Dissent in Numbers: Making Strong Anonymity Scale

David Isaac Wolinsky, Henry Corrigan-Gibbs, and Bryan Ford
Yale University

Aaron Johnson
U.S. Naval Research Laboratory

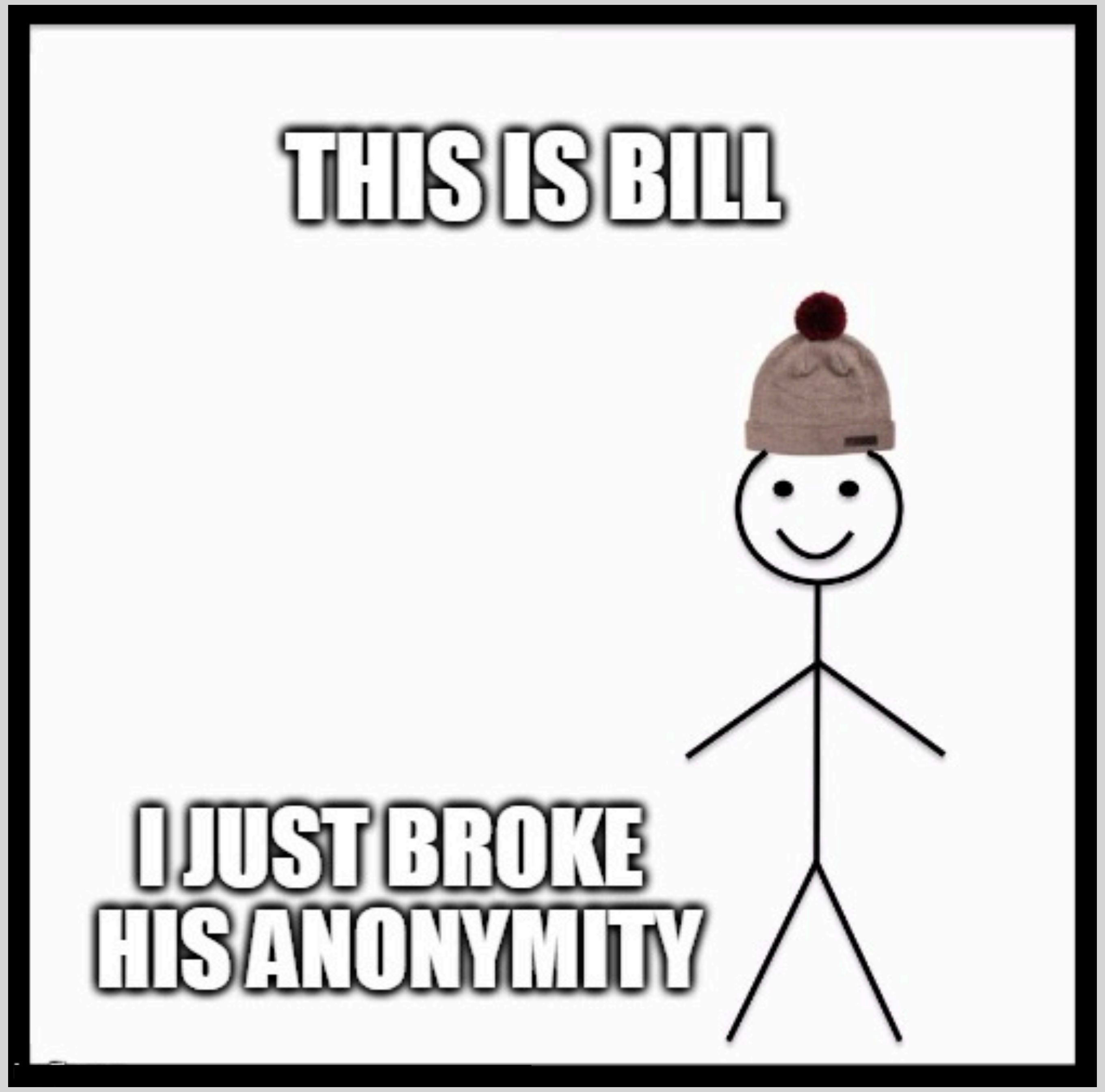
Published at USENIX OSDI'12

Presented By: Sharad Agarwal

23rd March 2021

Outline

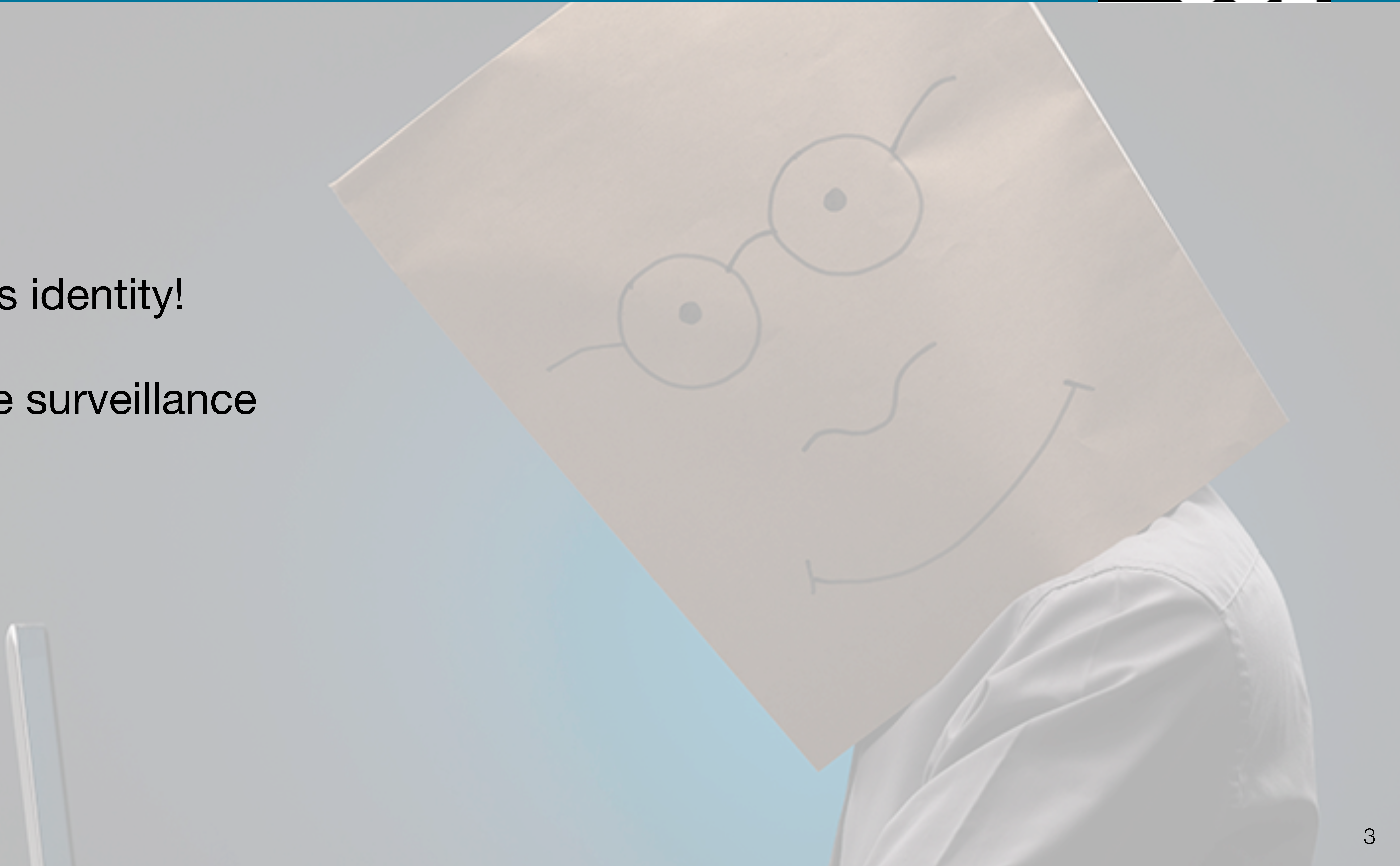
- Let’s talk Anonymity
- Is Tor not enough?
- Intro to Dining-Cryptographers (DC) Net
- Dissent
  - Working
  - Advantages
- Implementation
- Evaluation
- Strengths and Weaknesses



