Towards Building Secure and Efficient Decentralized Systems

Dr. Yuzhe Tang

EECS, Syracuse University

Towards Building Secure & Efficient **Decentralized Systems**

Modern infrastructures evolves to be more open and decentralized.

- Financial ledgers: blockchains
- Web infrastructures: transparency logs
- Cloud computing: decentralized storage

Towards Building Secure & Efficient **Decentralized Systems**

Intention of decentralization & open-membership designs:

- Trustworthy & more accountable

Towards Building **Secure** & Efficient Decentralized Systems

But two consequences of decentralized & open systems:

- Larger attack surface …

Towards Building **Secure** & Efficient Decentralized Systems

But two consequences of decentralized & open systems:

- Larger attack surface …

How to understand & harden security?

Towards Building Secure & **Efficient** Decentralized Systems

But two consequences of decentralized & open systems:

- Larger attack surface …

How to understand & harden security?

- Higher unit cost for basic operations …

Towards Building Secure & **Efficient** Decentralized Systems

But two consequences of decentralized & open systems:

- Larger attack surface …

How to understand & harden security?

- Higher unit cost for basic operations …

How to optimize the application cost?

Example Projects Presented in this Talk

- How to understand & harden the security in emerging large-scale systems?
  - Securing blockchains under DoS vectors (CCS'21, NDSS'21, IMC'21)
- How to analyze & optimize perf./costs in security-centric large-scale systems?
  - Cost-optimizing DApps without losing security (FSE'21, Middleware'20, ICDE'19)

Talk Outline

Theme 1: Securing blockchains under DoS vectors

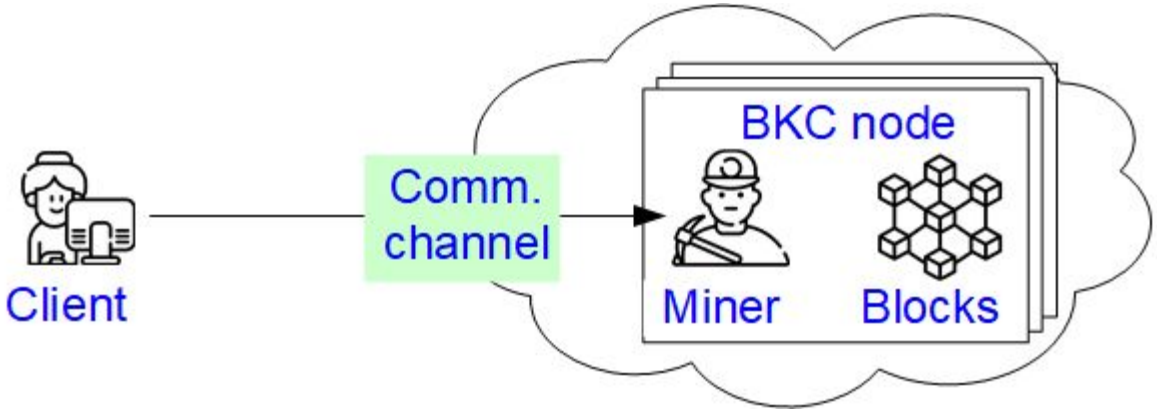
- RQ1: DoS security on TxRelay (published in NDSS'21)
- RQ2: DoS security on Tx propagation (ACM IMC'21)
- RQ3: DoS security on Mempool (ACM CCS'21)

