Defeating Spy Nodes on the Monero Network

Research by :

Rucknium at the Monero Research Lab

Boog900, cuprate developer

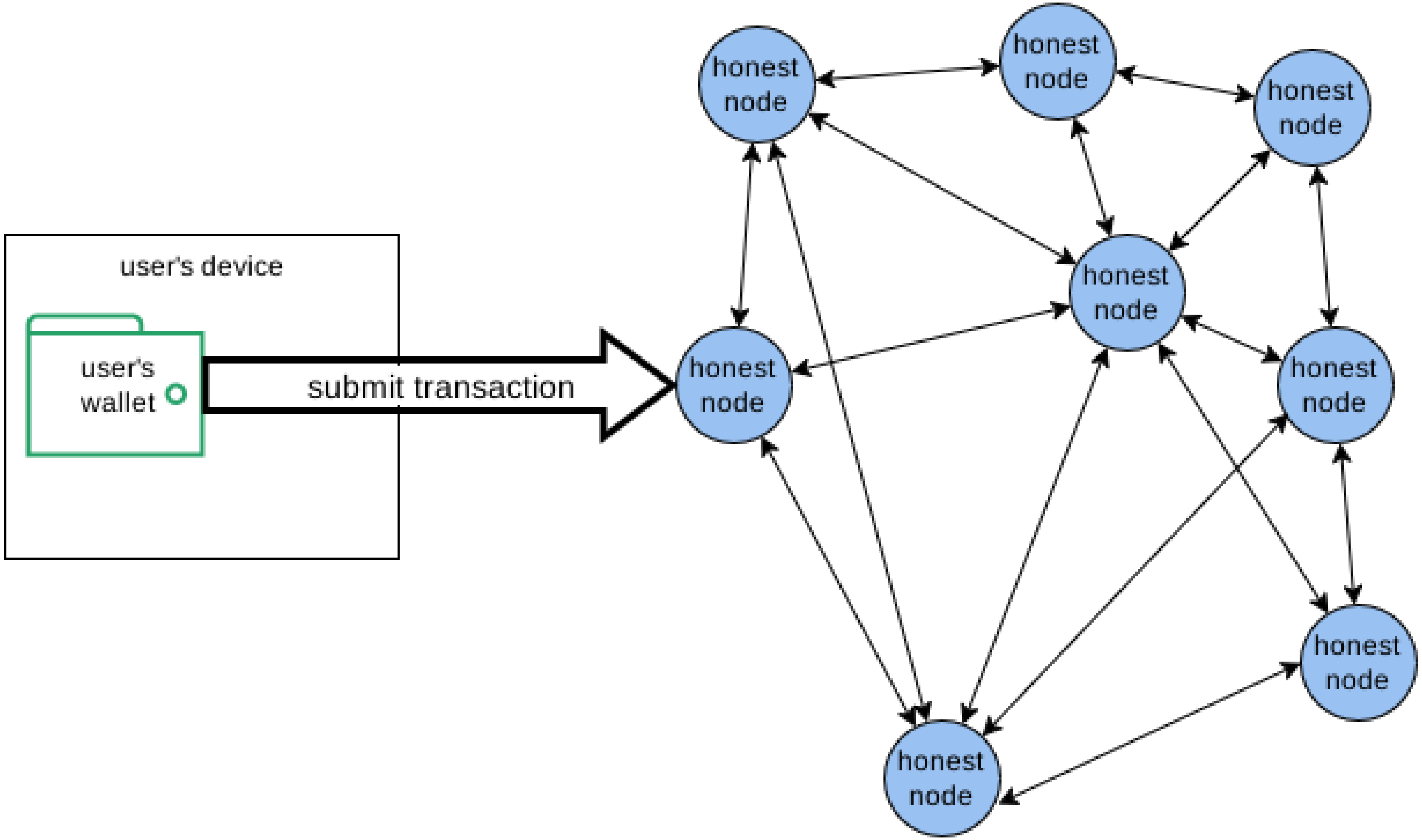
Voiced by xenu

MoneroKon 2025

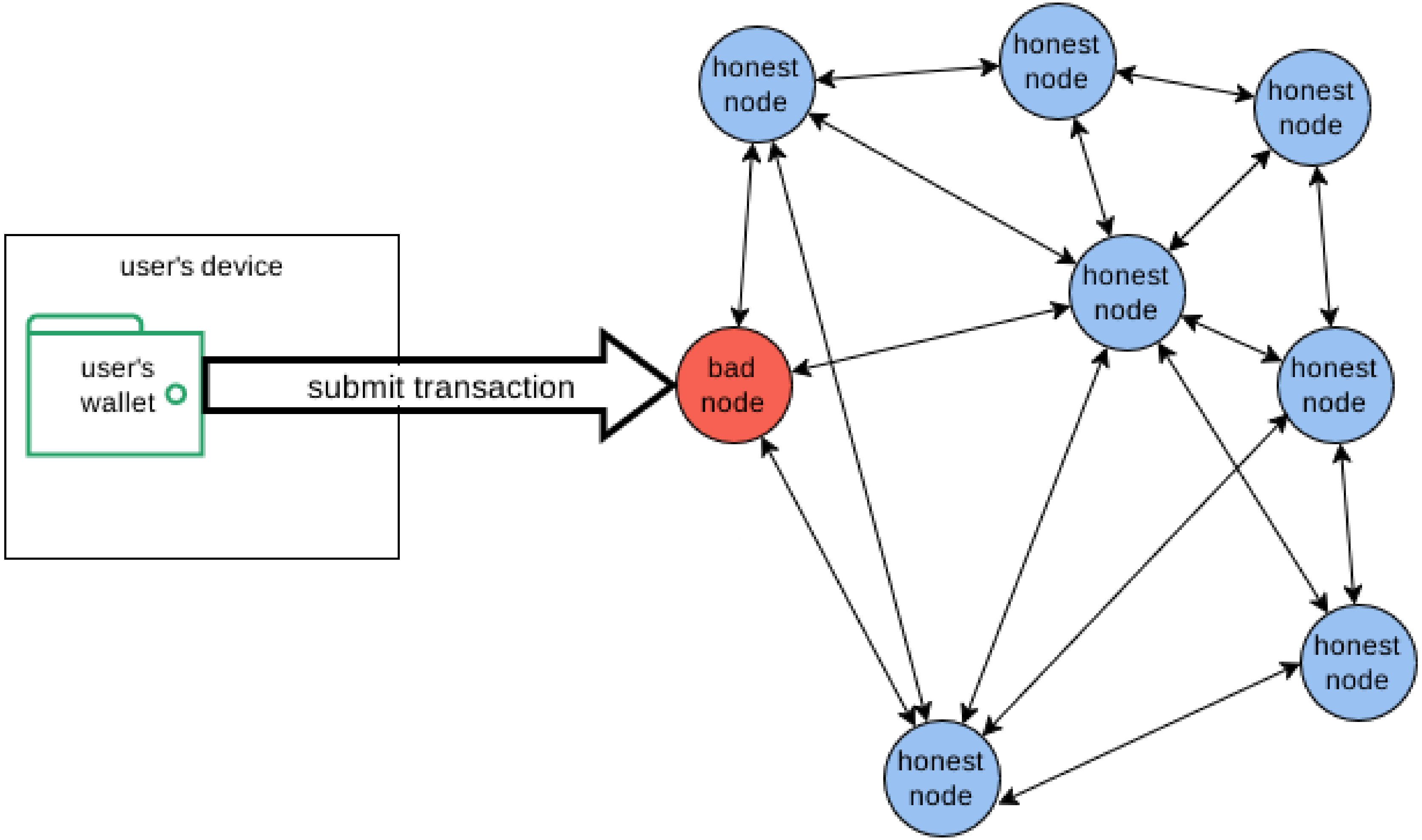
Network-level privacy, with FCMP

- Full-Chain Membership Proofs will vastly improve the privacy of blockchain data, but adversaries can still use network data to try to de-anonymize users.
