TU Delft

Bloom Filter-Based MPSI

Weekly Progress Meeting 29 Jan 2026

Andra Alăzăroaie

Supervisor: Lilika Markatou

Daily Supervisor: Tjitske Koster

29-01-2026

Bibliography

[1] Jelle Vos, Jorrit van Assen, Tjitske Koster, Evangelia Anna Markatou, and Zekeriya Erkin. On the insecurity of bloom filter-based private set intersections. Cryptology ePrint Archive, 2024.

[2] Aslı Bay, Zekeriya Erkin, Jaap-Henk Hoepman, Simona Samardjiska, and Jelle Vos. Practical multi-party private set intersection protocols. IEEE Transactions on Information Forensics and Security, 17:1–15, 2021.

[3] Ou Ruan, Changwang Yan, Jing Zhou, and Chaohao Ai. A practical multiparty private set intersection protocol based on bloom filters for unbalanced scenarios. Applied Sciences, 13(24):13215, 2023.

[4] Ou Ruan and Chaohao Ai. An efficient multi-party private set intersection protocols based on bloom filter. In Second International Conference on Algorithms, Microchips, and Network Applications (AMNA 2023), volume 12635, pages 282–287. SPIE, 2023.

[5] Jelle Vos, Mauro Conti, and Zekeriya Erkin. Fast multi-party private set operations in the star topology from secure ands and ors. Cryptology ePrint Archive, 2022.

[6] Vladimir Kolesnikov, Naor Matania, Benny Pinkas, Mike Rosulek, and Ni Trieu. Practical multi-party private set intersection from symmetric-key techniques.In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pages 1257–1272, 2017.

[7] Alireza Kavousi, Javad Mohajeri, and Mahmoud Salmasizadeh. Efficient scalable multi-party private set intersection using oblivious prf. In International Workshop on Security and Trust Management, pages 81–99. Springer, 2021.

Progress

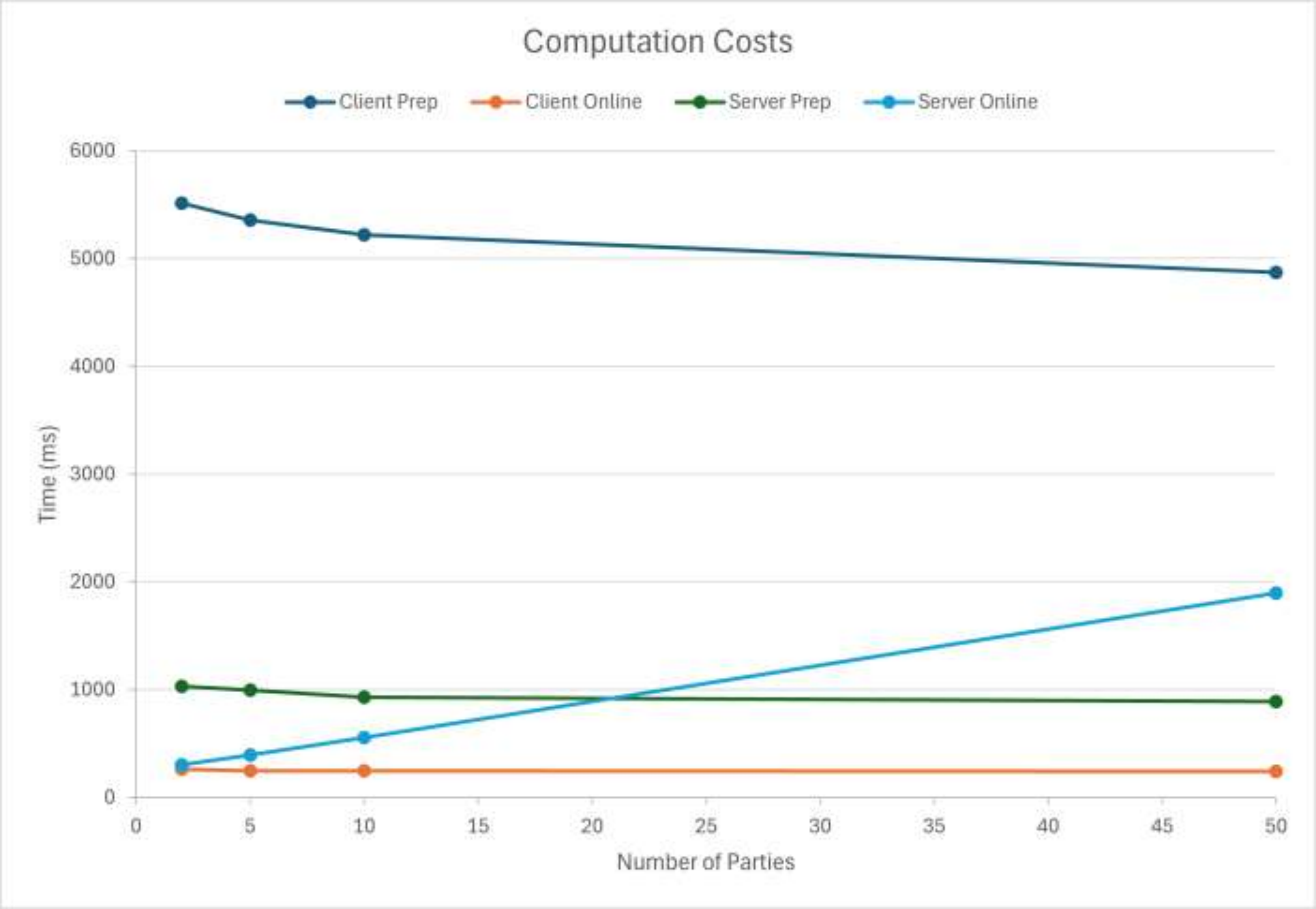
From the plan last week:

- Continue writing and experimenting.
- Mitigation OPRF:
    - Read [6] and [7], learn about oblivious PRFs.

Progress

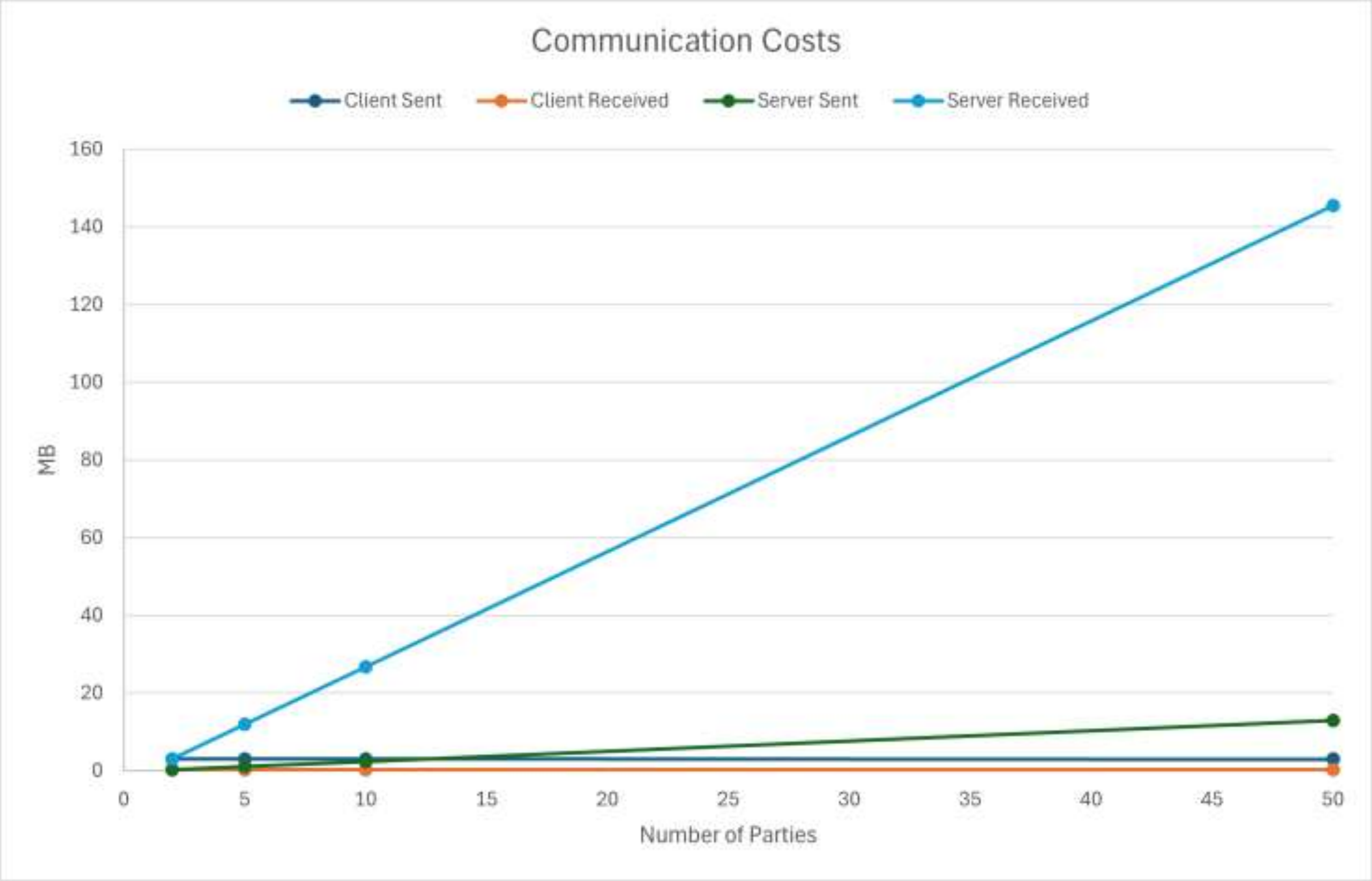
Computation on the online stage of [3]:

- Client’s online time does not increase with the number of parties
- Matches the results from the paper



Progress

Communication costs of [3]:

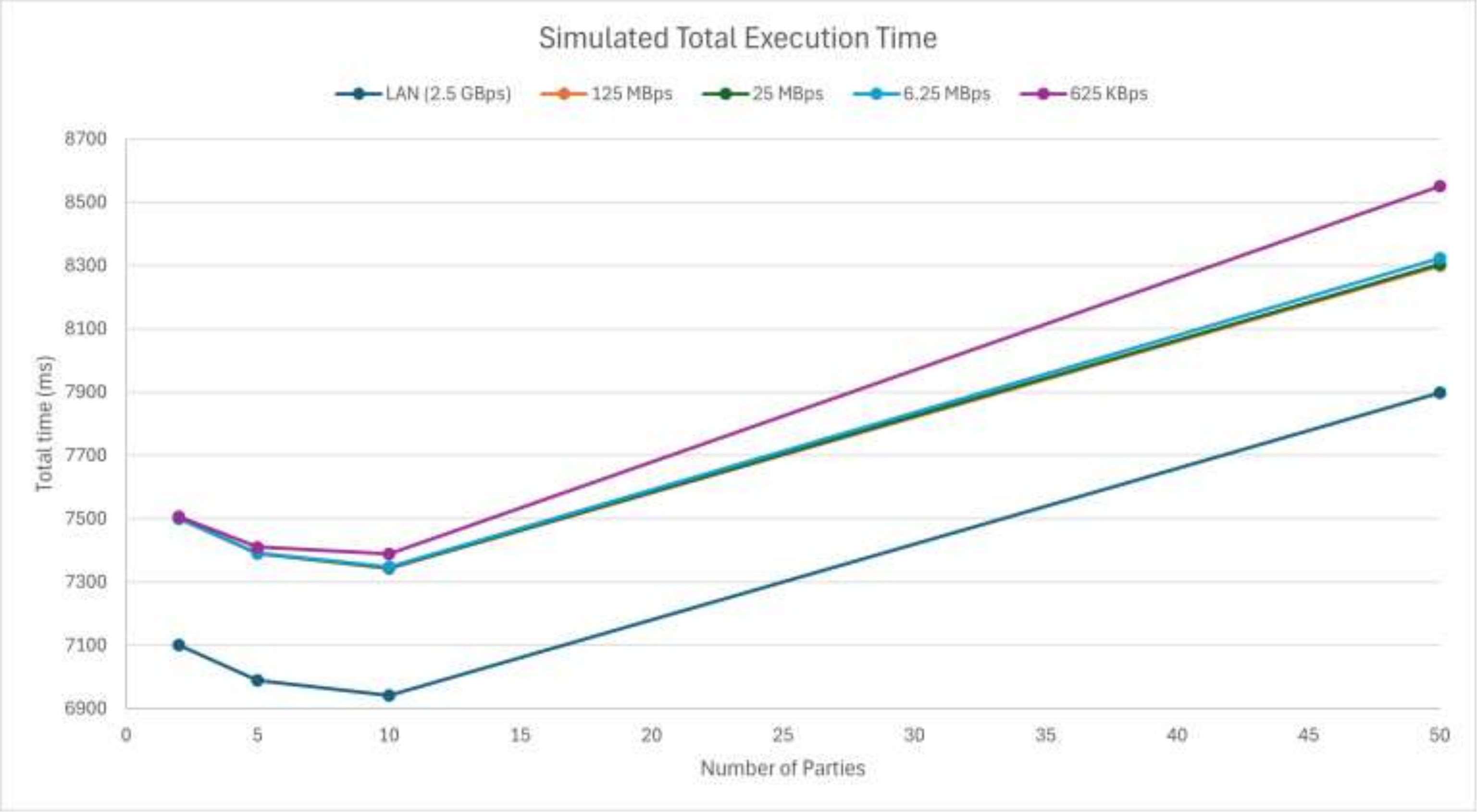


Progress

Total time of the protocol [3] calculated as:

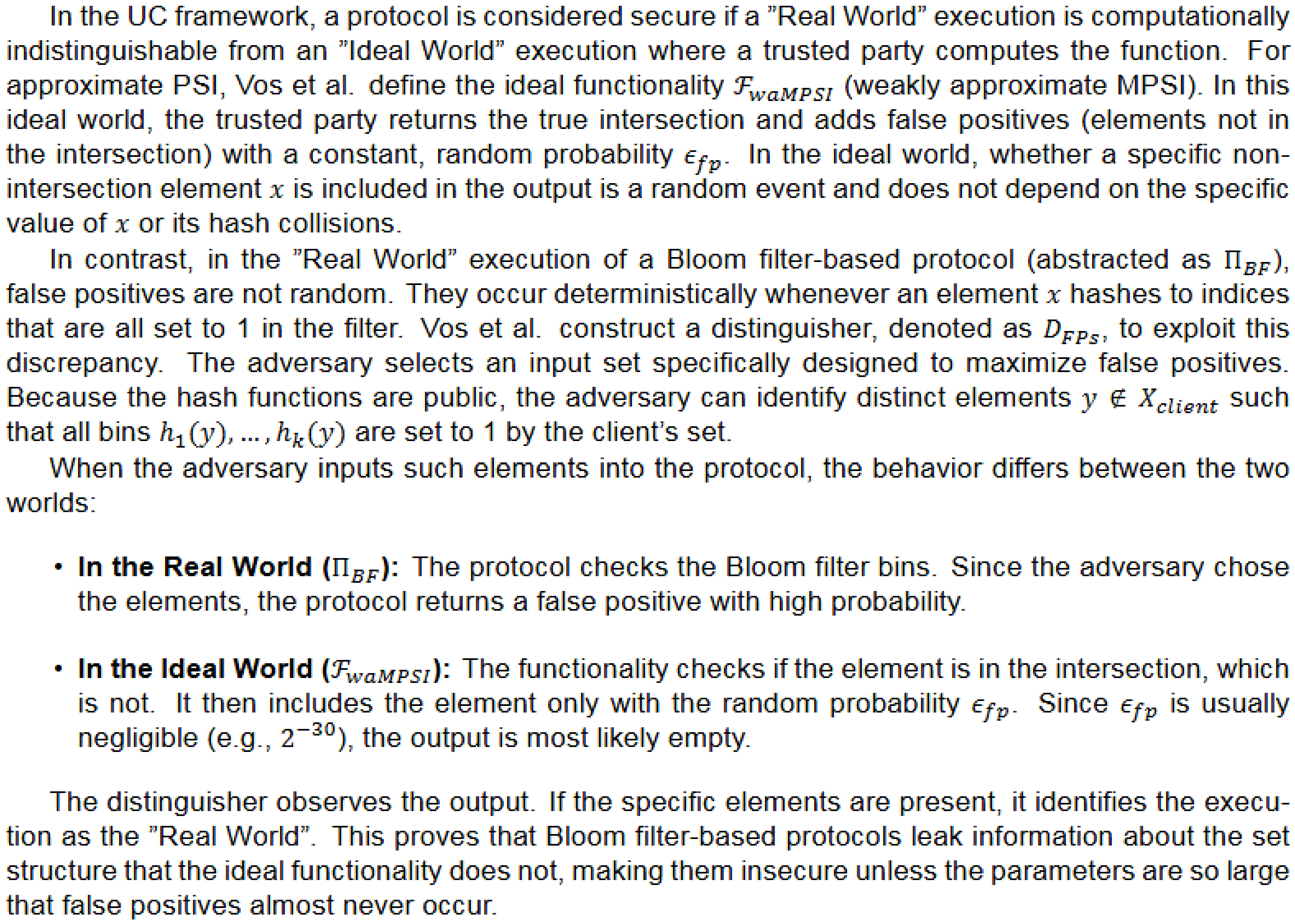
Latency + Communication time + Computation Time

- **Latency:** *number of messages * hardcoded network latency*
- **Communication time:** *Bytes sent * hardcoded bandwidth*
- **Computation time:** plotted before



Progress

- Explaining why [3] is vulnerable to the attack in [1]
    - through explaining why it reduces to the idealized behavior π_{BF} of a BF-based MPSI protocol



In the UC framework, a protocol is considered secure if a "Real World" execution is computationally indistinguishable from an "Ideal World" execution where a trusted party computes the function. For approximate PSI, Vos et al. define the ideal functionality $\mathcal{F}_{waMPSI}$ (weakly approximate MPSI). In this ideal world, the trusted party returns the true intersection and adds false positives (elements not in the intersection) with a constant, random probability ϵ_{fp}. In the ideal world, whether a specific non-intersection element x is included in the output is a random event and does not depend on the specific value of x or its hash collisions.

In contrast, in the "Real World" execution of a Bloom filter-based protocol (abstracted as Π_{BF}), false positives are not random. They occur deterministically whenever an element x hashes to indices that are all set to 1 in the filter. Vos et al. construct a distinguisher, denoted as D_{FPs}, to exploit this discrepancy. The adversary selects an input set specifically designed to maximize false positives. Because the hash functions are public, the adversary can identify distinct elements $y \notin X_{client}$ such that all bins $h_1(y), \ldots, h_k(y)$ are set to 1 by the client's set.