Theme 2: Optimizing the DApp cost without losing security

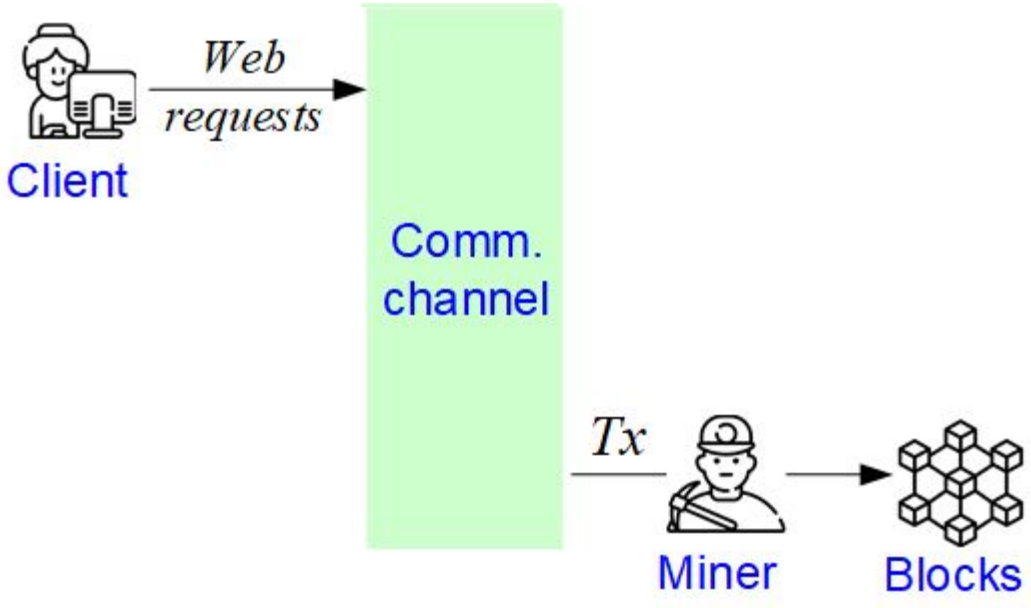
Overview of Other Research Themes

Future Research Directions

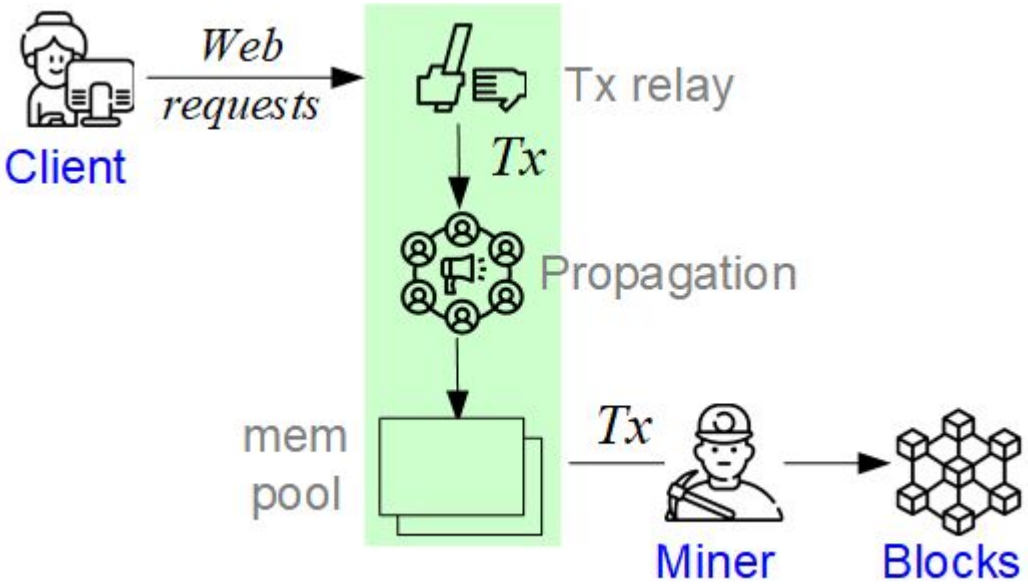
Blockchain Security under DoS



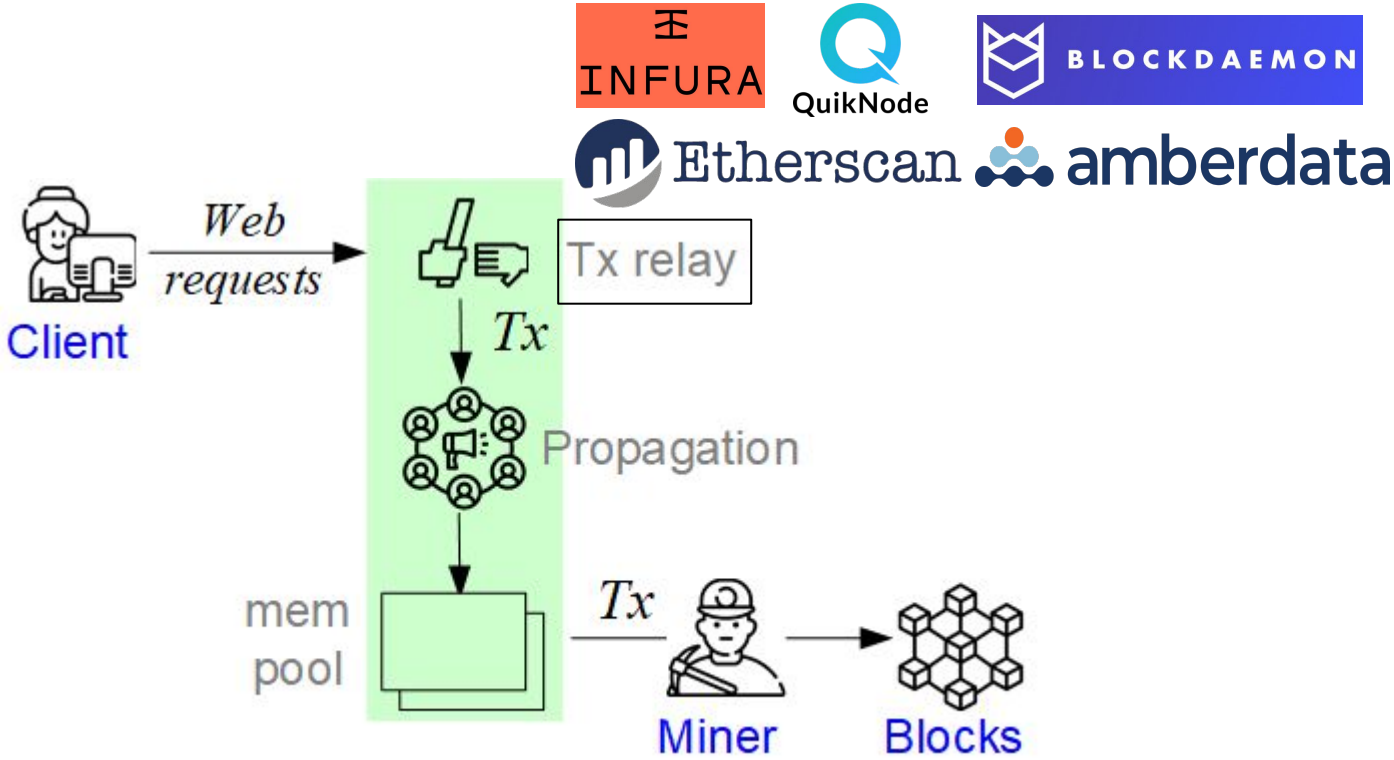
Blockchain Security under DoS



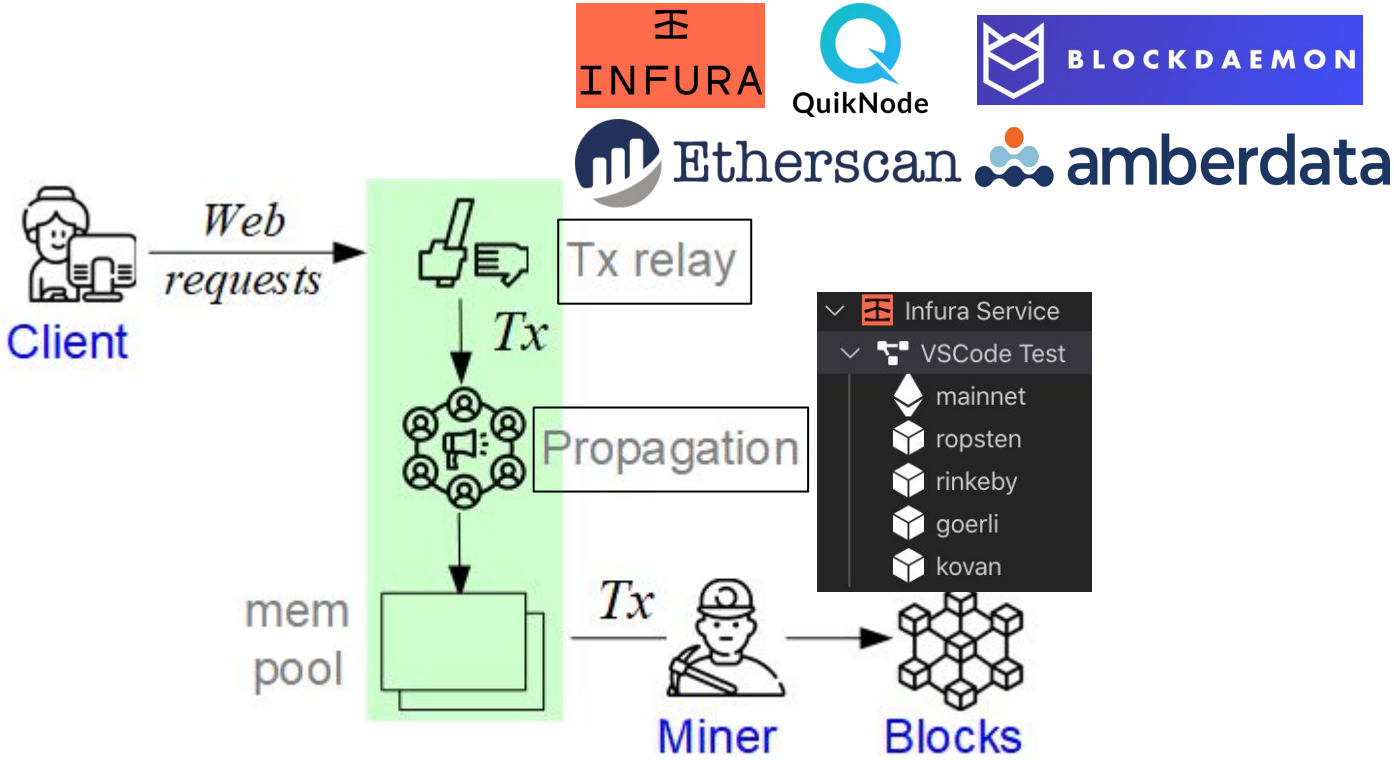
Blockchain Security under DoS



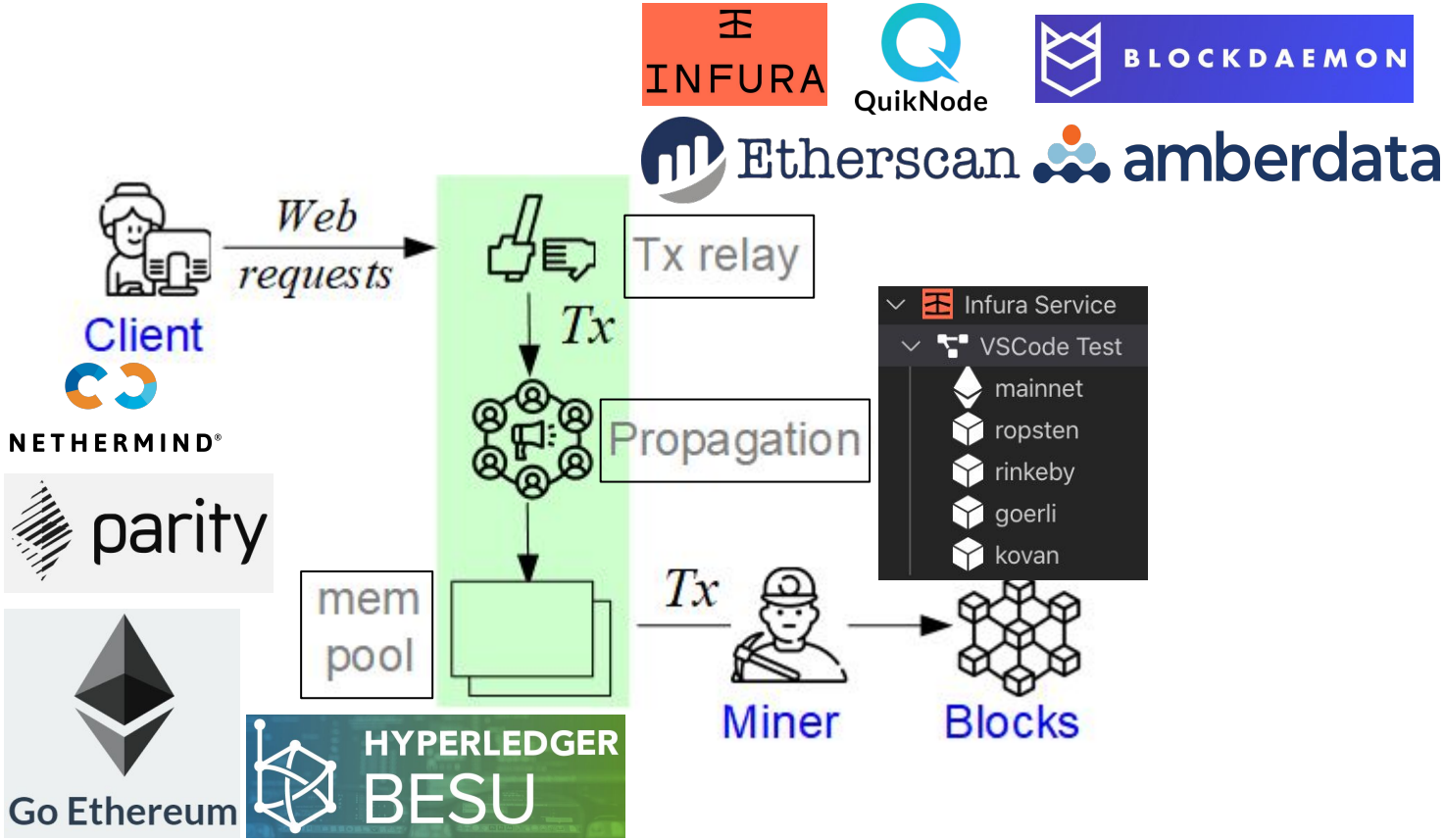
Blockchain Security under DoS



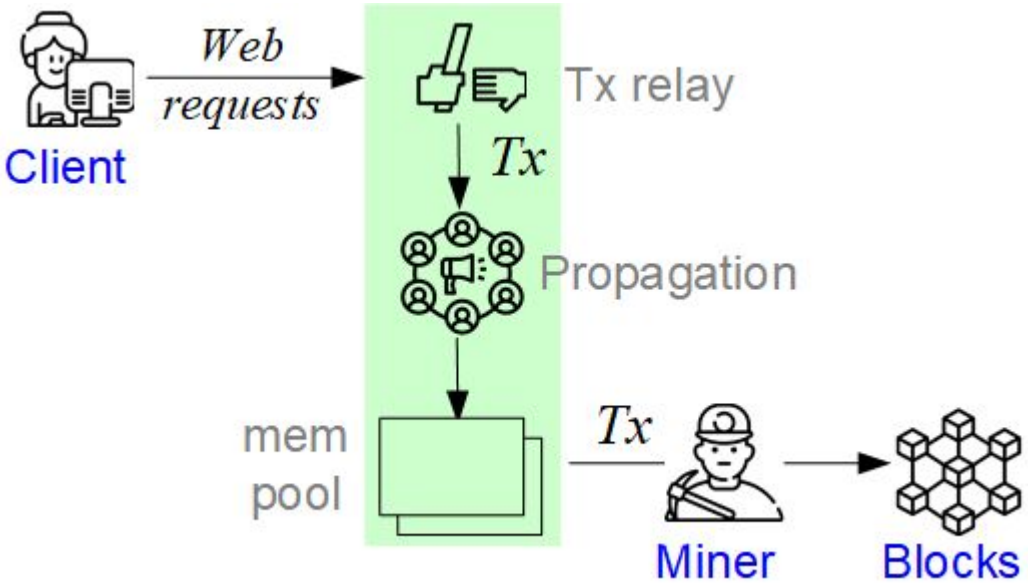
Blockchain Security under DoS



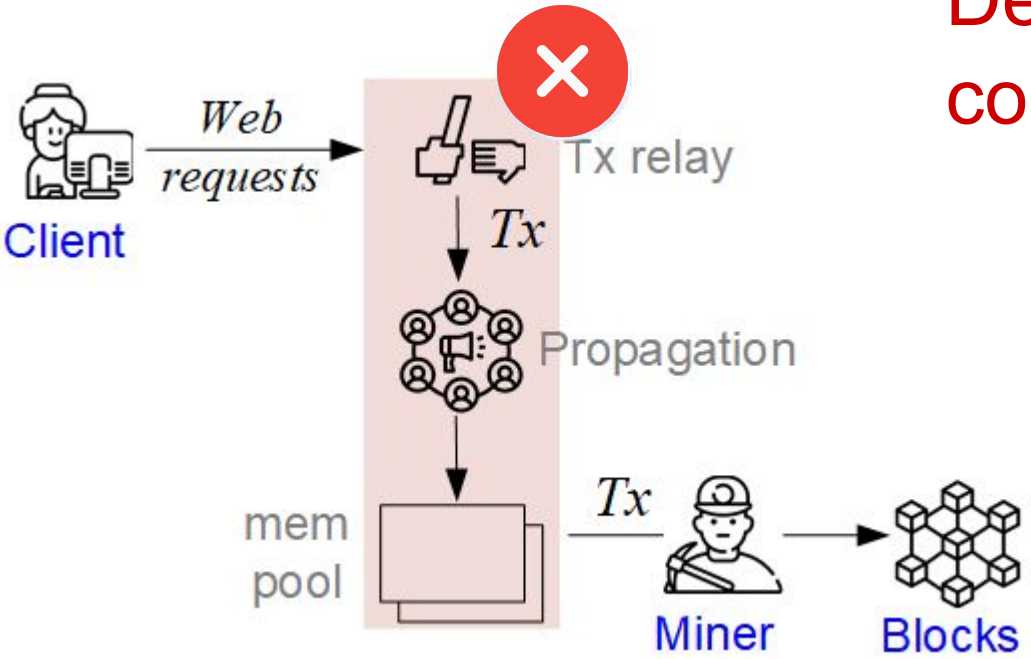
Blockchain Security under DoS



Blockchain Security under DoS

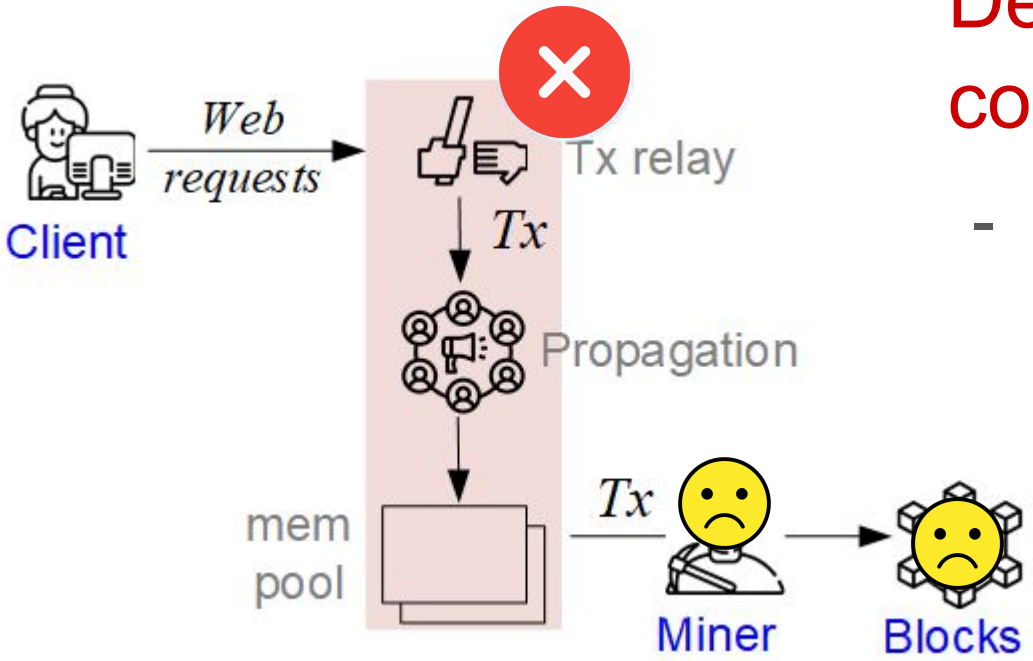


Blockchain Security under DoS



Denial of Blockchain comm. channel service?