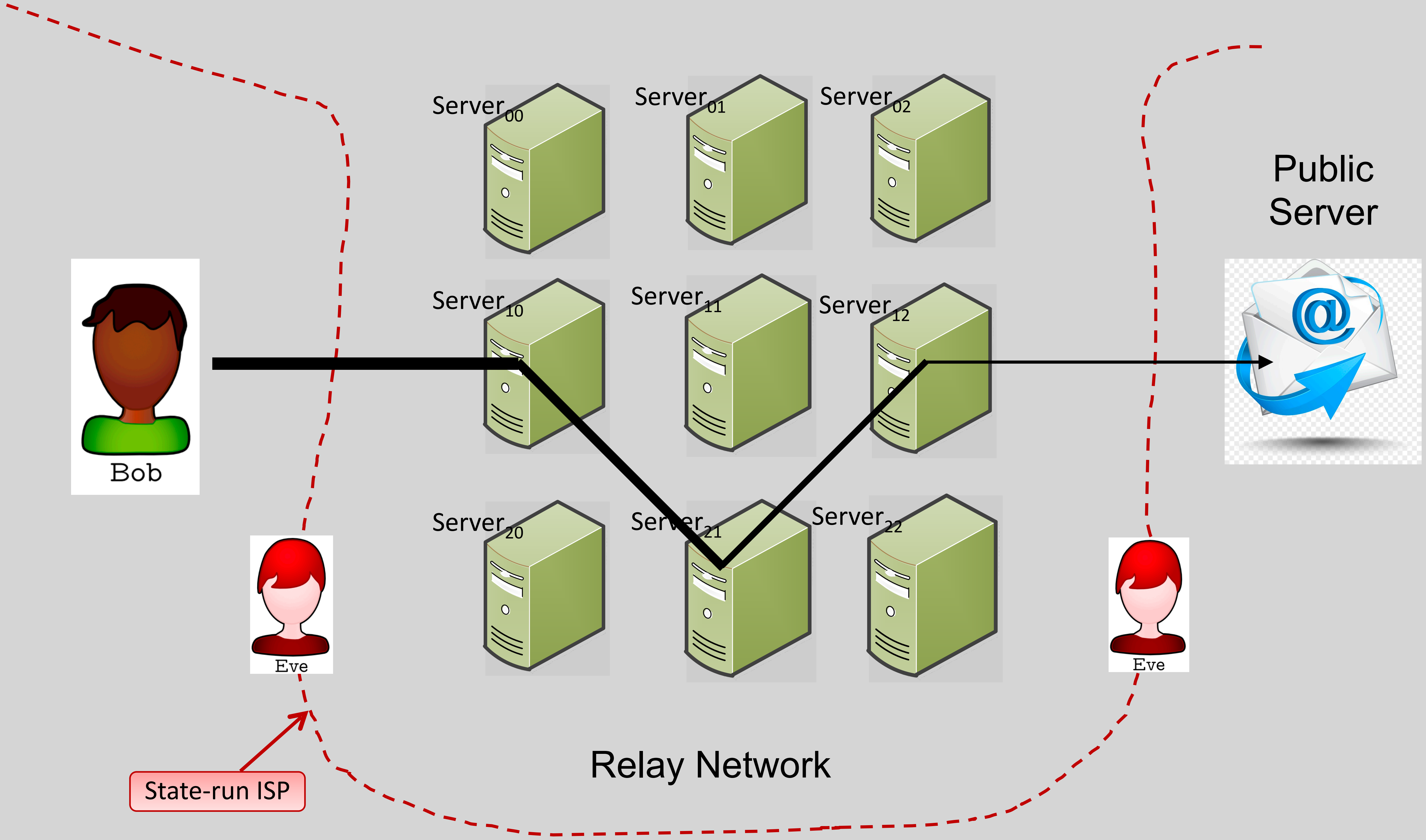
What is Anonymity

Hiding one’s identity!

To defeat online surveillance



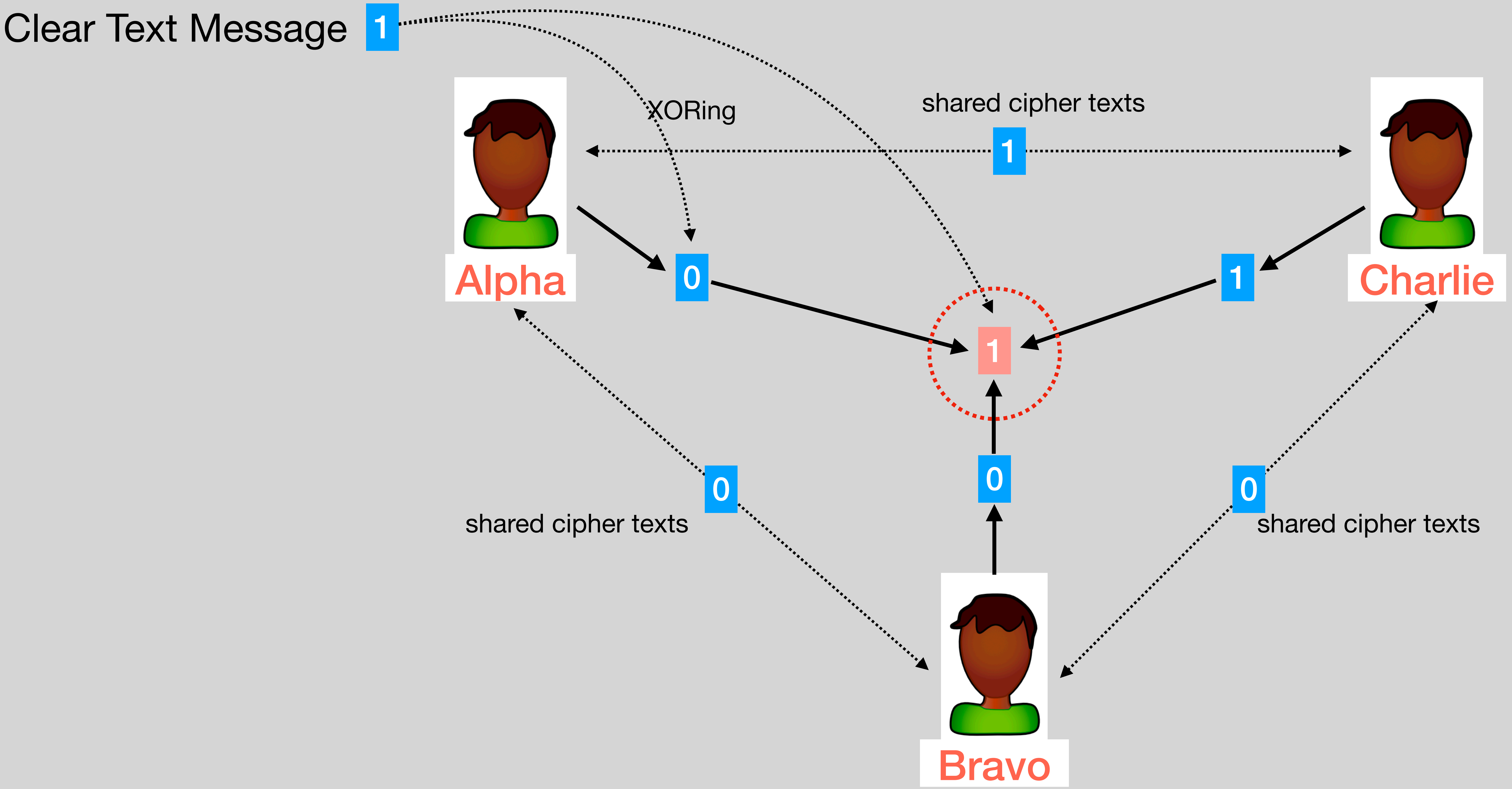
Tor is Scalable but prone to Network Timing Analysis (side-channel attack)



Credits: Wolinsky, David Isaac, et al. "Dissent in numbers: Making strong anonymity scale." *10th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 12)*. 2012.

