A Not-So-Trival Replay Attack Against DH-PSI

Hongrui Cui*[1] and Yu Yu†[1]

[1]Department of Computer Science, Shanghai Jiao Tong University

July 18, 2020

Abstract

In this note, we present a simple yet effective inter-session replay attack against the Diffie-Hellman style private set intersection protocol (cf. [Mea86]). The attack is indistinguishable from ordinary protocol execution, and yet allows the attacker to learn the cardinality of the intersection of honest party's input sets. This kind of attack demonstrates the inadequacy of semi-honest security guarantee when facing more serious adversarial threats, and highlights the necessity for security augmentation of protocols derived from [Mea86].

1 Introduction

Private set intersection received much attention in recent years from both academic and industry communities. Due to the nature of this brief report, we refer readers to the work of Pinkas et al. [PSZ18] for a summary in this field and several follow-up papers [PRTY20, PSTY19, RR17] for more recent developments.

Albeit the rapid recent development in this field, the protocol based on Diffie-Hellman key exchange [Mea86] is still preferable due to its simplicity and communication efficiency. Nevertheless, the original protocol offers a mere passive security guarantee, despite a lack of practical active attacks in the literature. This leads to a series of endeavors into the security potentials of DH-PSI protocol, epitomized by a recent blog post by the Alibaba Gemini Lab [Ali19]. In particular, we want to answer the following question:

Is the passive security of DH-PSI an artifact of the proof techniques, or does it suffer from security inadequacy in face of malicious adversaries?

In this report, we present a replay attack that demonstrates the insecurity of DH-PSI under parallel execution, highlighting the need for security enhancement of DH-PSI when facing against malicious adversaries. Although we have not yet come up with a lightweight solution to boost DH-PSI from passive security to active one, we wish that this finding would help understand the gap and inspire subsequent investigations.

*rickfreeman@sjtu.edu.cn

†yuyuathk@gmail.com

2 A Brief Review on DH-PSI

We denote the two participants as Alice and Bob, their sets as X and Y both holding length-σ strings, the group description as $\mathbb{G}$, with generator g and order q. There is also a hash function $H : \{0,1\}^\sigma \to \{0,1\}^l$ modeled as random oracle. The protocol, as shown in Fig. 1, contains four rounds. In the first round, Alice picks a secret random exponent $a \overset{\$}{\leftarrow} \mathbb{Z}_q$, hashes all items in her input set, raises the hashed set to the power of a, and sends the results to Bob. Bob follows the same procedure with exponent $b \overset{\$}{\leftarrow} \mathbb{Z}_q$ in the second round. In the third round, Alice raises the received Bob's message to the power of a and sends them back. Bob follows the same procedure in the fourth round.

As pointed out by Meadow in [Mea86], in order to let the other party identify the intersection, Alice (resp. Bob) must keep the original order at round 3 (resp. 4).

1. Alice → Bob : $S_1 = \{H(x)^a : x \in X\}$ where $a \overset{\$}{\leftarrow} \mathbb{Z}_q$
2. Bob → Alice : $S_2 = \{H(y)^b : x \in Y\}$ where $b \overset{\$}{\leftarrow} \mathbb{Z}_q$
3. Alice → Bob : $S_3 = \{s_2^a : s_2 \in S_2\}$
4. Bob → Alice : $S_4 = \{s_1^b : s_1 \in S_1\}$

Figure 1: DH-PSI Protocol as in [Mea86]

The semi-honest security of this protocol in the standard model (cf. [Lin16]) follows from the Decisional Diffie-Hellman assumption of the group $\mathbb{G}$ plus a standard hybrid argument. But in the next section, we point out in the concurrent setting against an active adversary, such protocol leaks information not simulatable in the ideal world.

3 The Replay Attack

Consider a concurrent execution setting of the protocol in Fig. 1. Alice performs two intersections with two parties both controlled by Bob. Let sid_1 and sid_2 denotes these two sessions. We may assume the adversary is "rushing"—it always sends messages after the honest party does. Let X_1 and X_2 (resp. a_1 and a_2) denotes the two input sets (resp. random number) of Alice in sid_1 and sid_2.