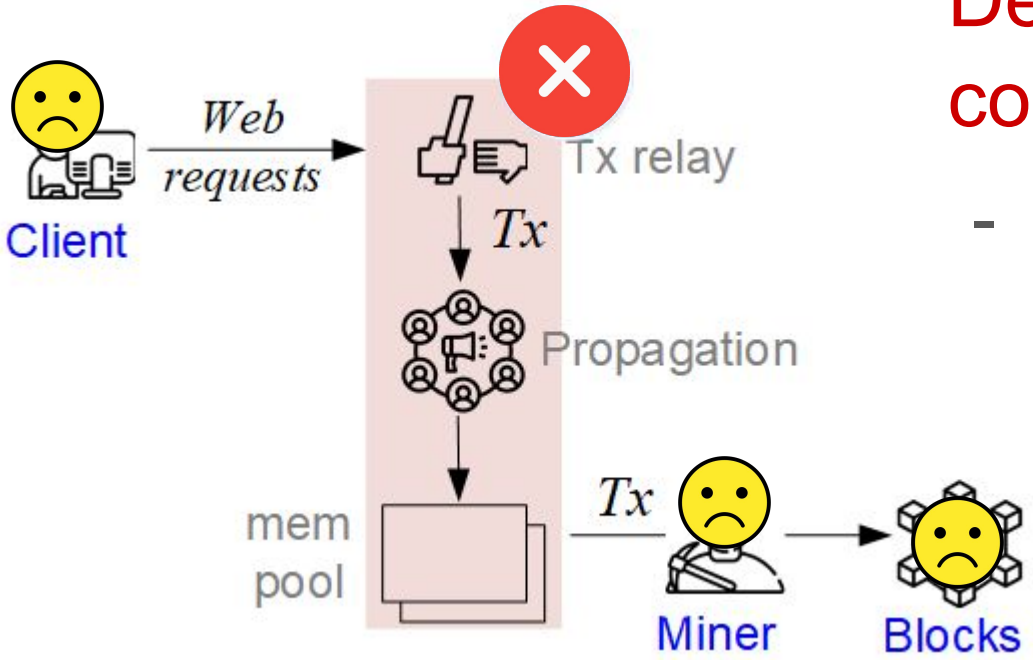
Blockchain Security under DoS



Denial of Blockchain comm. channel service?

- Miner unable to include txs; empty blocks.
  - low revenue, lose miners, 51% attacks

Blockchain Security under DoS



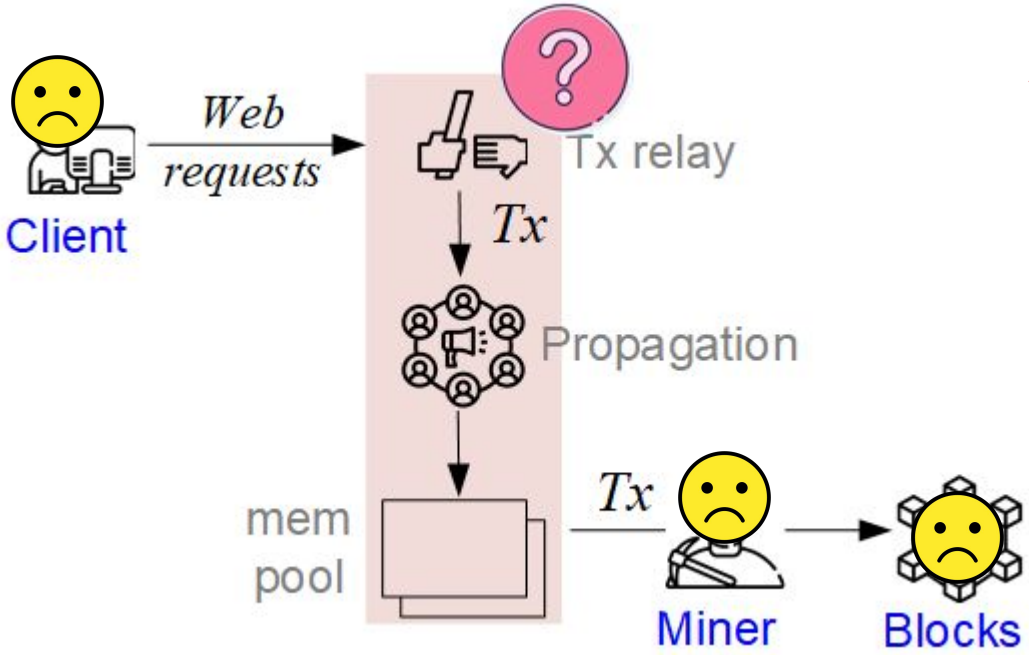
Denial of Blockchain comm. channel service?

- Miner unable to include txs; empty blocks.
  - low revenue, lose miners, 51% attacks

Clients cannot send txs.

- Frontrunning, lose clients

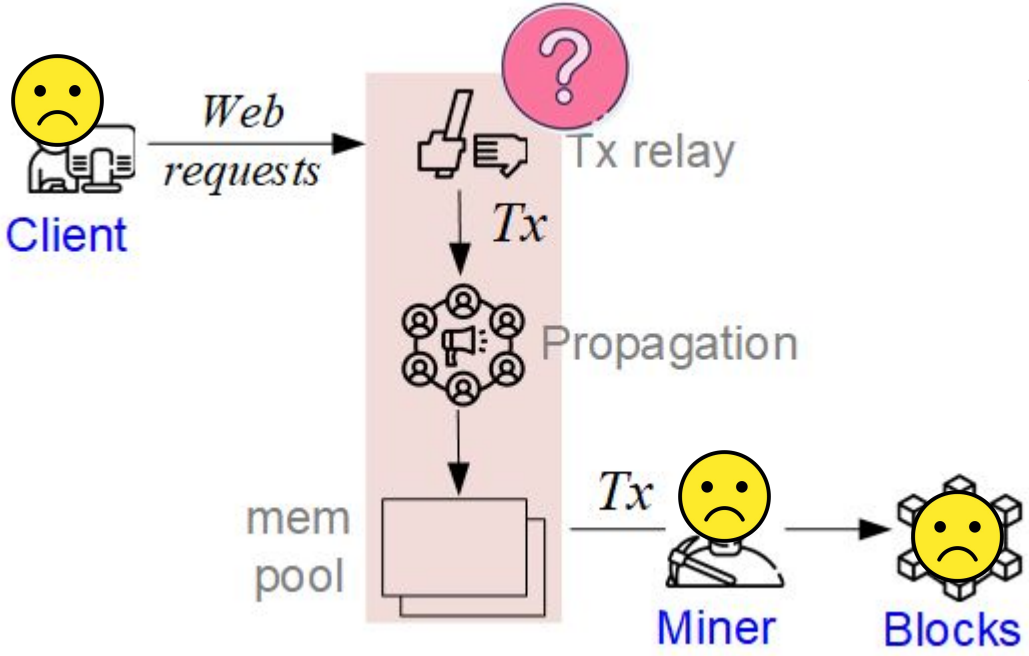
Blockchain Security under DoS



Research statement:

Whether & how resilient are Ethereum blockchains against denial of comm. channel service?

Blockchain Security under DoS

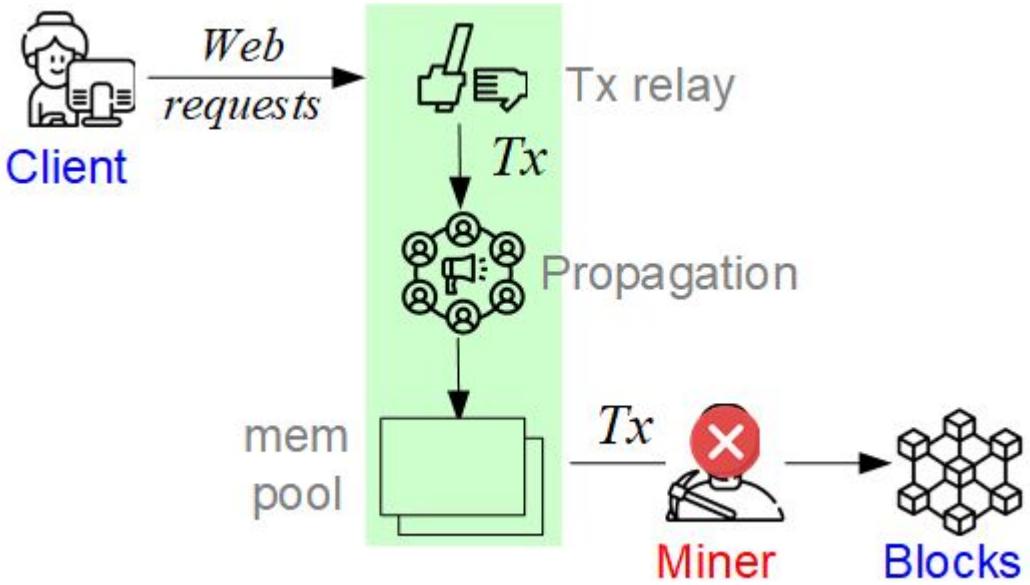


Research statement:

Whether & how resilient are Ethereum blockchains against denial of comm. channel service?