Dining Cryptographers Net (DC-net)

DC net is resistant against Network Timing Analysis Attack

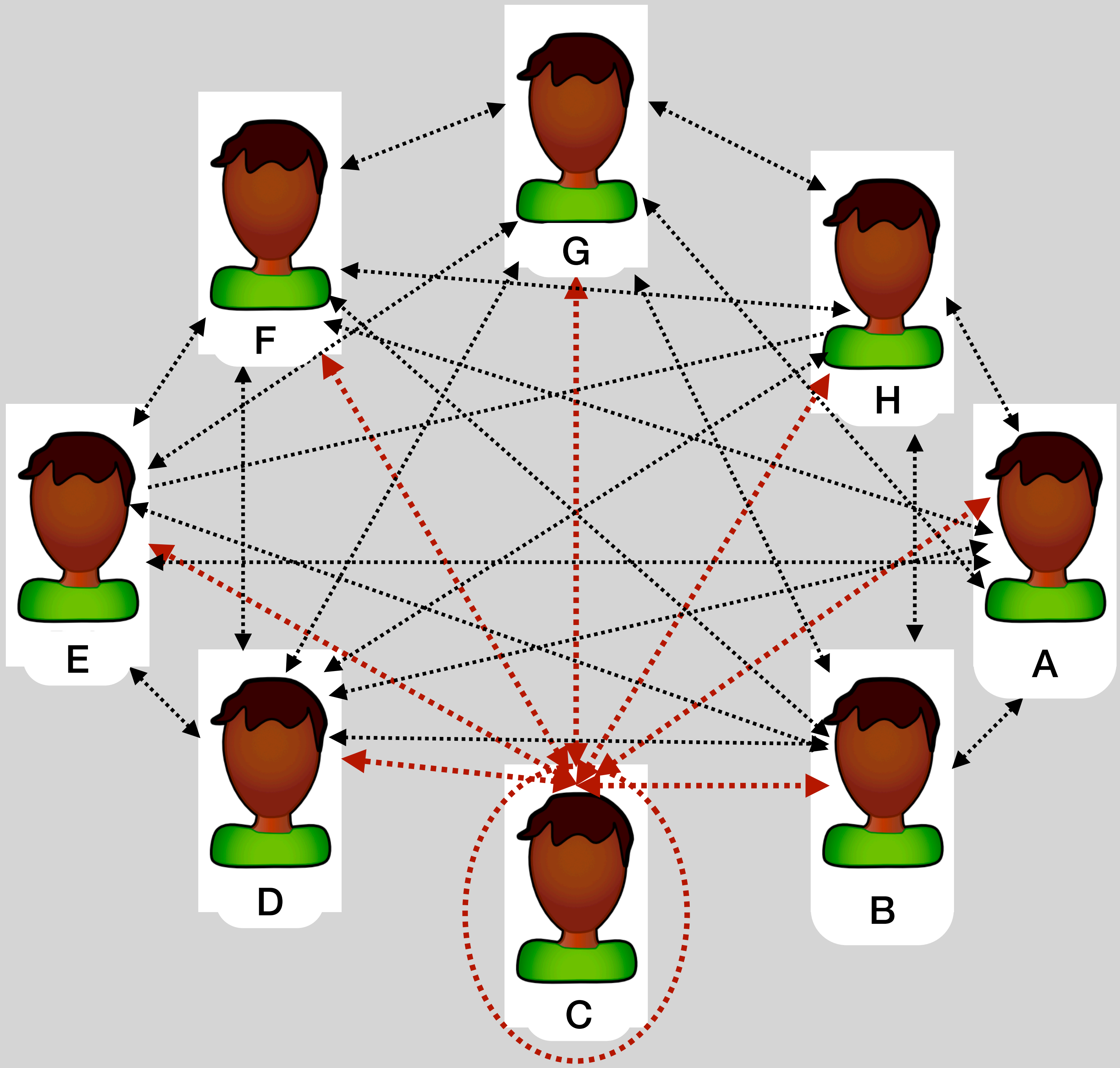


Peer-to-Peer Network sharing secrets.

- Peers have shared secrets as an outcome of coin flip protocol.
- Every user XOR all his shared secrets.
- All member transmits same amount of bits and they do it in sync.

Credits: Wolinsky, David Isaac, et al. "Dissent in numbers: Making strong anonymity scale." *10th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 12)*. 2012.

Dining Cryptographers Net (DC-net)