- A transaction can be linked to an IP address, which can be linked to a real-life identity.
- Multiple transactions broadcast from a single IP address can reveal that the transactions all belong to the same user.

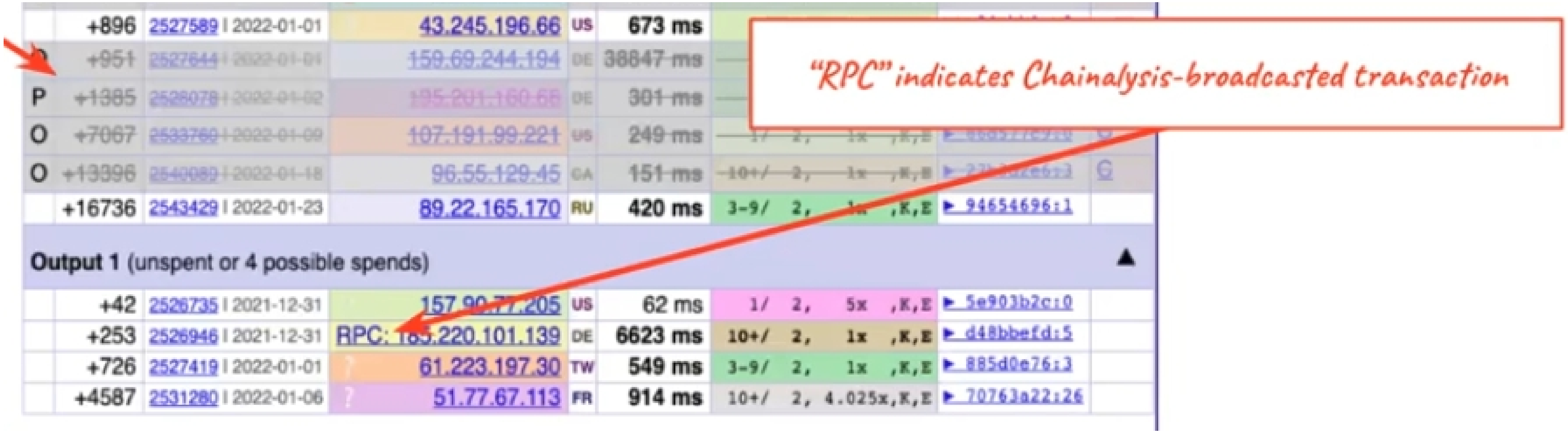
Remote node: Ideal case



Remote node: Malicious remote node records the IP address that submitted the transaction