Rank	Name	Symbol	Market Cap	Price
1	Bitcoin	BTC	$1,027,956,378,947	$54,567.67
2	Ethereum	ETH	$428,418,048,937	$3,635.44

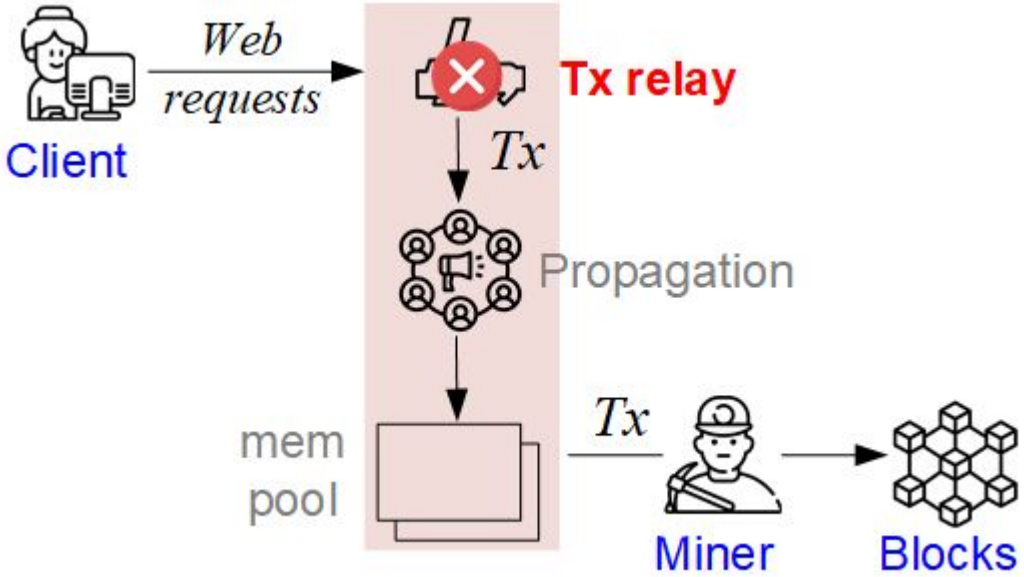
Blockchain Security under DoS: Related Works



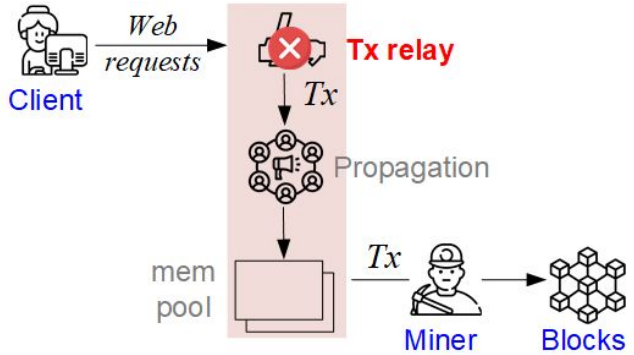
Existing works

- BDoS (CCS'20), selfish mining (FC'14), 51% attacks
- Smart-contract DoS (NDSS'20), Bribery (SP'21)

RQ1. Blockchain Security under DoS Tx Relay

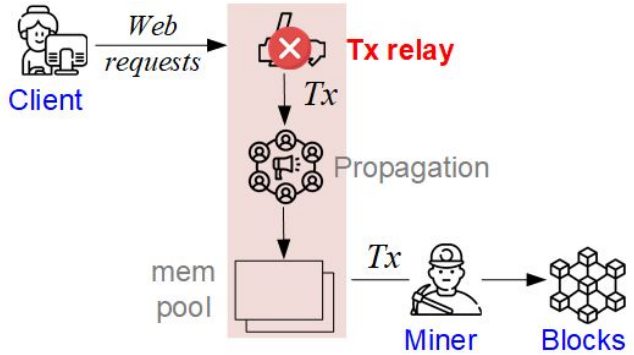


RQ1. Security under DoS Tx Relay



Formulated problem:

- Observe a vulnerable relay API (eth_call)
- If exposed, straightforward DoS exploiting eth_call



RQ1. Security under DoS Tx Relay

Formulated problem:

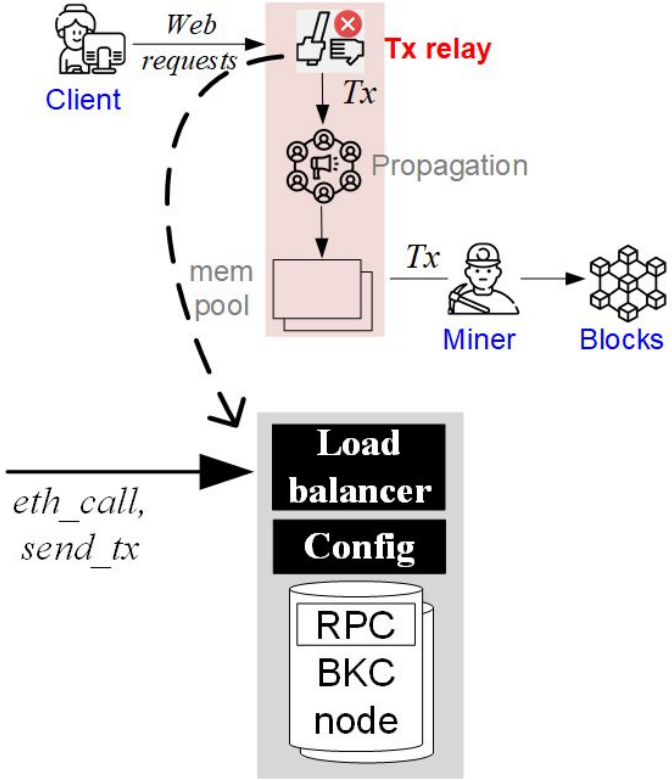
- Observe a vulnerable relay API (eth_call)
- If exposed, straightforward DoS exploiting eth_call

RQ1 (Exploitability measurement): Under the DoS exploiting the vulnerable API, how exploitable are real-world blackbox relay services are?

RQ1. Security under DoS Tx Relay

Proposed method (intuition)

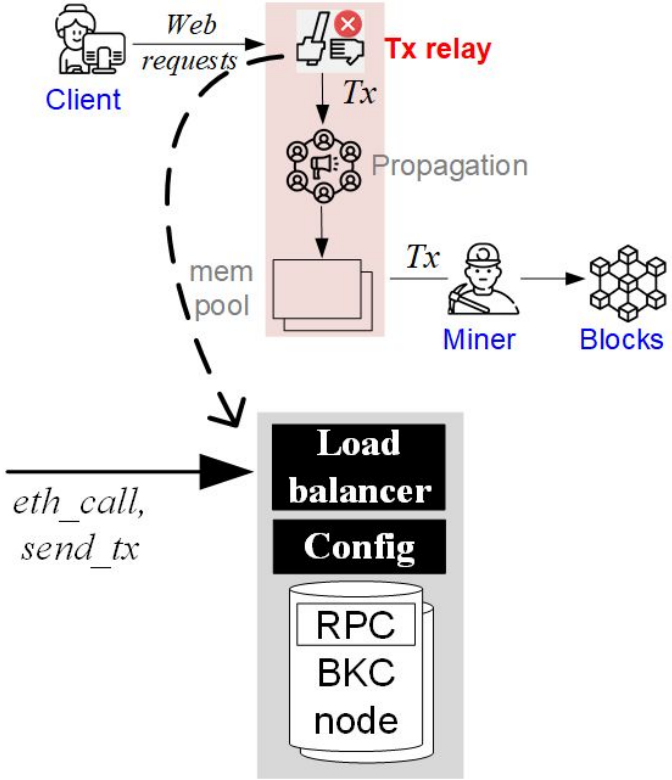
- Detect presence of load balancing inside tx relay services.



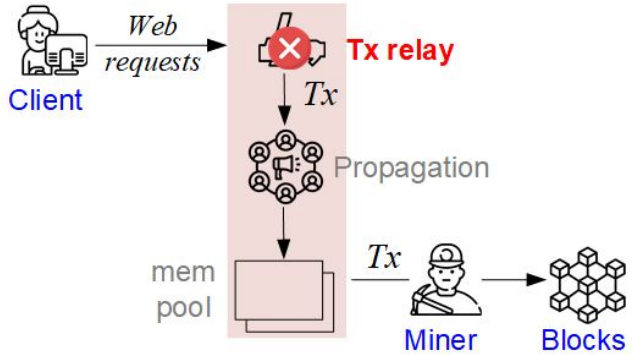
RQ1. Security under DoS Tx Relay

Proposed method (intuition)

- Detect presence of load balancing inside tx relay services.
- Send two double-spending txs to a RPC service & observe if both requests succeed.
  - Both requests succeed ⇒ load balancing detected.
  - One request fail ⇒ No load balancing detected.

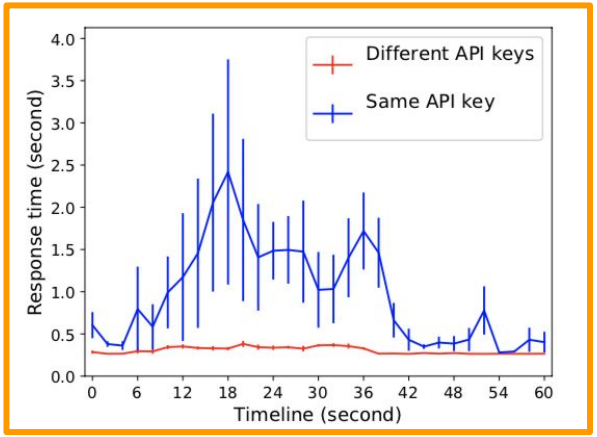


RQ1. Security under DoS Tx Relay



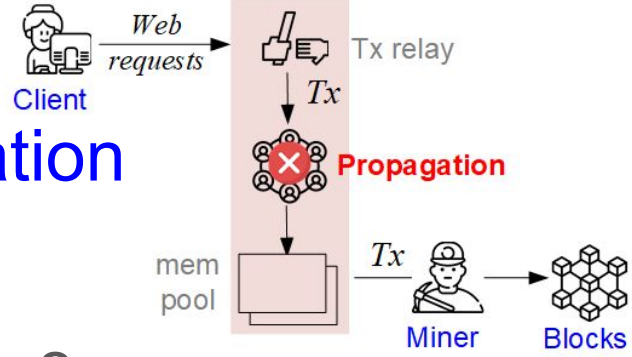
Results: on mainnet services

Type	RPC services	1IP-1key (LB0)	1IP-2key (LB1)	2IP-1key (LB2)	Gas limit
i	ServiceX1	✗	✗	✗	✗
	ServiceX2	✗	✗	✗	✗
	ServiceX3	✗	✗	✗	50
ii	ServiceX4	✗	✓	✗	✗
	ServiceX5	✗	✗	✓	✗
iii	ServiceX6	✓	✓	✓	10
	ServiceX7	✓	✓	✓	✗
	ServiceX9	✓	✓	✓	5
	ServiceX8	✓	✓	✓	1.5



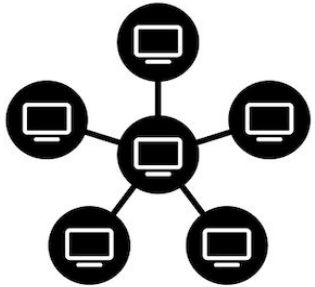
Publication: NDSS 2021

"As Strong As the Weakest Link: How to Break and Fix Blockchain DApps at RPC Service?"
Kai Li, Jiaqi Chen, Xianghong Liu, Yuzhe Tang, XiaoFeng Wang, Xiapu Luo.

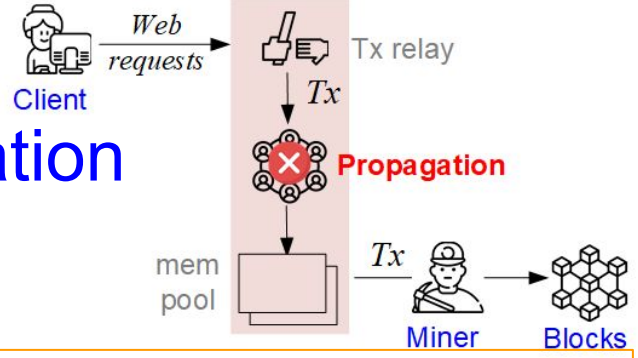


RQ2. Security under DoS Tx Propagation

Motivation: How resilient is Ethereum's Tx Propagation under single-point-of-failure?



- Single-point-of-failure by existing single-node attacks (e.g., eclipse attacks, DoERS, DETER)

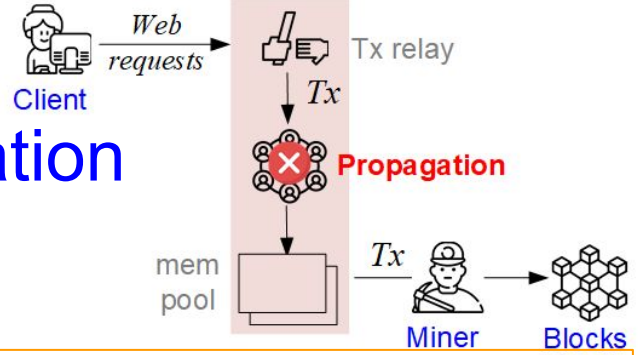


RQ2. Security under DoS Tx Propagation

Formulated problem:

RQ2 (Network measurement): What's Ethereum's network topology?

