Revisiting the Secret Hiding Assumption Used in Verifiable (Outsourced) Computation

Liang Zhao(✉)

College of Cybersecurity, Sichuan University, Chengdu, China
zhaoliangjapan@scu.edu.cn

Abstract. Privacy-preserving Verifiable (outsourced) Computation (PVC) is a hopeful primitive that enables a resource-constrained client to outsource expensive and sensitive workloads to powerful but possibly untrusted servers and to verify the correctness of the returned results. Specifically, the privacy property is of significance for this type of primitive. Then, how to provide the privacy property has become a central interest of many researchers. At ACM-ASIACCS 2010, Atallah and Frikken introduced a new hardness assumption called the Secret Hiding assumption (SH), which includes the Weak SH assumption (WSH) and Strong SH assumption (SSH). Moreover, for the outsourcing of the multiplication of large-scale matrices, the authors constructed two concrete PVC protocols whose privacy is based on the decisional-WSH assumption and decisional-SSH assumption, respectively.

Until our work, to the best of our knowledge, there is no paper that precisely explored the hardnesses of the WSH assumption and SSH assumption. Thus, in this paper, we first propose an analysis method, using the rank distribution of the matrix as the basic strategy, to evaluate the hardnesses of two problems corresponding to the decisional-WSH assumption and decisional-SSH assumption. Unfortunately, our analysis can efficiently break the decisional-WSH assumption and decisional-SSH assumption for a wide range of parameters with overwhelming probability. Then we employ the idea of the above analysis for breaking the SH assumption to similarly break the privacy of Atallah and Frikken's PVC protocols. The results show that the adversary's advantages are non-negligible. Finally, we present some detailed experimental results to support our theoretical argument.

Keywords: Privacy-preserving verifiable (outsourced) computation · Indistinguishability · Rank · Linear relation

1 Introduction

1.1 Background

Privacy-preserving Verifiable (outsourced) Computation (PVC), characterized by four properties [8], i.e., correctness, security, privacy and efficiency, has

M. Matsui (Ed.): CT-RSA 2019, LNCS 11405, pp. 514–534, 2019.
https://doi.org/10.1007/978-3-030-12612-4_26

attracted many researchers from the cryptography and information security community. Various protocols [2–6,8,9,15,16] have been proposed to solve the problems related to the outsourcing of computations on general and specific functions. In particular, for those protocols, privacy is a significant property guaranteeing that the information hidden in the data structure related to the input and output of the outsourced computation cannot be revealed to any unauthorised entity who has access to the data. The analysis of privacy is based on the notion of indistinguishability (see [7] about this notion). This implies that the input and output data are semantically hidden to the unauthorised entity.

To construct the PVC protocols for outsourcing expensive linear algebraic computations, at ACM-ASIACCS 2010, Atallah and Frikken [2] introduced a new hardness assumption called the Secret Hiding assumption (SH) (see Sect. 2.1). Specifically, the authors presented two concrete versions, i.e., the Weak SH assumption (WSH) and the Strong SH assumption (SSH), respectively. The WSH assumption, informally, states that it is hard to distinguish (with knowing the prime p) between the uniform distribution over $\mathbb{Z}_p^{n\times m}$ and the distribution $\chi(p)^{n\times m}$ that outputs the matrix with $\lambda+1$ rows $[\Sigma_{j=1}^{\lambda} a_{1,j}\cdot k_r^j \ldots \Sigma_{j=1}^{\lambda} a_{m,j}\cdot k_r^j]$ and λ rows uniformly distributed over $\mathbb{Z}_p^m$, where $n=2\cdot\lambda+1$, where $\lambda\in\mathbb{N}_+$, $m\in\mathbb{N}_+$ (e.g., $m=2\cdot\lambda+1$), $\forall r\in\{1,\ldots,\lambda+1\}k_r$ is chosen from $\mathbb{Z}_p^*$ uniformly at random, and $\forall i\in\{1,\ldots,m\}, j\in\{1,\ldots,\lambda\}a_{i,j}$ is chosen from $\mathbb{Z}_p$ uniformly at random. The SSH assumption is similar to the WSH assumption, and it states that the uniform distribution over $\mathbb{Z}_p^{n\times m}$ is computationally indistinguishable from the distribution $\chi(p)^{n\times m}$ that outputs the matrix with $\lambda+e+1$ rows $[\Sigma_{j=1}^{\lambda} a_{1,j}\cdot k_r^j \ldots \Sigma_{j=1}^{\lambda} a_{m,j}\cdot k_r^j]$ and $\lambda+e+1$ rows uniformly distributed over $\mathbb{Z}_p^m$, where $n=2\cdot\lambda+2\cdot e+2$, where $e\in\mathbb{N}_+$ (e.g., $e=\lambda$). To validate the plausible hardnesses of the above assumptions, Atallah and Frikken provided a proof to show that the SH assumption (i.e., the WSH assumption) implies the existence of one-way functions. This means that proving the SH assumption is at least as hard as proving P $\neq$ NP [2].