Spying remote nodes in 2024 Chainalysis leaked video



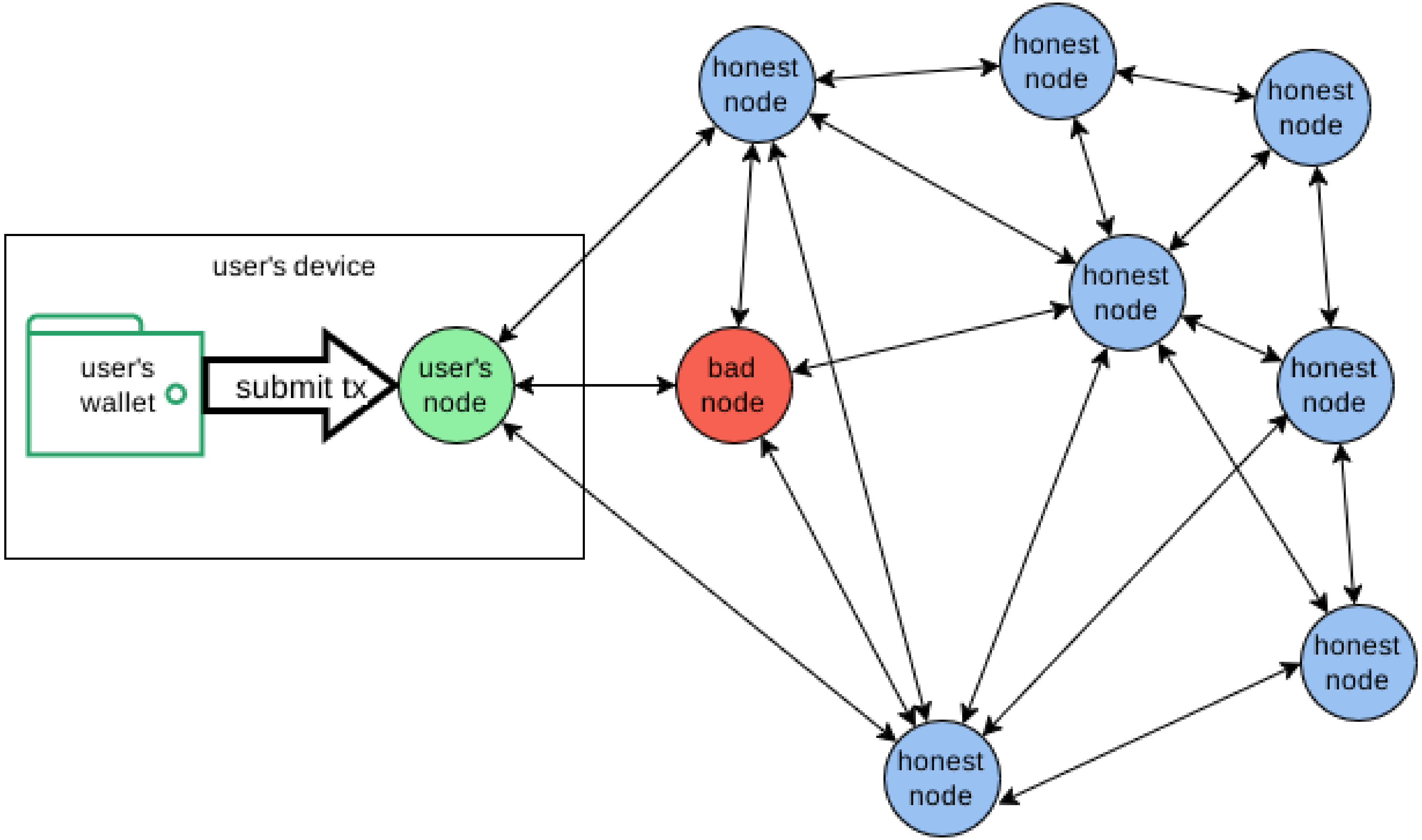
	+896	2527589 | 2022-01-01	43.245.196.66	US	673 ms	[illegible]	[illegible]	
[illegible]	~~+951~~	~~2527644 | 2022-01-01~~	~~159.69.244.194~~	DE	~~38847 ms~~	[illegible]	[illegible]	
P	~~+1385~~	~~2528078 | 2022-01-02~~	~~195.201.160.66~~	DE	~~301 ms~~	[illegible]	[illegible]	
O	~~+7067~~	~~2533760 | 2022-01-09~~	~~107.191.99.221~~	US	~~249 ms~~	~~1/ 2, 1x ,K,E~~	~~► [illegible]~~	G
O	~~+13396~~	~~2540089 | 2022-01-18~~	~~96.55.129.45~~	CA	~~151 ms~~	~~10+/ 2, 1x ,K,E~~	~~► [illegible]e6:3~~	G
	+16736	2543429 | 2022-01-23	89.22.165.170	RU	420 ms	3-9/ 2, 1x ,K,E	► 94654696:1	
Output 1 (unspent or 4 possible spends)								▲
	+42	2526735 | 2021-12-31	157.90.77.205	US	62 ms	1/ 2, 5x ,K,E	► 5e903b2c:0	
	+253	2526946 | 2021-12-31	RPC: 185.220.101.139	DE	6623 ms	10+/ 2, 1x ,K,E	► d48bbefd:5	
	+726	2527419 | 2022-01-01	? 61.223.197.30	TW	549 ms	3-9/ 2, 1x ,K,E	► 885d0e76:3	
	+4587	2531280 | 2022-01-06	? 51.77.67.113	FR	914 ms	10+/ 2, 4.025x,K,E	► 70763a22:26	

"RPC" indicates Chainalysis-broadcasted transaction

Possible solution: Instead of using a possibly malicious remote node, run your own node on a device you control

- This talk is *not* about privacy when using a remote node.
- This talk is about privacy when running your own node.

Even when running your own node, spy nodes can create a privacy problem



Spy nodes are not just a hypothetical!

- In 2024, boog900 uncovered evidence of a large number of spy nodes on the Monero network

- These spy nodes behave differently from honest nodes. And they are mostly concentrated in small IP address groupings, called “subnets”.