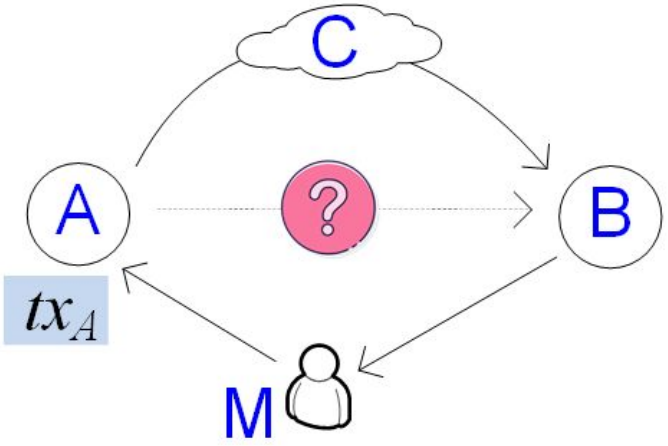
Related works	Blockchain	Measurement Target
[Neudecker et al. TR'18]	Bitcoin	Nodes
*[Miller et al. TR'**15**], [IEEE ATC'16], TxProbe [FC'19]*	Bitcoin	Edges
[FC'20]	Monero	Edges
[IMC'18], [FC'21]	Ethereum	Nodes
?	**Ethereum**	**Edges**

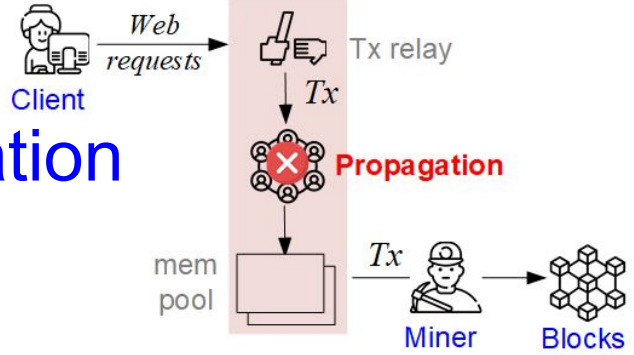


RQ2. Security under DoS Tx Propagation

Formulated problem:

RQ2 (Network measurement):
How to measure if remote Ethereum nodes (A & B) are connected?

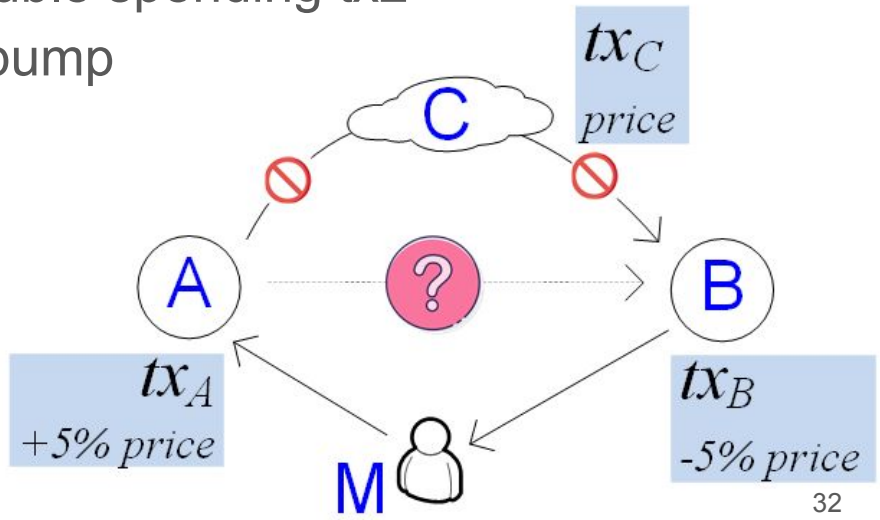


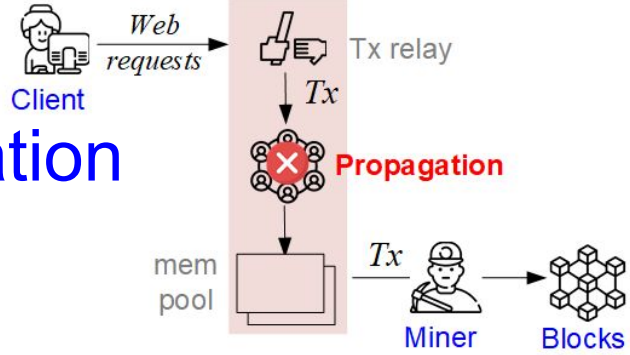


RQ2. Security under DoS Tx Propagation

Proposed method (intuition)

- Preliminary: tx replacement policies
  - Old tx1 replaced by a newer, double spending tx2 if tx2 has sufficient (10%) price bump
- Key insight:
  - Price bump & future txs to enforce isolation.

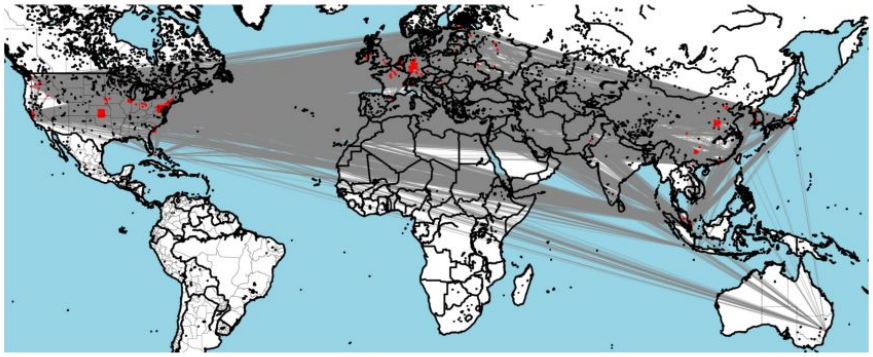




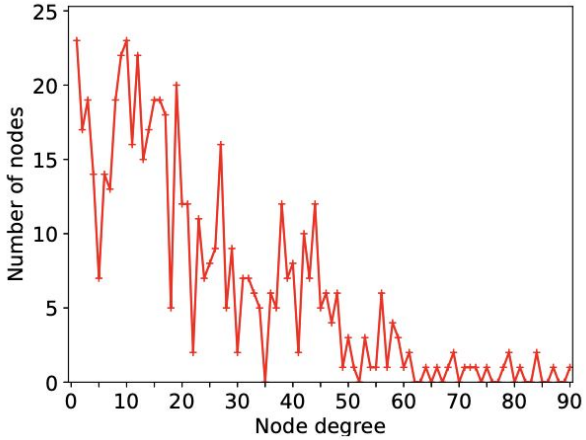
RQ2. Security under DoS Tx Propagation

Results: Full-network topology in testnets

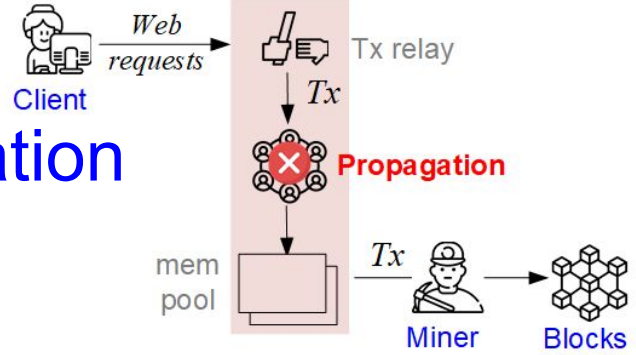
- Lower modularity & fewer cliques than random graphs
- Resilient to single-point-of-failure, but unsecure for low-degree nodes



(b) Geo distribution of Rinkeby



RQ2. Security under DoS Tx Propagation



Results: Critical-node subnet in mainnet

- Biased node connections towards popular services

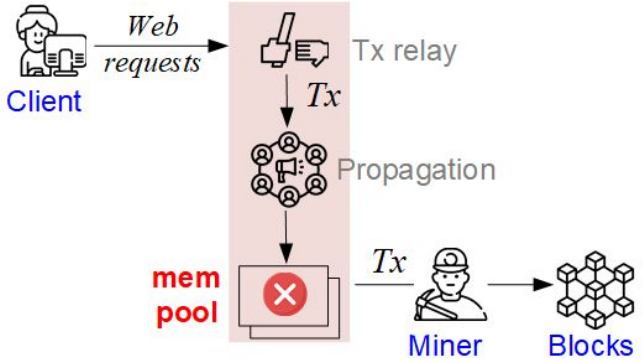
- Centralization leads to risks

Table 6: Connections among critical nodes

Type	Conn.	Type	Conn.
SrvR1- SrvM1	✓	SrvM1- SrvM1	✗
SrvR1- SrvM2	✓	SrvM1- SrvM2	✓
SrvR1- SrvM3	✓	SrvM1- SrvM4	✓
SrvR1- SrvM4	✓	SrvM1- SrvM3	✓
SrvR2- SrvM1	✗	SrvM2- SrvM2	✓
SrvR2- SrvM2	✗	SrvM2- SrvM3	✓
SrvR2- SrvM3	✗	SrvM2- SrvM4	✓
SrvR2- SrvM4	✗	SrvM3- SrvM4	✓
SrvR2- SrvR1	✗	SrvR1- SrvR1	✓

Publication: **ACM IMC 2021**

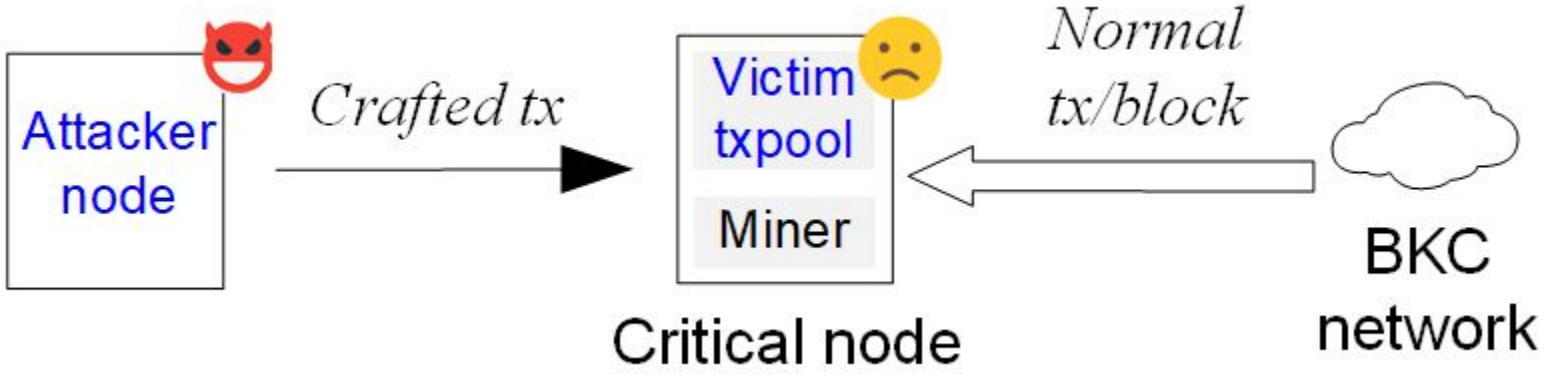
TopoShot: Uncovering Ethereum's Network Topology Leveraging Replacement Transactions.
Kai Li, Yuzhe Tang, Jiaqi Chen, Yibo Wang, Xianghong Liu



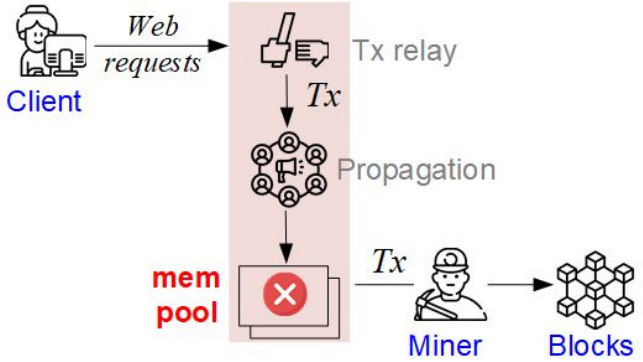
RQ3. Security under DoS Mempool

Formulated problem:

RQ3 (Attack design): Whether possible and how to spam a remote mempool at low cost?