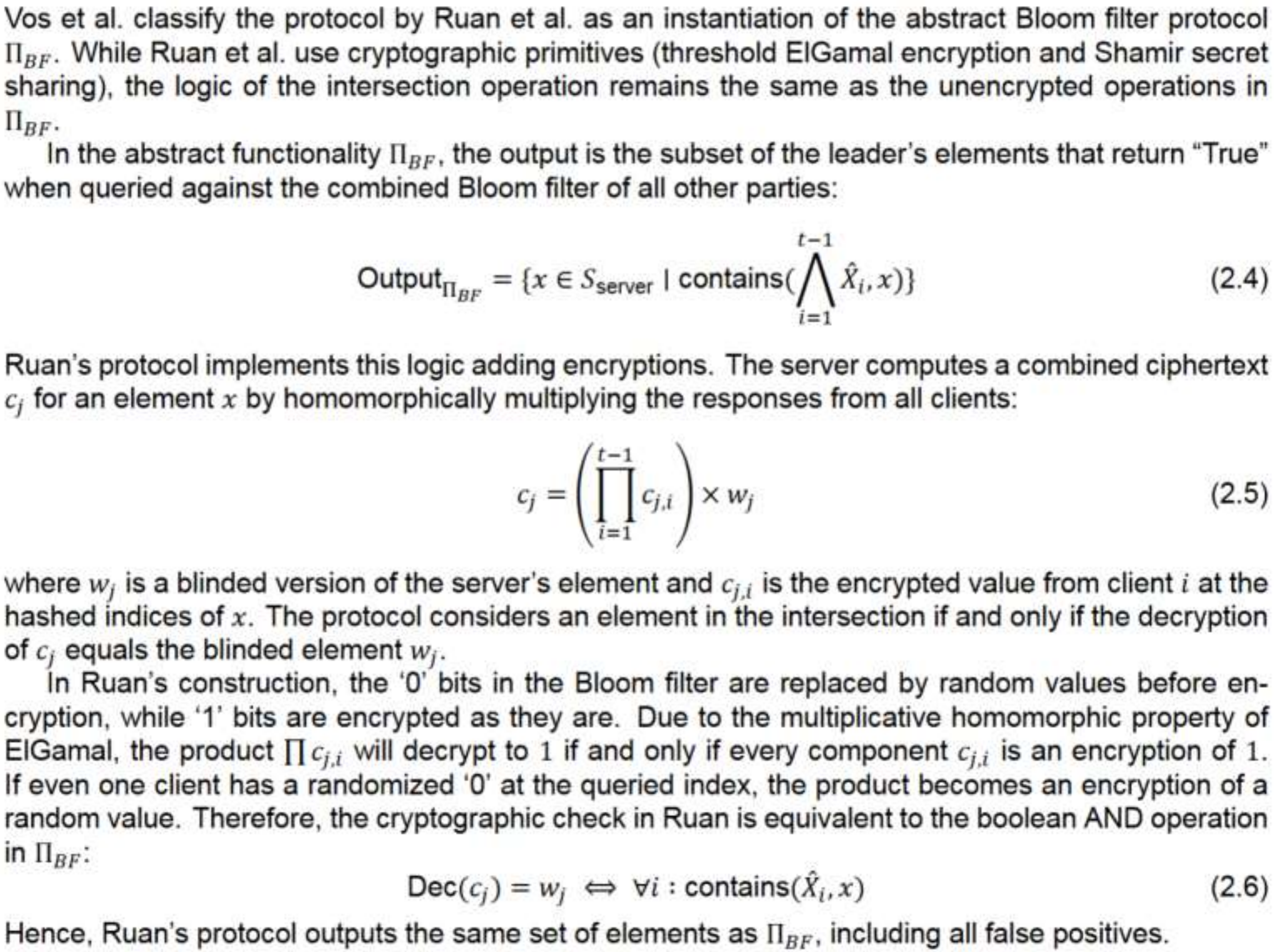
When the adversary inputs such elements into the protocol, the behavior differs between the two worlds:

- **In the Real World (Π_{BF}):** The protocol checks the Bloom filter bins. Since the adversary chose the elements, the protocol returns a false positive with high probability.
- **In the Ideal World ($\mathcal{F}_{waMPSI}$):** The functionality checks if the element is in the intersection, which is not. It then includes the element only with the random probability ϵ_{fp}. Since ϵ_{fp} is usually negligible (e.g., 2^{-30}), the output is most likely empty.