Let’s define some terms

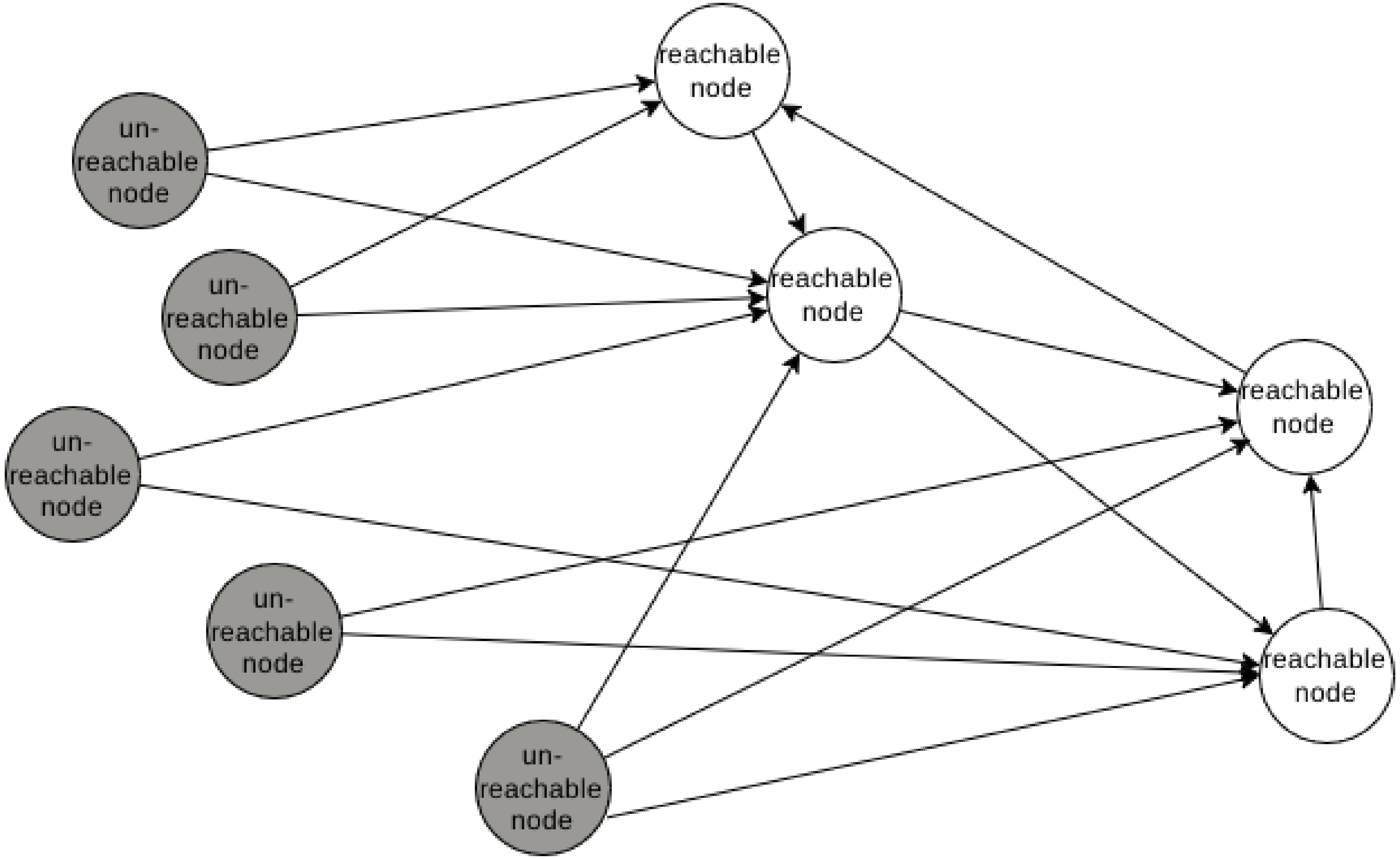
- Outbound/inbound connections
    - Monero nodes can form two types of connections: outbound and inbound
    - The difference between the two is which node *initiates* the connection. Both outbound and inbound connections can carry transactions and blocks
    - When my node initiates the connection, it is an *outbound* connection from my node’s point of view. My node chooses which node to connect to.
    - When another node connects to my node, it is an *inbound* connection from my node’s point of view. My node doesn‘t actively choose which nodes connect to it.
    - By default, Monero nodes allow up to 12 outbound connections and an unlimited number of inbound connections. (Most nodes that allow inbound connections have over 100 inbound connections.)

Reachable/unreachable nodes

- Most nodes do not accept inbound connections. These are “unreachable” nodes.

- When most users set up Monero nodes at home without any special configuration, their ISP or router will block inbound connections.