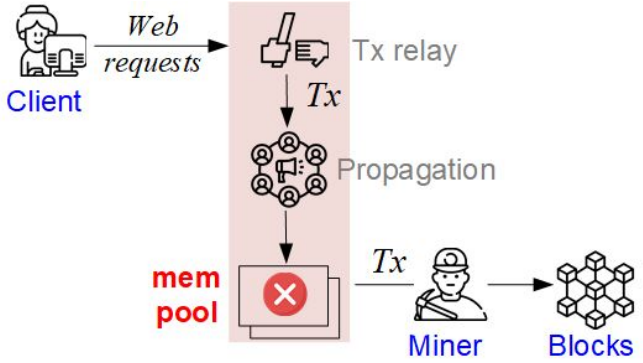
RQ3. Security under DoS Mempool



Proposed method (intuition):

- Exploit the design flaw in auction-based tx admission.
  - Ethereum uses auction to determine tx admission priority.
    - Necessary to mitigate spamming (Bitcoin16)
  - Protocol level: Assume all txs are profitable…
  - Implementation level: False assumption!
    - Unconfirmed Ethereum txs are invalid and unprofitable.
- Idea: Send unprofitable and high-priced txs to occupy an Ethereum node's mempool.

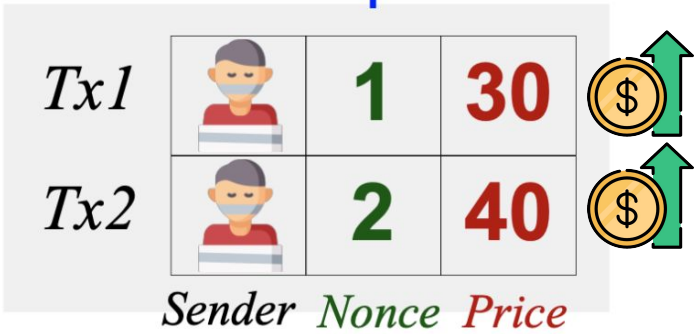
RQ3. Security under DoS Mempool



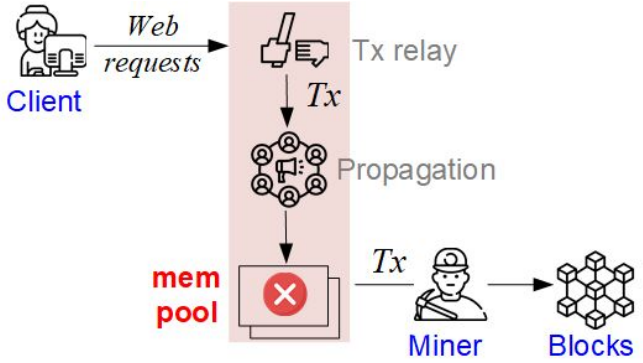
Proposed method (intuition): Attack

Attacker node

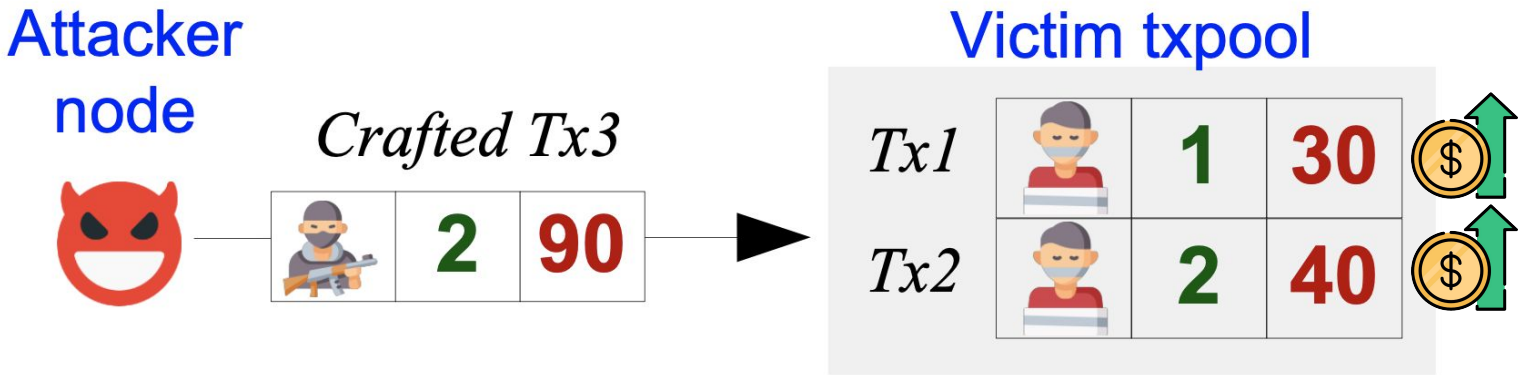
Victim txpool



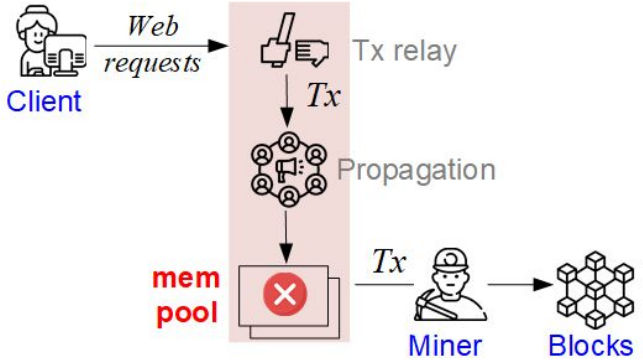
RQ3. Security under DoS Mempool



Proposed method (intuition): Attack



RQ3. Security under DoS Mempool



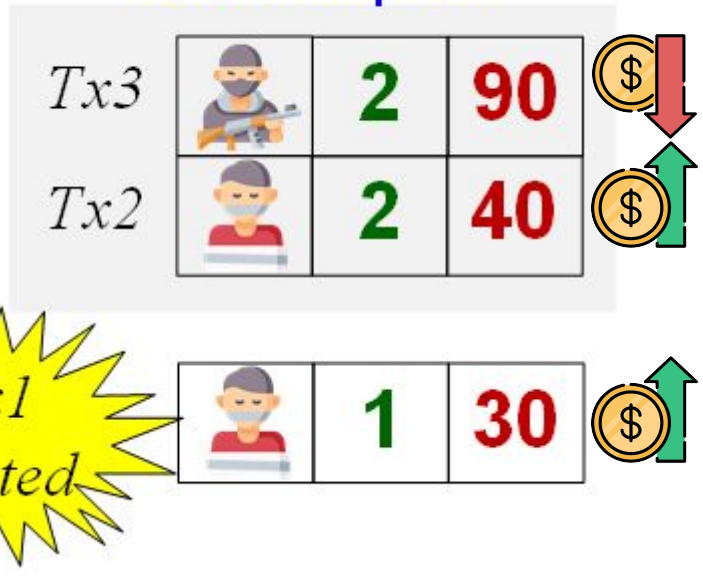
Proposed method (intuition): Attack

Attacker node

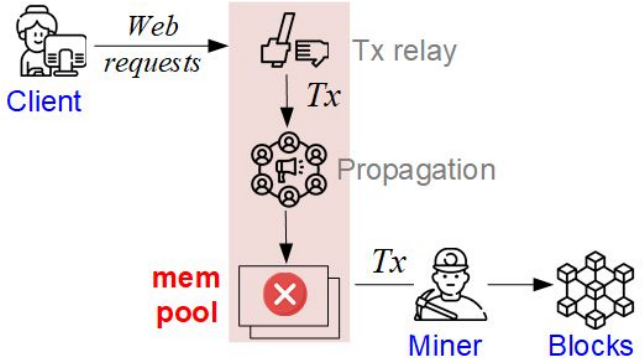
admitting tx3 leads to

1. evict tx1

Victim txpool



RQ3. Security under DoS Mempool



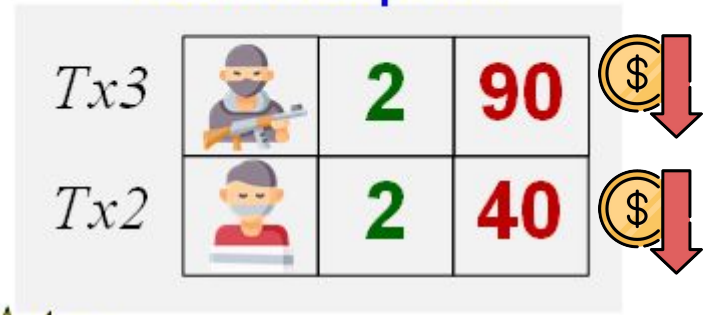
Proposed method (intuition): Attack

Attacker node

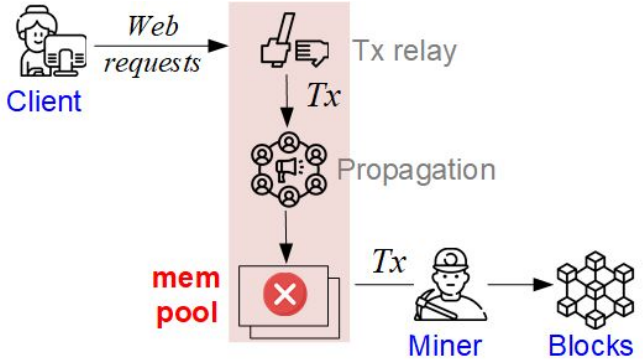
admitting tx3 leads to

1. evict tx1
2. turn tx2 to future

Victim txpool

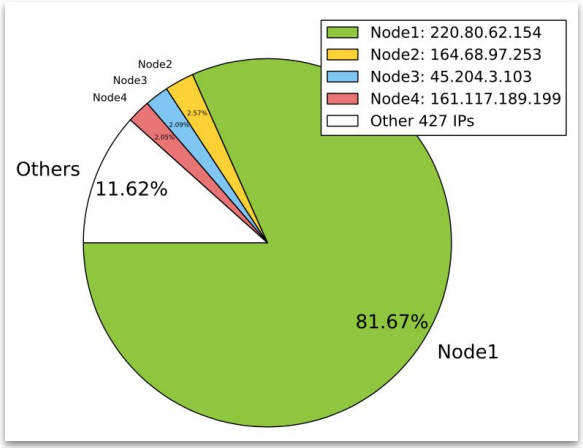


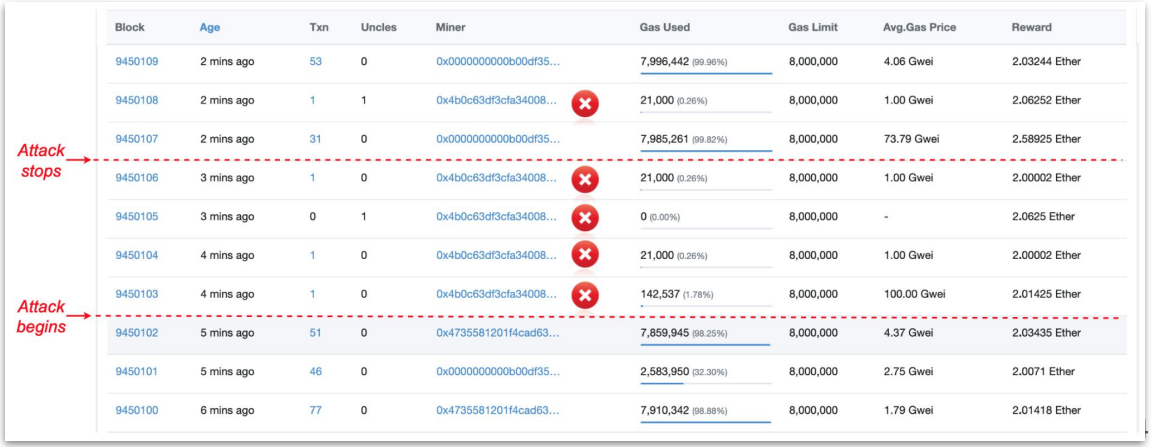
RQ3. Security under DoS Mempool



Results: Attack success & cost in testnets

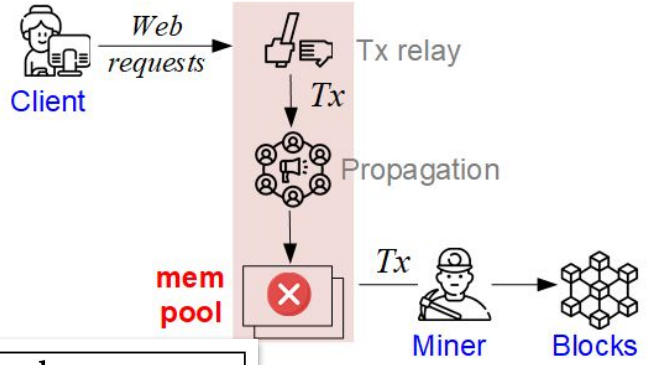
- Launching two supernodes joining Ropsten testnets.
- Using the node discoverability method to discover top miners.





Block	Age	Txn	Uncles	Miner	Gas Used	Gas Limit	Avg.Gas Price	Reward
9450109	2 mins ago	53	0	0x0000000000b00df35...	7,996,442 (99.96%)	8,000,000	4.06 Gwei	2.03244 Ether
9450108	2 mins ago	1	1	0x4b0c63df3cfa34008...	21,000 (0.26%)	8,000,000	1.00 Gwei	2.06252 Ether
9450107	2 mins ago	31	0	0x0000000000b00df35...	7,985,261 (99.82%)	8,000,000	73.79 Gwei	2.58925 Ether
9450106	3 mins ago	1	0	0x4b0c63df3cfa34008...	21,000 (0.26%)	8,000,000	1.00 Gwei	2.00002 Ether
9450105	3 mins ago	0	1	0x4b0c63df3cfa34008...	0 (0.00%)	8,000,000	-	2.0625 Ether
9450104	4 mins ago	1	0	0x4b0c63df3cfa34008...	21,000 (0.26%)	8,000,000	1.00 Gwei	2.00002 Ether
9450103	4 mins ago	1	0	0x4b0c63df3cfa34008...	142,537 (1.78%)	8,000,000	100.00 Gwei	2.01425 Ether
9450102	5 mins ago	51	0	0x4735581201f4cad63...	7,859,945 (98.25%)	8,000,000	4.37 Gwei	2.03435 Ether
9450101	5 mins ago	46	0	0x0000000000b00df35...	2,583,950 (32.30%)	8,000,000	2.75 Gwei	2.0071 Ether
9450100	6 mins ago	77	0	0x4735581201f4cad63...	7,910,342 (98.88%)	8,000,000	1.79 Gwei	2.01418 Ether

RQ3. Security under DoS Mempool