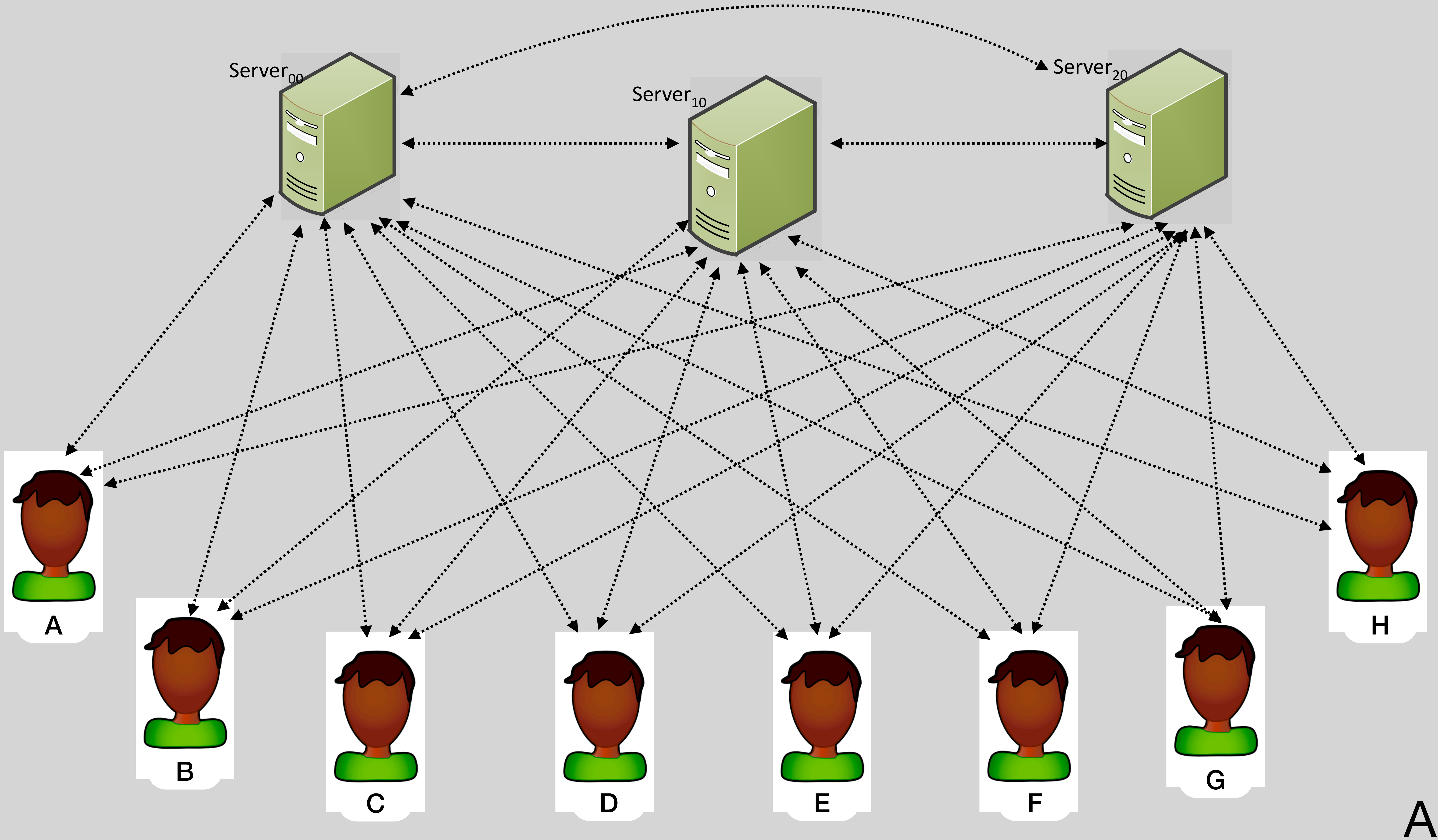


Herbivore is an example which can have at most 40-50 concurrent users only.

- If one member leaves the communication in between or drops out, the whole communication needs to be repeated.
- Computation is redone in that case as the earlier XORs won't hold.

Hence **not Churn Tolerant!**

Dissent



- Clients sends the bit(s) to the server.
- Server waits for a time window.
- Servers communicate and compute the XORs.
- Server send the plain text back to the clients

Assumption: at least one server is honest.