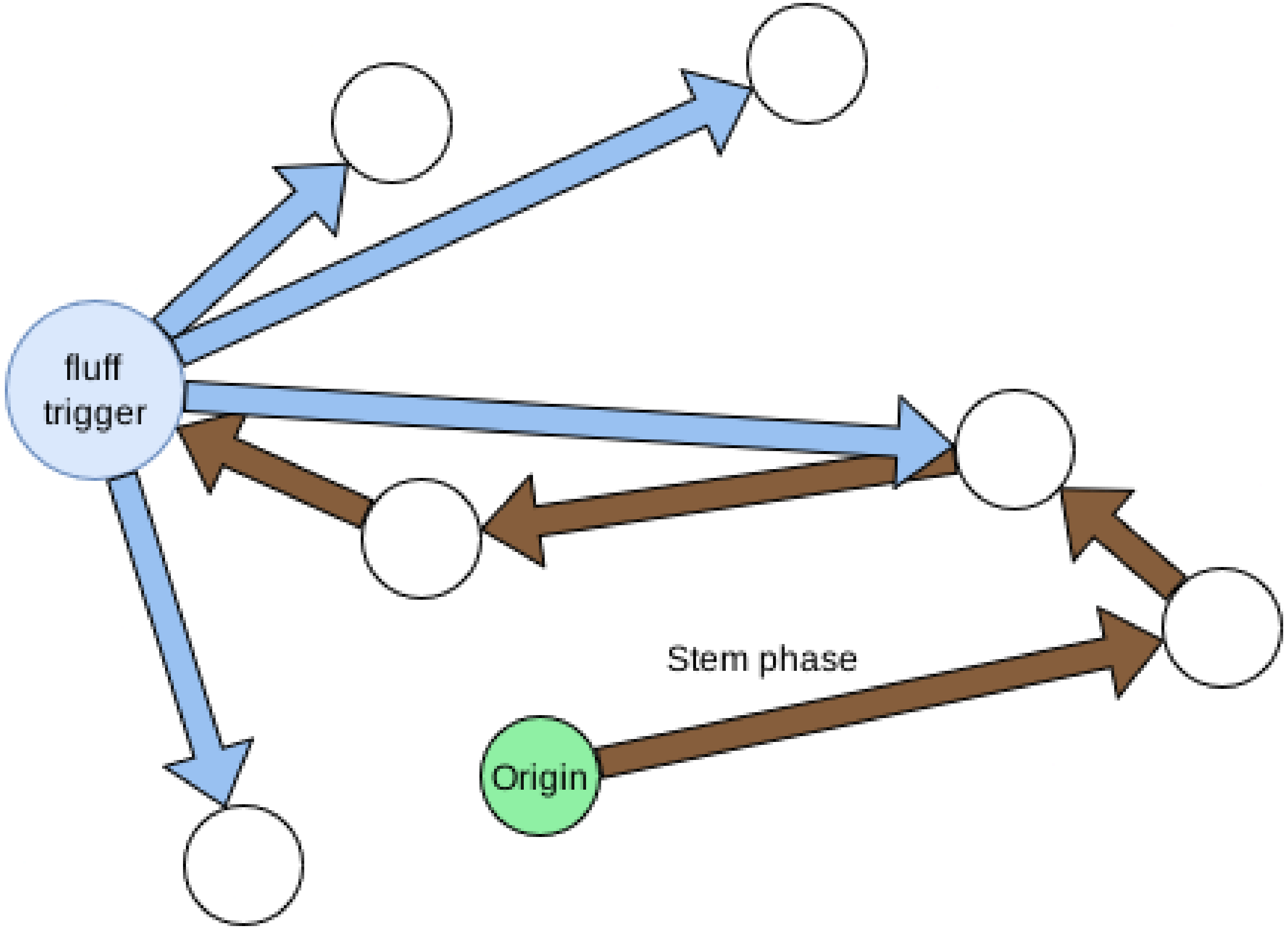
Reachable/unreachable nodes



Current mitigation: Dandelion++

- Dandelion++ is implemented in monerod to make linking transactions to IPs more difficult.
- Dandelion++ works in two stages: the privacy-preserving stem, and the fast-propagating fluff.
- When a node is in the stem state, it relays stem-phase transactions through one of two outbound connections it receives.
- All transactions start in the stem state, getting routed to 1 outbound peer at a time until reaching a node in the fluff state. Then, they will be diffused to all connections.

Dandelion++ diagram



Dandelion++ is not perfect

- An adversary’s probability of guessing the IP address origin of a transaction still scales up when there are many spy nodes.
- It only routes stem transactions to outbound peers, unreachable nodes have no transactions to mix with their own transactions.
- Any stem transactions from an unreachable nodes can be assumed to have originated from that node.

Discovery of active spy node threat

What is the method to tell apart spy nodes?

- Up until now, the method to detect spy nodes for the recommended ban list has been kept a secret to prevent the spy from updating their nodes.
- A ban list is not a good long-term solution, though, so the MRL decided to release the method once the subnet deduplication countermeasure has been written and reviewed.
- By releasing the method, we hope to encourage more research into spy nodes and proxy node prevention.

How we tell apart spy nodes for the ban list

- Every Monero node has a self-assigned 64-bit identifier called a peer_id (on privacy networks like Tor, the peer_id is set to 1).
- The peer_id is randomly generated when a node is started and does not change until the node is restarted again.
- The peer_id is shared during a P2P handshake and ping.

How we tell apart spy nodes for the ban list

- The spy nodes look normal during a handshake. Their peer_id will be unique.
- When you send a ping, however, their peer_id will be different to the one given during a handshake.
- An example of a bad node:

159.89.163.68:18080	
handshake: 2889744552615260348	ping: 17019777233037429604

- An example of a good node:

*:18080	
handshake: 17019777233037429604	ping: 17019777233037429604

How we tell apart spy nodes for the ban list

- The spy nodes look normal during a handshake. Their peer_id will be unique.
- When you send a ping, however, their peer_id will be different to the one given during a handshake.
- An example of a bad node:

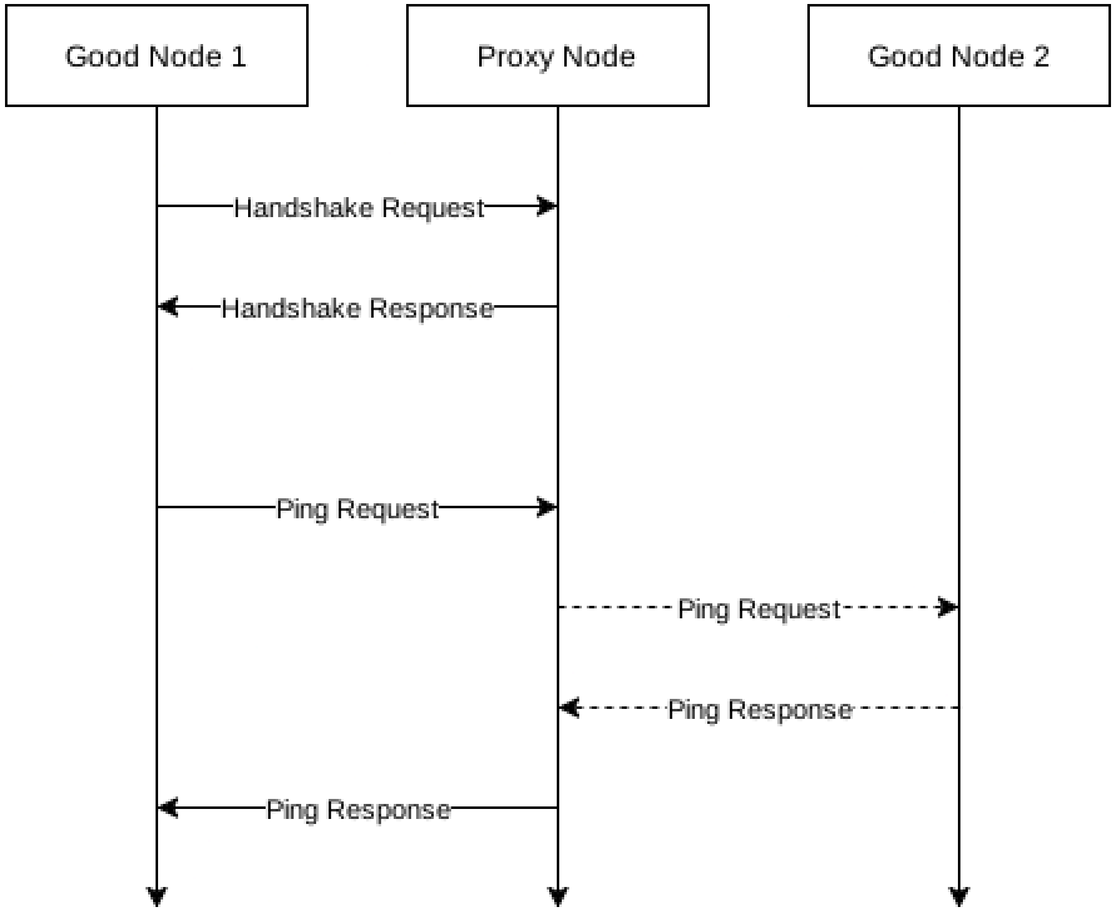
159.89.163.68:18080	
handshake: 2889744552615260348	ping: 17019777233037429604

- An example of a good node:

*:18080	
handshake: 17019777233037429604	ping: 17019777233037429604

- The bad node is returning the same peer_id as the good node during a ping!

What this means



Code for finding spy/proxy nodes: https://github.com/Boog900/p2p-proxy-checker

May 2025 network scan

- Total number of IP addresses with reachable nodes: **4607**
- Number of IP addresses with reachable nodes on the spy node list: **1843**
- Probability that the status quo peer selection algorithm selects a spy node when it chooses a peer: **31%**

Spy node behavior

- Spy nodes seem to relay transactions and blocks in the same manner as honest nodes.
- The average time that a connection stays alive, in minutes, is lower than the average of honest nodes
- Most spy nodes only share IP addresses of other spy nodes during a peer list exchange handshakests
- Spy nodes make very few outbound connections
  - Behavior is consistent with optimal attack strategy against Dandelion++, since nodes never receive stem-phase transactions from their outbound connections

Spy node behavior: Subnets

- A subnet is a set of adjacent IP addresses.
- An important grouping is the “/24” subnet level, which are the 254 IP addresses that all share the first three numbers in a four-number IP address.
    - Example: Subnet **91.198.115.0/24**, which includes all IP addresses from **91.198.115.0** to **91.198.115.255**
- It is cheaper to rent IP addresses in bulk as subnets.
- Spy nodes are using 6 subnets, where they have spy nodes at all 254 IP addresses
- They also have a few hundred IP addresses that are separate from these subnets lease in bulk.

CIA analysis

- In information security, the CIA triad is a set of properties that a data management protocol should provide

- **Confidentiality**: Data should only be available to those users who have a right to know it.
- **Integrity**: Data should not be changed or lost in an accidental or malicious manner
- **Availability:** Data should be accessible to users at all times.

Network-level threats to Confidentiality

- Linking a transaction to an IP address
- Linking multiple transaction to each other, through a shar IP address

Network-level threats to Integrity

- Network partitioning
- Eclipse
- Double spending
- Unfair miner revenue

Network-level threats to Availability

- Low bandwidth
- Denial-of-Service attack
- Insufficient reachable nodes

Countermeasure: Ban lists

- monerod has a system for specifying a file with a list of addresses to ban.
- monerod also has a system to propagate addresses to ban over DNS but that list is currently full.
- Limitations:
  - You must be able to detect spy nodes from real nodes.
  - People using the ban list must trust the person who created it.
  - The list must be kept up to date if they switch addresses.
  - If the method to detect spy nodes becomes public, the spy node operator(s) can patch their node software.
- Monero Research Lab recommendation: Configure your node to use this ban list: github.com/monero-project/meta/issues/1124

	Confidentiality	Integrity	Availability	Deployment
Ban lists	B	B	B	6% estimated adoption

Countermeasure: Tor/I2p

- monerod has partial Tor/I2P support for the P2P network.
- The Tor/I2P network zones are only used to send transactions one hop, then they are propagated over clearnet.
- The node software does not directly support full blockchain synchronisation over Tor and I2P because of the possibility of Sybil attacks.
  - It costs nothing to generate a new address identity on Tor

Tor/I2P Denial-of-Service attacks

- Tor was under a denial-of-service attack for almost a year, from June 2022 to April 2023.