Based on the SH assumption, Atallah and Frikken [2] proposed two concrete PVC protocols for efficiently outsourcing the multiplication of large-scale matrices (see Sect. 2.3). Specifically, these two provably private protocols can be seen as the ingenious extensions of Shamir's secret sharing [14], and they are always regarded as the typical work in the PVC community. Atallah and Frikken first introduced a protocol based on the WSH assumption under the two non-colluding servers model (denoted by $\mathsf{AF\text{-}PVC}_{two}$). In this warm-up protocol, a client needs to generate λ and $2\cdot\lambda+1$ pairs of matrices for each server, respectively. The servers perform $O(\lambda)$ matrix multiplications. Then, the authors developed a protocol based on the SSH assumption under the single server model (denoted by $\mathsf{AF\text{-}PVC}_{single}$). In this main protocol, a client must create $4\cdot\lambda+2$ pairs of matrices for the single server, and this server also perform $O(\lambda)$ matrix multiplications. Furthermore, the authors provided a method to make the protocol under the single server model hold the security (i.e., the property related to the integrity verification). Of course, there exist some other researchers who are also interested in the SH assumption. For example, Laud and Pankova [12] tried

to construct a PVC protocol for outsourcing solutions of linear programming problems based on the SSH assumption.

1.2 Our Contributions

Atallah and Frikken have given some theoretical consequences related to the SH assumption, but whether the SH problem corresponding to the assumption is a hard problem is still a worthwhile research area, particularly when the assumption is proposed for applications in the concrete real-world scenarios. In this paper, we present some rigorous analyses, targeting the SH problem and the Atallah-Frikken PVC Protocols for matrix multiplication, as follows:

- We present the decisional and search variants of the SH problem in Sect. 2, which are more standard problems when compared with the originals.
- We propose an analysis, discussed in Sect. 3, to break the decisional variant of the SH assumption (including the WSH assumption and SSH assumption) in a wide range of parameters. Our precise analysis focusing on evaluating the rank of a matrix shows that the decisional-SH problem (including the decisional-WSH problem and decisional-SSH problem) is not a hard problem, and the given SH distribution $\chi(p)^{n\times m}$ can be distinguished from the uniform distribution over $\mathbb{Z}_p^{n\times m}$ with overwhelming probability.
- We invoke the idea of the analysis for solving the decisional-SH problem to undermine the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$ in Sect. 3. Our analyses running in polynomial-time take advantage of the distinctions between the rank distributions of two types of given ciphertext matrices (see Theorems 7 and 8 for the two types of ciphertext matrices). The success probabilities of the analyses are close to 1, which shows that neither of those protocols is private against passive eavesdropping (i.e., a ciphertext-only attack (COA) (see Definition 3)) and also a chosen-plaintext attack (CPA) (see Definition 4).
- We implement the simulation experiments on our theoretical analyses for solving the decisional-SH problem and breaking the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$. The experimental results, presented in Sect. 4, confirm our analyses, which demonstrates that the decisional-SH problem is not a hard problem for a wide range of parameters, and $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$ are not semantically private PVC protocols.

1.3 Organization of the Rest of the Paper

The remainder of the paper is organized as follows. Section 2 introduces the decisional and search versions of the SH assumption (including the WSH assumption and SSH assumption), the Atallah and Frikken's theoretical exploration on the SH assumption, the PVC protocols $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$ and the formal definition of privacy. Section 3 describes the adversary's strategy and the detailed theoretical analyses for solving the decisional-SH problem and breaking the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$. Section 4 gives some detailed

experimental verifications about our theoretical analyses in Sect. 3. The paper is concluded in Sect. 5 with a direction for future research.