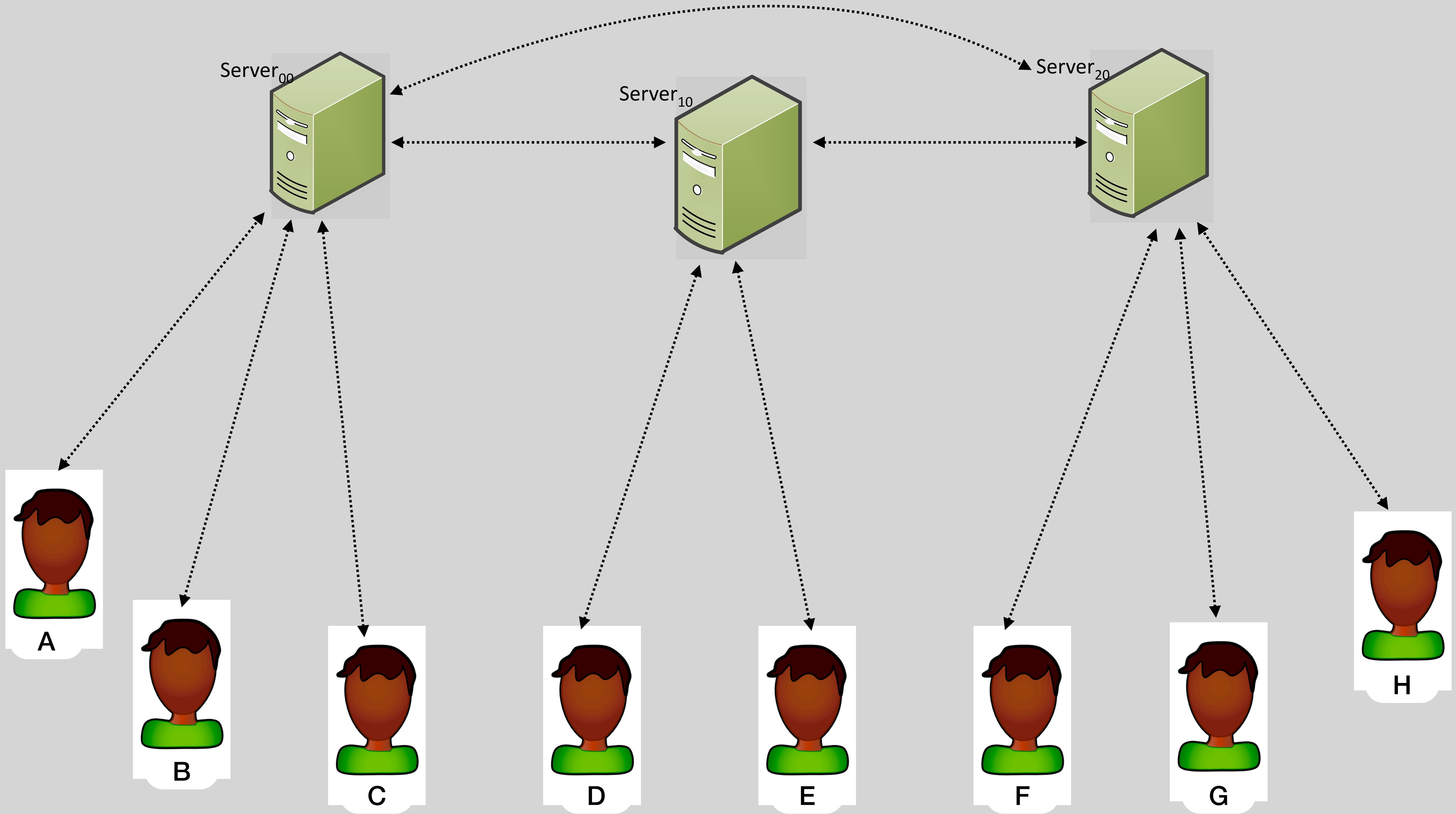
Client-Server model based on DC-net

Dissent Vs DC-Net

	Dissent	DC-Net	Considering 100 clients (and 5 servers in case of Dissent)
Computation	M Servers and N clients: O(M x N) where M << N	$O(N^2)$	DC-Net takes ~10k operations whereas 1k in Dissent
Communication	Can construct DC-Net aware multicast tree (Linear)	$O(N^2)$	DC-Net needs ~10k cipher text exchanges whereas ~200 in Dissent