- I2P also suffered a denial-of-service attack in 2023.

	Confidentiality	**Integrity**	**Availability**	**Deployment**
Tor/I2P	A-	D	C	Optional

Countermeasure: Clover

- Clover is an alternative to Dandelion++.
- Like D++ it has a privacy stage and a fast propagation stage. Unlike D++ it includes inbound peers in its privacy stage.
- In the privacy stage, any tx received from an outbound peer is routed to an outbound peer, chosen at random for each tx. For any tx received from an inbound peer, a weighted coin is flipped to decide whether to fast propagate it or route it to another random inbound peer.
- The fast propagation stage is the same as D++.

Clover

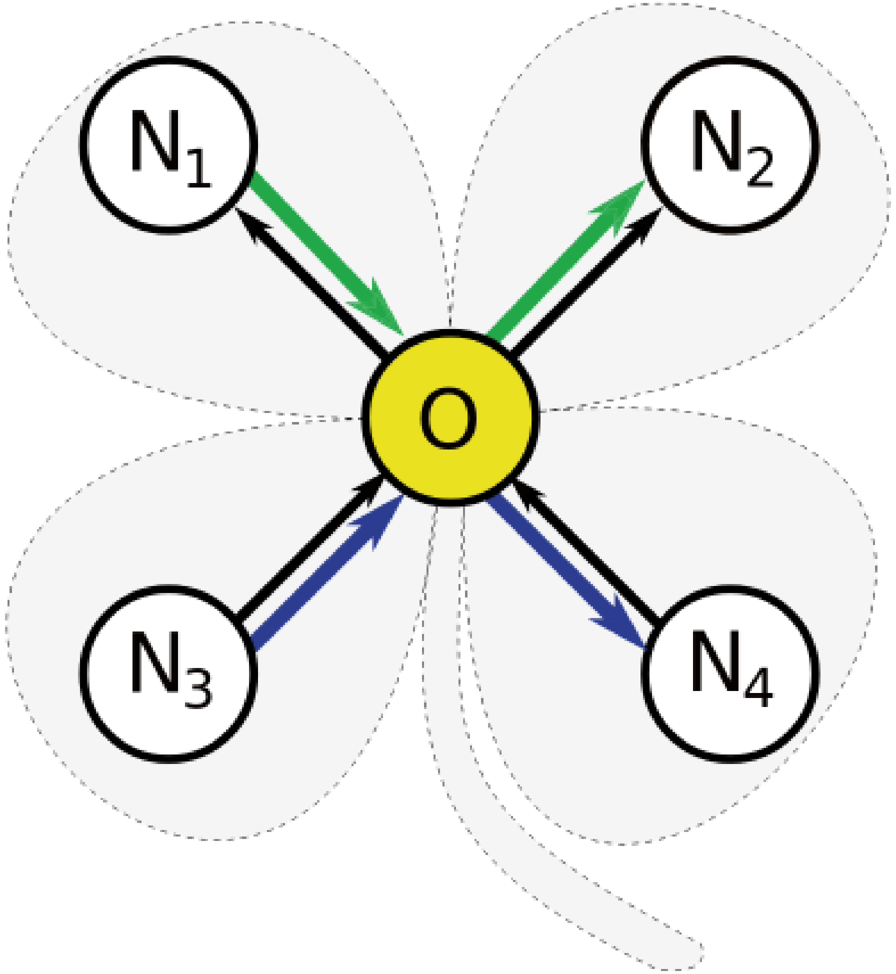


Fig. 1 The Clover relay protocol pattern: black arrows represent the direction of the connection; colored arrows represent relays of proxy transactions

Clover downsides

- The Clover paper did a less thorough self-analysis than the D++ paper.
    - The theoretical analysis has questionable assumptions
    - The simulation results do not use the same metrics as D++
- Presumably vulnerable to intersection attacks due to each tx taking an independent path.
- Plausible deniability is maximized when more transactions appear to come from few nodes, the best case is the spy seeing all transactions originate from one node.
    - In Dandelion this is why a line graph is used, although D++ changes this due to the fact line graphs perform badly when the line graph is known to the spy. However, in Clover there is no separate privacy graph from the connection graph.