The distinguisher observes the output. If the specific elements are present, it identifies the execution as the "Real World". This proves that Bloom filter-based protocols leak information about the set structure that the ideal functionality does not, making them insecure unless the parameters are so large that false positives almost never occur.

Progress

- Explaining why [3] is vulnerable to the attack in [1]
  - through explaining why it reduces to the idealized behavior π_{BF} of a BF-based MPSI protocol



Vos et al. classify the protocol by Ruan et al. as an instantiation of the abstract Bloom filter protocol Π_{BF}. While Ruan et al. use cryptographic primitives (threshold ElGamal encryption and Shamir secret sharing), the logic of the intersection operation remains the same as the unencrypted operations in Π_{BF}.

In the abstract functionality Π_{BF}, the output is the subset of the leader's elements that return "True" when queried against the combined Bloom filter of all other parties:

$$\text{Output}_{\Pi_{BF}} = \{x \in S_{\text{server}} \mid \text{contains}(\bigwedge_{i=1}^{t-1} \hat{X}_i, x)\} \qquad (2.4)$$

Ruan's protocol implements this logic adding encryptions. The server computes a combined ciphertext c_j for an element x by homomorphically multiplying the responses from all clients:

$$c_j = \left(\prod_{i=1}^{t-1} c_{j,i}\right) \times w_j \qquad (2.5)$$

where w_j is a blinded version of the server's element and $c_{j,i}$ is the encrypted value from client i at the hashed indices of x. The protocol considers an element in the intersection if and only if the decryption of c_j equals the blinded element w_j.

In Ruan's construction, the '0' bits in the Bloom filter are replaced by random values before encryption, while '1' bits are encrypted as they are. Due to the multiplicative homomorphic property of ElGamal, the product $\prod c_{j,i}$ will decrypt to 1 if and only if every component $c_{j,i}$ is an encryption of 1. If even one client has a randomized '0' at the queried index, the product becomes an encryption of a random value. Therefore, the cryptographic check in Ruan is equivalent to the boolean AND operation in Π_{BF}:

$$\text{Dec}(c_j) = w_j \iff \forall i : \text{contains}(\hat{X}_i, x) \qquad (2.6)$$

Hence, Ruan's protocol outputs the same set of elements as Π_{BF}, including all false positives.

Challenges

- When tweaking the false positive probability (which determines the number of bins in the Bloom filter and the number of hash functions) in order to see false positives appearing in the final intersection, one issue was setting the domain size for the experiments in order to have a non-empty intersection as the number of parties grew.
- In [2]'s experiments, the FP probability is set to only 2^{-7}, while in [3]'s, it is 2^{-30}. [3]'s experiments include a comparison in computation costs with [2], showing that it performs better (since ElGamal is faster than Paillier). How do we show the mitigation if the experiments don't reveal any false positives when using 2^{-30}?
- OPRF reading from [6], [7]:
    - Both [6] and [7] use GBFs to implement OPRF
    - In [3] (the protocol I implemented), the 0 bins in the BFs are randomized before they are encrypted, as ElGamal cannot encrypt 0 – Can this be an issue for GBFs?
    - It is unclear so far how to replace the hash functions with OPRFs without changing the protocol.

Next Week

- Prepare the presentation for the 1st stage evaluation
- Continue working on the mitigations

1st Stage Evaluation Presentation Plan

- Start with the same slides from the kick-off introducing **MPSI, Bloom filters, the attack + the research question(s)**
- Go into the methodology and present the progress there
  - Describe the implemented protocol and why it is vulnerable to the attack
  - Show the graphs
  - Talk about the progress on the mitigations
- Talk about what is left to do and show the timeline

TU Delft

Thank you!

29-01-2026