Churn Tolerance	Protocol continues as normal. Thanks to the servers.	If user leaves, have to repeat the entire protocol.	
Identify Disruptions	Solved using scheduling.	Easily disrupted.	
Scalable Anonymity	More than 5000 clients	Can scale upto 40-50 users eg. Herbivore	

Dissent

DC-Net aware multicast tree



If direct upstream server is malicious, it cannot still decode the transmission without cooperation of all other servers.

Verifiable Shuffles

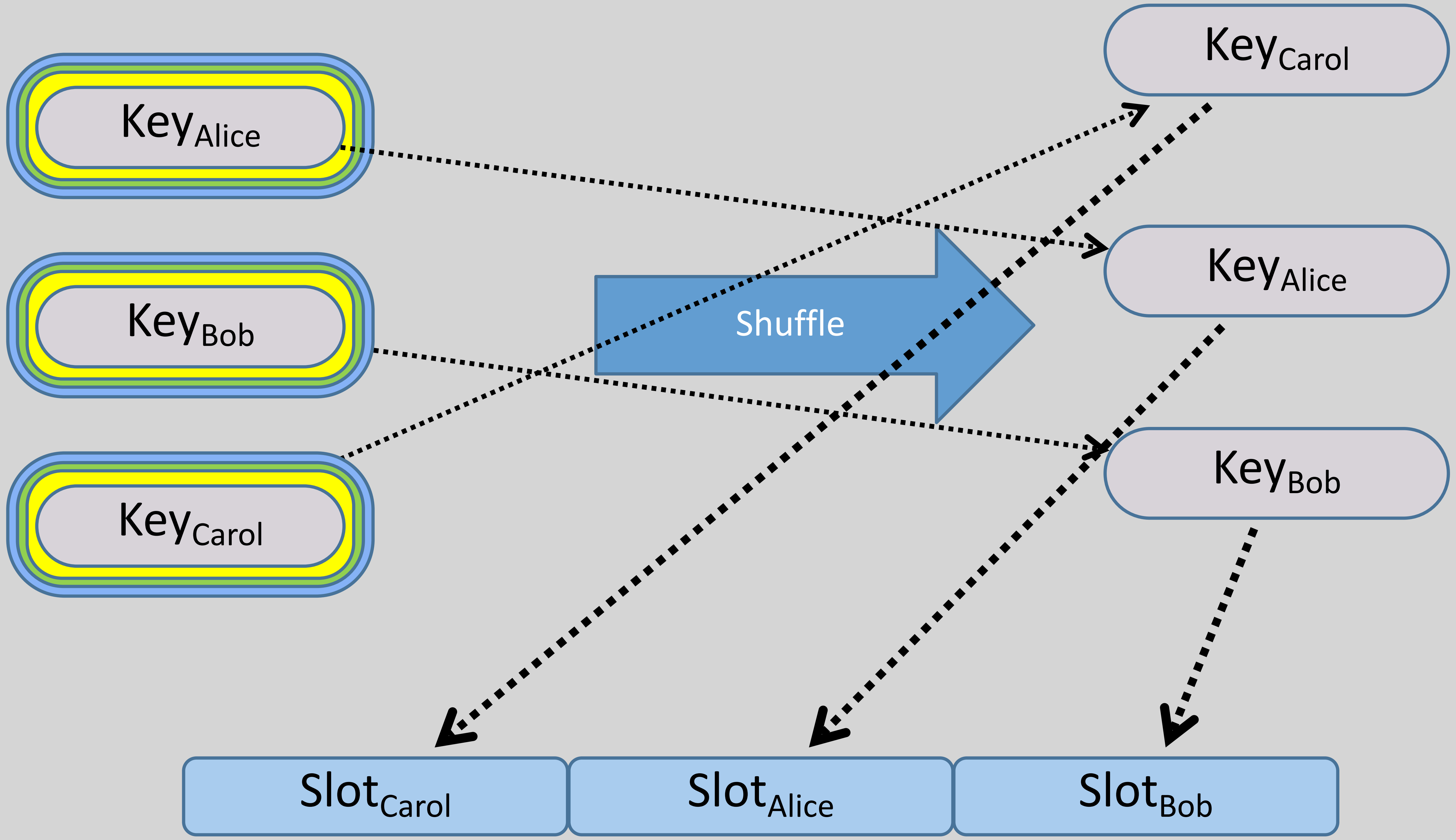
- Schedule and distribute the keys for every round.