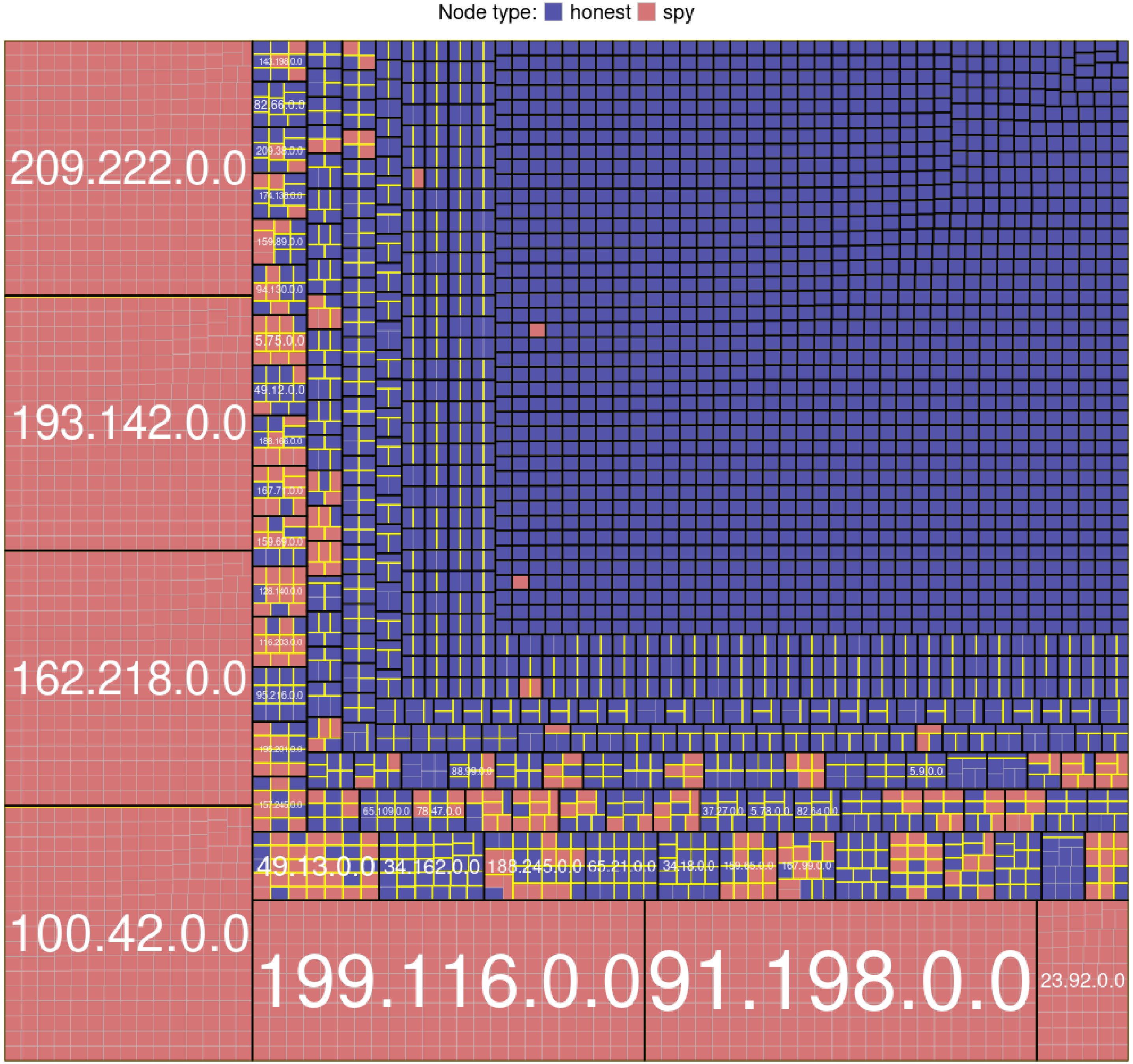
	Confidentiality	Integrity	Availability	Deployment
Clover	C	B	B	Needs more study

Countermeasure: Subnet deduplication

- Renting whole IP address subnets in bulk is a cheap way for a spying adversary to attract more connections from honest nodes.

Subnet treemap of honest and spy nodes

Black perimeters indicate /16 subnet groupings. Yellow indicates /24 subnets.



Subnet deduplication

- Simple idea: when there is more than one candidate peer in a given subnet, randomly remove peers from consideration until there is only one candidate peer per subnet.
    - Then, choose your peer from the deduplicated candidate peer list.
- This peer selection algorithm would reduce the influence of spy nodes that are concentrated in a few subnets
- According to simulations, this simple change would reduce the percentage of spy nodes on candidate peer list from 31% to 9%.
- Even if the spying adversary stops buying subnets in bulk, subnet deduplication is not worse than the status quo algorithm
- Forthcoming Monero Research Lab bulletin:
    - “Subnet Deduplication for Monero Node Peer Selection” by Rucknium and Boog900

	Confidentiality	**Integrity**	**Availability**	**Deployment**
Subnet deduplication	B	B	B	Implementation under review (PR #9939 by rbrunner7)

Countermeasure PPoS: Practical Proof of Storage for Blockchain Full Nodes

- PPoS is a way to link the stored blockchain to the claimed internet address, so that every reachable address must store its own copy of the blockchain.
- It works by using a mapping function that is expensive to map and cheap to unmap and having nodes store the mapped data.
- The address of the node and previous mapped blocks are used as a key for the mapping to make each copy unique.
- To verify a node is storing a unique copy of the blockchain, a request is sent for random consecutive blocks. The prover will provide the hash of each mapped block in the range with some full mapped blobs chosen pseudo-randomly.
    - The prover must respond in under a certain amount of time to prevent them from performing the mapping on the fly.
- If the node responded fast enough and the data is mapped correctly they pass the test.

PPoS Downsides

- We can only use it on one network zone (i.e. clearnet, Tor, I2P), or we can link addresses across network zones, or we can require every node which operates across network zones to store a copy of the blockchain for each zone.
- Although unmapping is cheaper than mapping, it is still an added cost that needs to happen whenever blocks are requested. Adding even more load to nodes.
  - Monero's current wallet sync protocol requests blocks and pruned tx blobs, so this added cost is not just on nodes syncing.
- New blocks must be mapped before being stored in the database.

	Confidentiality	**Integrity**	**Availability**	**Deployment**
Practical Proof of Storage	A-	C	B	Needs more study

References

Biryukov & Pustogarov (2015). "Bitcoin over Tor isn't a good idea."

Fanti et al. (2018). "Dandelion++: Lightweight Cryptocurrency Networking with Formal Anonymity Guarantees."

Franzoni & Daza (2022). "Clover: An anonymous transaction relay protocol for the bitcoin P2P network."

Franzoni & Daza (2022). "SoK: Network-Level Attacks on the Bitcoin P2P Network."

Heo, Ramachandran, & Jurdak (2023) "PPoS : Practical Proof of Storage for Blockchain Full Nodes."