As shown in Fig. 2, Bob first samples two random numbers $r_1, r_2 \overset{\$}{\leftarrow} \mathbb{Z}_q$, and then invokes the protocol execution with Alice, after which he receives $S_1^1 = \{H(x)^{a_1} : x \in X_1\}$ and $S_1^2 = \{H(x)^{a_2} : x \in X_2\}$. He then computes $S_2^1 = \{s^{r_1} : s \in S_1^2\}$ and $S_2^2 = \{s^{r_2} : s \in S_1^1\}$ and sends to Alice. In the third round, Bob receives $S_3^1 = \{s^{a_1} : s \in S_2^1\}$ and $S_3^2 = \{s^{a_2} : s \in S_2^2\}$ and computes $\tilde{X}_1 = \{s^{\frac{1}{r_2}} : s \in S_3^2\} = \{H(x)^{a_1 a_2} : x \in X_2\}$ and $\tilde{X}_2 = \{s^{\frac{1}{r_1}} : s \in S_3^1\} = \{H(x)^{a_1 a_2} : x \in X_1\}$. Now by computing $|\tilde{X}_1 \cap \tilde{X}_2|$ Bob can learn $|X_1 \cap X_2|$ without having prior information about X_1 and X_2.

1. Alice → Bob : $S_1^1 = \{H(x)^{a_1} : x \in X_1\}$ where $a_1 \overset{\$}{\leftarrow} \mathbb{Z}_q$
2. Alice → Bob : $S_1^2 = \{H(x)^{a_2} : x \in X_2\}$ where $a_2 \overset{\$}{\leftarrow} \mathbb{Z}_q$
3. Bob → Alice : $S_2^1 = \{s^{r_1} : s \in S_1^1\}$ where $r_1 \overset{\$}{\leftarrow} \mathbb{Z}_q$
4. Bob → Alice : $S_2^2 = \{s^{r_2} : s \in S_1^2\}$ where $r_2 \overset{\$}{\leftarrow} \mathbb{Z}_q$
5. Alice → Bob : $S_3^1 = \{s^{a_1} : s \in S_2^1\}$
6. Alice → Bob : $S_3^2 = \{s^{a_2} : s \in S_2^2\}$
7. Bob learns $\tilde{X}_2 = \{s^{1/r_1} : s \in S_3^1\} = \{H(x)^{a_2 r_1 a_1 1/r_1}\} = \{H(x)^{a_1 a_2} : x \in X_2\}$ and $\tilde{X}_1 = \{s^{1/r_2} : s \in S_3^2\} = \{H(x)^{a_1 r_2 a_2 1/r_2}\} = \{H(x)^{a_1 a_2} : x \in X_1\}$. $|X_1 \cap X_2| = |\tilde{X}_1 \cap \tilde{X}_2|$

Figure 2: The Concurrent Replay Attack

References

[Ali19] Alibaba Gemini Lab. Challenge in breaking the ECDH PSI, 2019. https://alibaba-gemini-lab.github.io/docs/blog/psi_challenge/ , retrieved on July 17, 2020.

[Lin16] Yehuda Lindell. How to simulate it - a tutorial on the simulation proof technique. Cryptology ePrint Archive, Report 2016/046, 2016. https://eprint.iacr.org/2016/046.

[Mea86] C. Meadows. A more efficient cryptographic matchmaking protocol for use in the absence of a continuously available third party. In *1986 IEEE Symposium on Security and Privacy*, pages 134–134, 1986.

[PRTY20] Benny Pinkas, Mike Rosulek, Ni Trieu, and Avishay Yanai. Psi from paxos: Fast, malicious private set intersection. Cryptology ePrint Archive, Report 2020/193, 2020. https://eprint.iacr.org/2020/193.

[PSTY19] Benny Pinkas, Thomas Schneider, Oleksandr Tkachenko, and Avishay Yanai. Efficient circuit-based psi with linear communication. Cryptology ePrint Archive, Report 2019/241, 2019. https://eprint.iacr.org/2019/241.

[PSZ18] Benny Pinkas, Thomas Schneider, and Michael Zohner. Scalable private set intersection based on ot extension. *ACM Trans. Priv. Secur.*, 21(2), January 2018.

[RR17] Peter Rindal and Mike Rosulek. Malicious-secure private set intersection via dual execution. Cryptology ePrint Archive, Report 2017/769, 2017. https://eprint.iacr.org/2017/769.