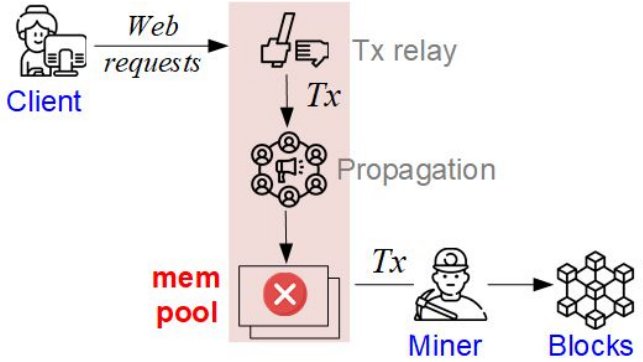


Results: Exploitability probes in mainnet

Service name	# of nodes	t_{1m}/X	t_{2m}/Z	Client-codename
Mining pools				
SrvM1	59	✓	✓	Geth-*turbo*
SrvM2	8	✓	✓	Geth-*ethereumsolo*, Geth-*ethereumppIns*
SrvM3	6	✓	✓	Geth-XX
SrvM4	2	✓	✓	Geth-XX
RPC services				
SrvR1	48	✓	✓	Geth-*omnibus*
SrvR2	1	✓	✓	Geth-*ethshared*

Publication: **ACM CCS 2021**

DETER: Denial of Ethereum's Txpool Service.
Kai Li, Yibo Wang, Yuzhe Tang



RQ3. Security under DoS Mempool

Mitigation scheme:

- Goal: DETER security versus miner revenue.
- Ideal: Decline any unprofitable txs.
- But profitability cannot be known upon admission?
- Heuristics: decline future txs (M0), decline exploitable tx eviction (M1).
- Evaluation: M0/M1 impl.'ed as middleware on mempool

Schemes	Miners' revenue (Ether)	DETER security	
		t_1/X	t_2/Z
Geth (default)	16.5388	✓/✗ (Table 2)	
M_0 (in Appendix 14.1)	15.9506(−3.56%)	✗	✗
M_1	16.5423(+0.002%)	✗	✗

Bug Reporting

- DETER/DoERS bugs confirmed by Ethereum client developers, RPC services & mining pools.

Bug Reporting

- DETER/DoERS bugs confirmed by Ethereum client developers, RPC services & mining pools.
- Bug bounty rewarded: >$22,000

Bug Reporting

- DETER/DoERS bugs confirmed by Ethereum client developers, RPC services & mining pools.
- Bug bounty rewarded: >$22,000
  - Acknowledgements https://bounty.ethereum.org/

Bug Reporting

- DETER/DoERS bugs confirmed by Ethereum client developers, RPC services & mining pools.
- Bug bounty rewarded: >$22,000
  - Acknowledgements https://bounty.ethereum.org/
- Quick code fix deployed, and advanced fixes in progress.

Talk Outline

Theme 1: Securing blockchains under DoS vectors

Theme 2: Optimizing the DApp cost without losing security

- RQ4: Reducing data movement costs
  - By batched processing (FSE'21)
  - By workload-aware data placement (Middleware'20)

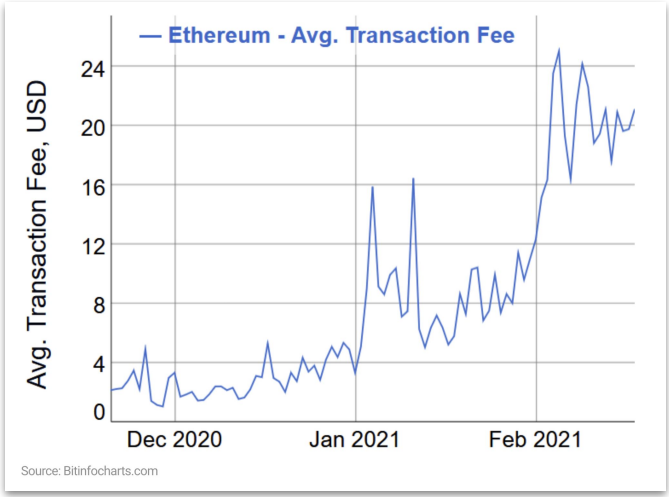
Overview of Other Research Themes

Future research directions

Theme 2[3M]: DApp Cost Efficiency

Observation: Fees are skyrocketing!

- Fees set by a market/auction (FPA)
- High fees as a result of economics:
  - More demand than supply.

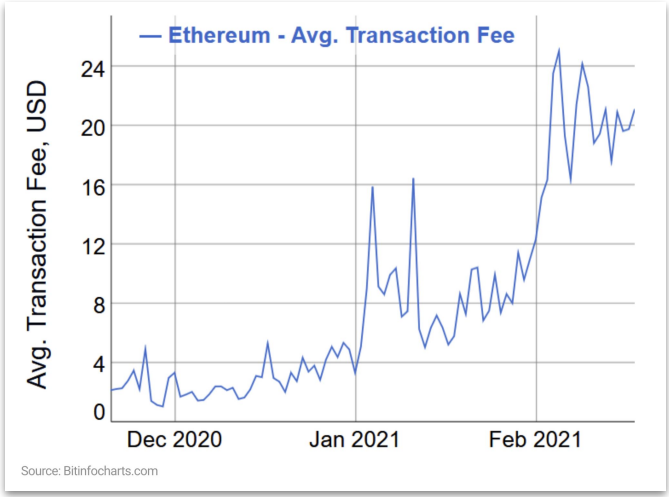


Theme 2^{3M}: DApp Cost Efficiency

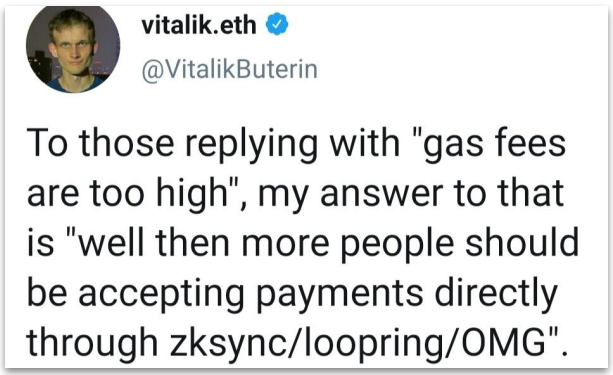
Observation: Fees are skyrocketing!

- Fees set by a market/auction (FPA)
- High fees as a result of economics:
  - More demand than supply.

Consequence: Scared away customers



Too Costly Ethereum is Pushing DeFi Users Away, Fuelling BNB Rally



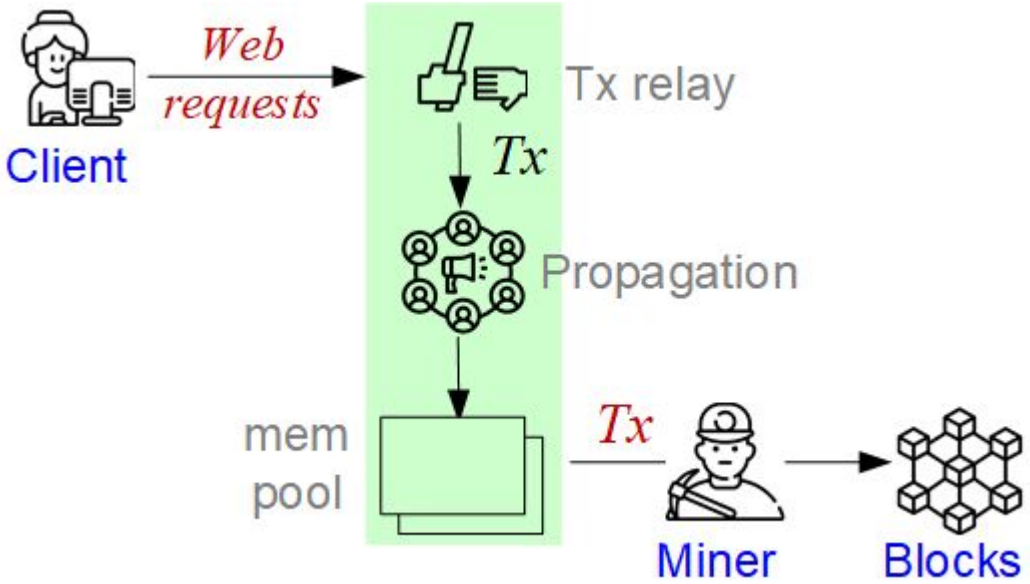
Theme 2^{3M}: DApp Cost Efficiency

Goal: Make DApps'/smart-contracts' use of blockchain efficient, without modifying underlying blockchains.

