCKCHAIN VULNERABILITY SCORING SYSTEM

BVSS CALCULATOR

OVERVIEW | **ENGAGEMENTS** | **ADVISORY**

// LAST UPDATED A MONTH AGO

RISK ASSESSMENT

- Storing private keys securely.
- Application Logic Flaws.
- Fuzzing of input parameters.
- Areas where insufficient validation allows for hostile input.
- Research into architecture and purpose.

EXPLOITABILITY METRICS

Attack Origin

ARBITRARY | SPECIFIC

Any account can trigger the bug

Attack Cost

LOW | MEDIUM | HIGH

The cost is comparable to sending a few transactions

Attack Complexity

LOW | MEDIUM | HIGH

No specific conditions are required or the required conditions are relatively common

COEFFICIENTS

Reversibility

NONE | PARTIAL | FULL

The consequences are irreversible

Scope

UNCHANGED | CHANGED

The impact is isolated to the affected system and its direct users

BVSS VECTOR

AO:A/AC:L/AX:M/R:N/S:P/C:L/A:C/I:L/D:L/Y:L - 8.38 - High

QUESTIONS?
COME TALK TO ME!

THANK YOU!

STEVEN WALBROEHL

CO-FOUNDER @ HALBORN