- Also used for transmitting accusation to servers.

- Clients submit the messages (keys) to the shuffle protocol.

- Shuffle outputs random permutation of messages (keys).

- Dissent uses the Neff's algorithm for verifiable shuffle.


$\text{Key}_{\text{Alice}}$
Key_{Bob}
$\text{Key}_{\text{Carol}}$
Shuffle
$\text{Key}_{\text{Carol}}$
$\text{Key}_{\text{Alice}}$
Key_{Bob}
$\text{Slot}_{\text{Carol}}$
$\text{Slot}_{\text{Alice}}$
Slot_{Bob}

Credits: Wolinsky, David Isaac, et al. "Dissent in numbers: Making strong anonymity scale." *10th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 12)*. 2012.

Dissent - Implementation

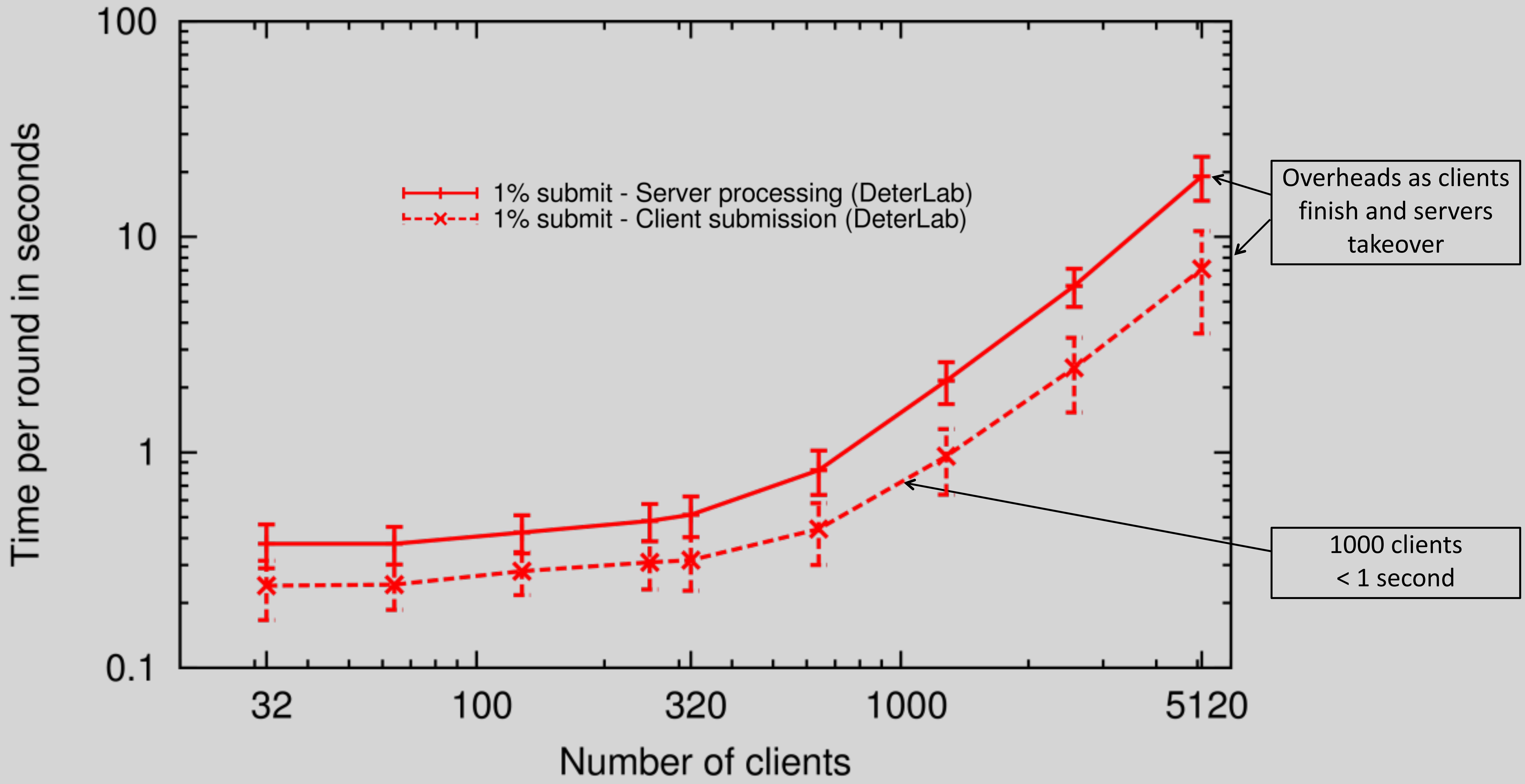
- Implemented in C++
  - Uses Qt framework.