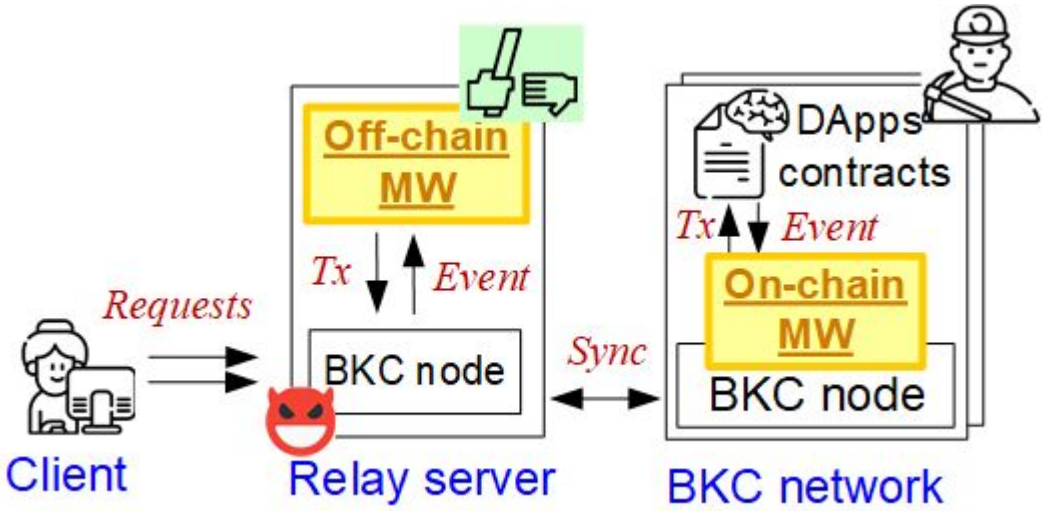
- Ease of deployment as a middleware onto operational blockchain.
- Distinct from research on "more efficient blockchains"

Theme 2^{3M}: DApp Cost Efficiency

Recall the communication-channel view of blockchain ecosystem



Theme 2[3M]: DApp Cost Efficiency: System Model



Two middlewares in BKC-client comm. channel.

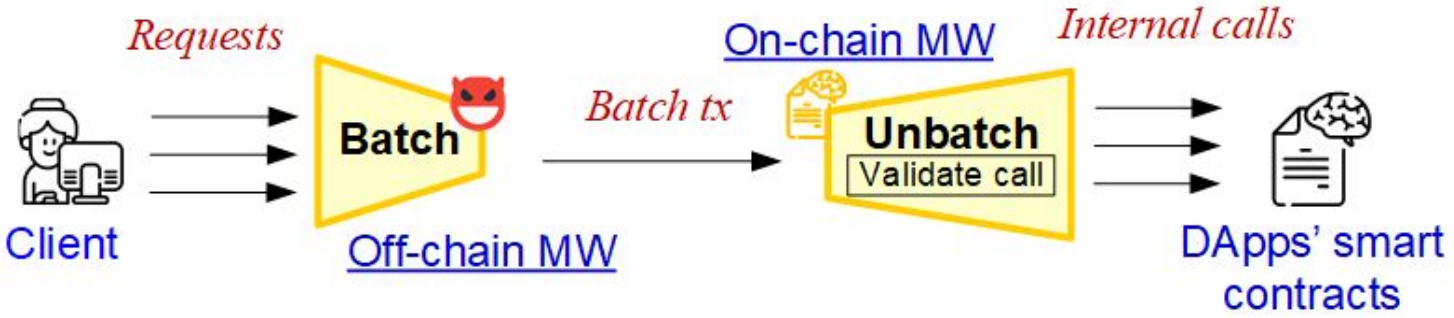
- Untrusted off-chain relay server
- Trusted on-chain smart contract.

Approach: Design & impl. cost-optimization schemes in the two blockchain middlewares.

Theme 2[3M]: DApp Cost Efficiency

Optimization approach 1:

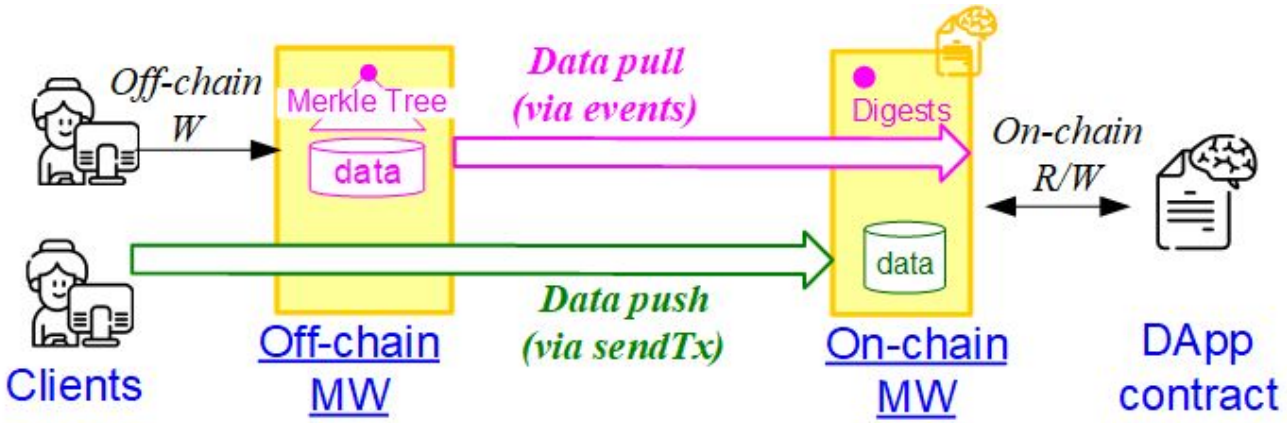
- Batched processing of smart-contract invocations (FSE'21)



Theme 2[3M]: DApp Cost Efficiency

Optimization approach 2:

- Dynamic data replication on/off-chain (Middleware'20)



Talk Outline

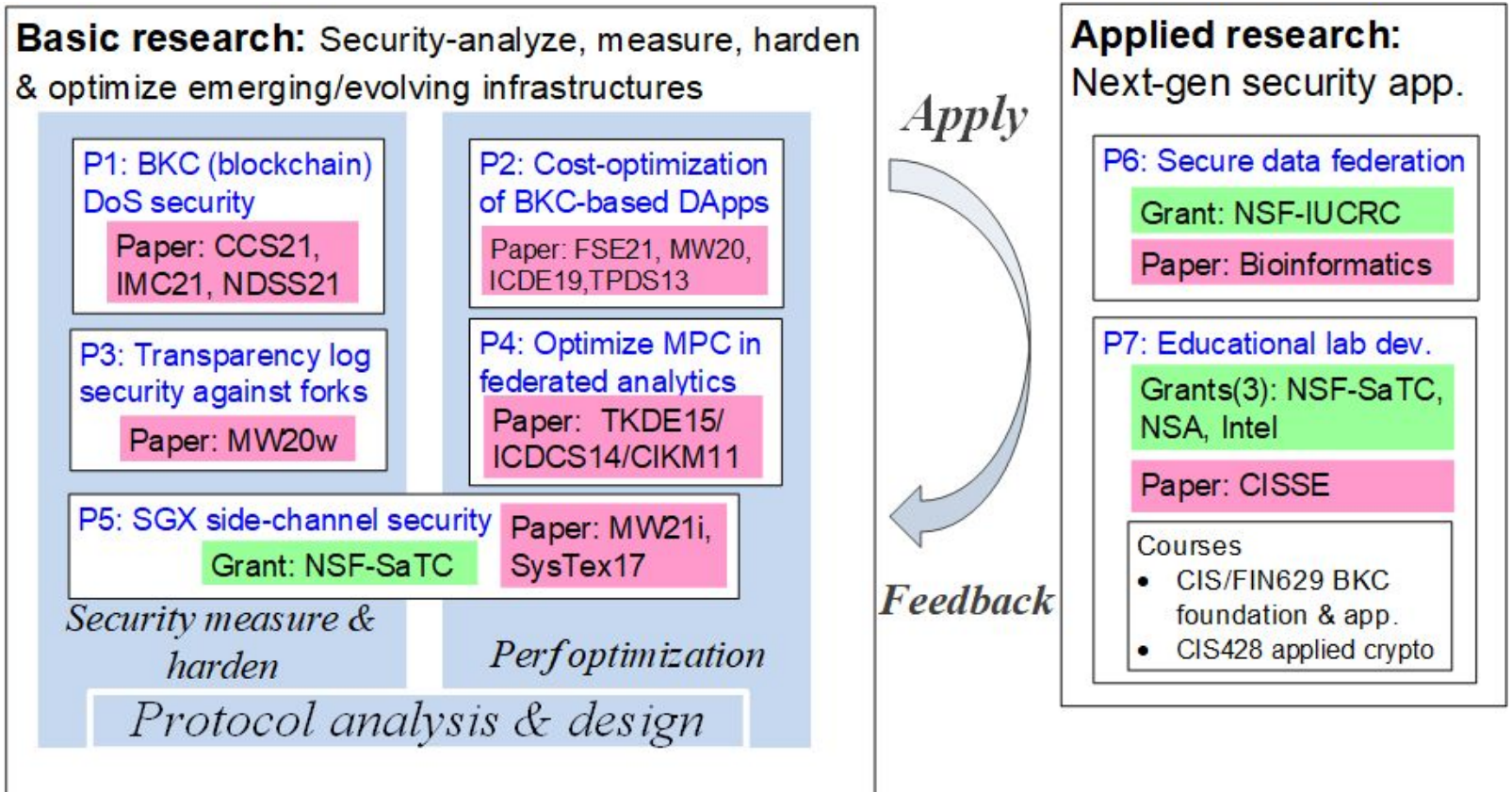
Theme 1: Securing blockchains under DoS vectors

Theme 2: Optimizing the DApp cost without losing security

Overview of Other Research Themes

Future Research Directions

Current Research: *Methods*, Projects, Grants & Papers.



Acknowledgement

- Collaborators: XiaoFeng Wang, Xiapu Luo, Jianliang Xu, etc.

- My students: Kai Li, Yibo Wang, Jiaqi Chen, Yuxuan Zhou, Sencer Burak Somuncuoglu

- Grant support from NSF, NSA, AFRL and industrial gifts from Intel.

Takeaway Points

- Research methodology
  - Target (sub)system: large-scale infrastructures, emerging/evolving features, code in security/cost-critical path.
  - Security research: Understand, measure & harden security?
  - Systems research: Analyze & optimize performance?

- Example research themes in this talk
  - Securing blockchains under DoS vectors (CCS, NDSS, IMC)
    - Hard problem, clever methods, and significant results
  - Cost-optimizing DApps without losing security (FSE, MW, ICDE)

Backup Slides

Bug Reporting & Ethical Considerations

Ethical measurement on mainnet

- On mainnet, the test evicts at most 10% txs in the txpool.
- As remedy, we resent 10% txs after each test to “refill” the txpool.
- No impacts on txs already included in the block.