- CryptoPP library

- Assumes a Certificate Authority (CA) - managing public keys of servers and clients.

- User application interact using HTTP API or a SOCKS v5 proxy interface with the Dissent Node.

Dissent - Evaluation



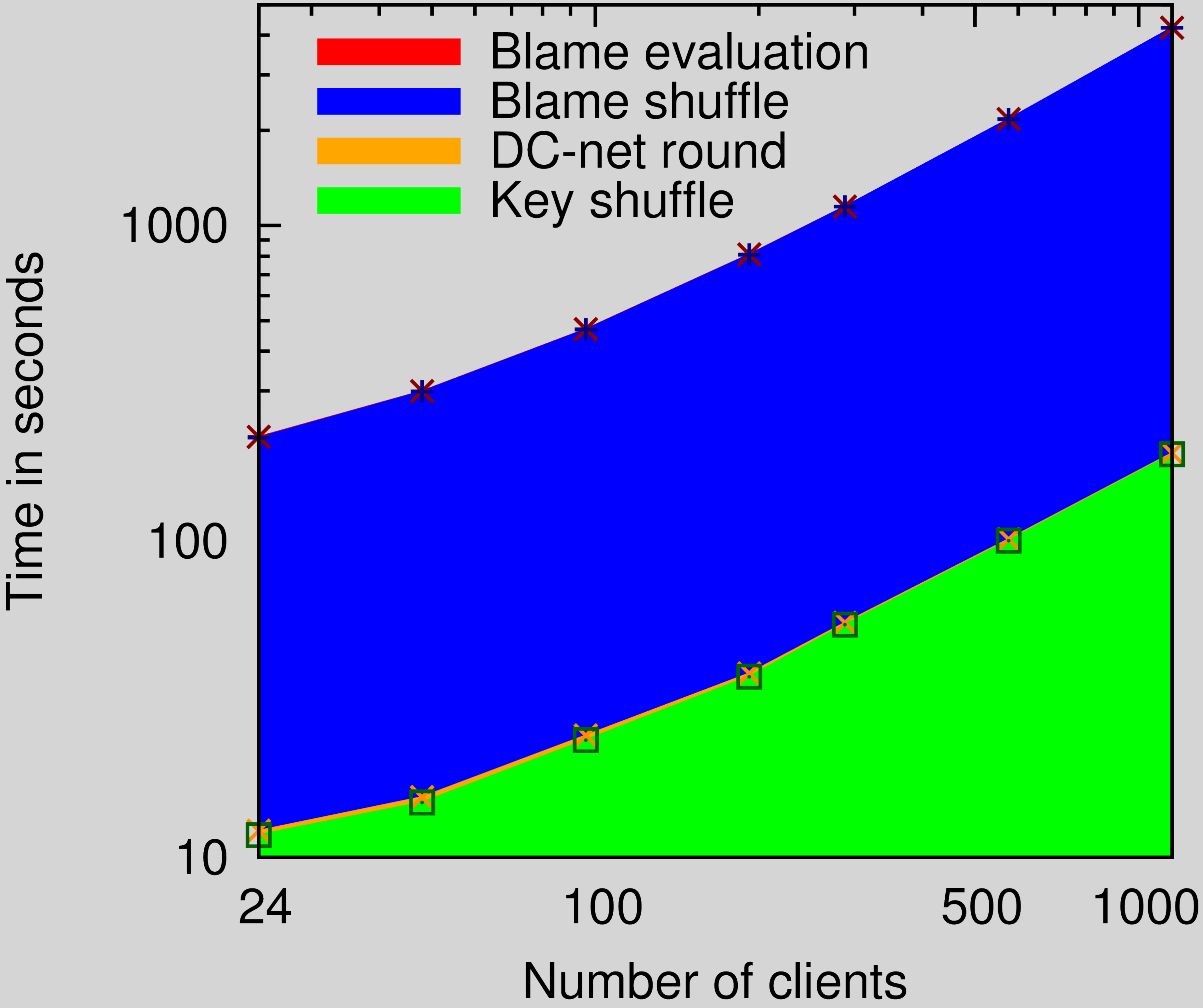
Overheads as clients finish and servers takeover

1000 clients
< 1 second

Measurements for one round in microblog with 32 servers

Credits: Wolinsky, David Isaac, et al. "Dissent in numbers: Making strong anonymity scale." *10th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 12)*. 2012.

Dissent - Evaluation



Time elapsed during a whole Dissent protocol run with 24 servers and 128 byte messages

- Key shuffle is costly but is done rarely.
- Accountability/Accusation is costly operation as it's a different shuffle and not same as key shuffle.

Credits: Wolinsky, David Isaac, et al. "Dissent in numbers: Making strong anonymity scale." *10th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 12)*. 2012.

Strengths and Weaknesses

Strengths:

- Improvement on previous Dissent Paper making it “scalable”.
- Computations can be done in parallel along with lesser communication.
- Provides participation count - clients know how many users will get the message.
- Eliminates empty slots overhead.
- Provides churn tolerance and identifies disruptors.
- Evaluation proves the goals of the system (not all scenarios though).
- Paper is well structured.

Strengths and Weaknesses

Weaknesses:

- No formal security analysis of the system.
- Data corruption recovery mechanism is missing (while transmitting messages).
- No churn tolerance for servers.
- No protection against membership intersection attacks.
- What should be efficient M vs N values?
- Possibility to leak cleartext messages when server broadcasts it to client.
- Client has to wait for 2 rounds to send a larger message.

Latest updates at
https://dedis.cs.yale.edu/dissent/

Questions?