Notation: Throughout the paper, we generally do math modulo p for some prime p. We denote by bold lower-case letters vectors over $\mathbb{Z}_p^n$ for $n \geq 2$, and by bold upper-case letters matrices over $\mathbb{Z}_p^{n \times m}$ for n, $m \geq 2$, where $\mathbb{Z}_p$ is a finite field of size p. We refer a set of elements from a row or a column of a matrix to as a vector. We denote by $x_{i,j}$ the individual element in the i^{th} row and j^{th} column of a matrix $\mathbf{X}$. For any integer n, we denote the set $\{1, \ldots, n\}$ by $[n]$. We denote a security parameter by $\lambda \in \mathbb{N}_+$. We denote the transpose of x by x^T, the rank of a matrix $\mathbf{X}$ by $\mathsf{rank}(\mathbf{X})$, and the minimum of two values by $\mathsf{min}(\cdot, \cdot)$. We denote the class of polynomial functions in λ by $\mathsf{poly}(\lambda)$, and some unspecified negligible function in λ by $\mathsf{negl}(\lambda)$. We use $x \xleftarrow{\$} \Psi$ to denote the operation of uniformly sampling an element x from a finite set Ψ. For some probability distribution χ, $x \leftarrow \chi$ refers to sampling x according to χ.

2 Preliminaries

In this section, we recall the SH assumption, $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$ proposed by Atallah and Frikken at ACM-ASIACCS 2010 [2]. We also present the formal definition of privacy for the PVC protocol.

2.1 The SH Assumption

We first describe the probability distribution $\chi(p)^{n \times m}$ that results from the following steps, where $n \in \{2 \cdot \lambda + 1, 2 \cdot \lambda + 2 \cdot e + 2\}$ and $m = \mathsf{poly}(\lambda) \geq 2$, where $e \in \mathbb{N}_+$.

1. Choose a uniformly random matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_p^{m \times \lambda}$, where each element $a_{i,j} \xleftarrow{\$} \mathbb{Z}_p$ for $i \in [m]$ and $j \in [\lambda]$. Choose $\ell = \lambda + 1$ (resp. $\ell = \lambda + e + 1$) distinct values $k_1, \ldots, k_\ell \xleftarrow{\$} \mathbb{Z}_p^*$.
2. For $r \in [\lambda + 1]$ (resp. $r \in [\lambda + e + 1]$), compute $\mathbf{d}_r = (\mathbf{A} \cdot \mathbf{k}_r)^T$, where $\mathbf{k}_r = [k_r k_r^2 \ldots k_r^\lambda]^T$. Obtain $\ell = \lambda + 1$ (resp. $\ell = \lambda + e + 1$) row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_\ell$, where $\mathbf{d}_r = [\Sigma_{j=1}^{\lambda} a_{1,j} \cdot k_r^j \ldots \Sigma_{j=1}^{\lambda} a_{m,j} \cdot k_r^j]$ for $r \in [\lambda + 1]$ (resp. $r \in [\lambda + e + 1]$).
3. For $r \in [\lambda]$ (resp. $r \in [\lambda + e + 1]$), choose $\mathbf{u}_r \xleftarrow{\$} \mathbb{Z}_p^m$. Obtain $\tau = \lambda$ (resp. $\tau = \lambda + e + 1$) row vectors $\mathbf{u}_1, \ldots, \mathbf{u}_\tau$.
4. Combine the $\ell = \lambda + 1$ (resp. $\ell = \lambda + e + 1$) row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_\ell$ with the $\tau = \lambda$ (resp. $\tau = \lambda + e + 1$) row vectors $\mathbf{u}_1, \ldots, \mathbf{u}_\tau$ to generate an $n \times m$ matrix $\mathbf{R}$. Choose a random permutation of the set $[n]$ to permute the rows of $\mathbf{R}$. The permuted matrix is the final matrix.

Then, we present the WSH problem and SSH problem as follows:

Definition 1 (WSH Problem). *Let* $n = 2 \cdot \lambda + 1$ *and* $m = \mathsf{poly}(\lambda) \geq 2$. *The WSH distribution* $\chi(p)^{n \times m}$ *for a given prime* p *is the set of the permuted matrices, where each matrix includes* $\lambda + 1$ *row vectors* $\boldsymbol{d}_1, \ldots, \boldsymbol{d}_{\lambda+1}$ *and* λ *row vectors* $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_\lambda$.

- *The decisional-WSH problem is: For some fixed prime* p *and given arbitrarily many samples (i.e., a polynomial number of samples) from* $\mathbb{Z}_p^{n \times m}$, *to computationally distinguish whether these samples are distributed uniformly or whether they are distributed as* $\chi(p)^{n \times m}$.
- *The search-WSH problem is: For some fixed prime* p *and given* n *samples from the distribution* $\chi(p)^m$ *(i.e., a sample from* $\chi(p)^{n \times m}$*), to find* $k_1, \ldots, k_{\lambda+1}$ *(or* $\boldsymbol{A}$*).*

Definition 2 (SSH Problem). *Let* $n = 2 \cdot \lambda + 2 \cdot e + 2$ *and* $m = \mathsf{poly}(\lambda) \geq 2$, *where* $e \in \mathbb{N}_+$. *The SSH distribution* $\chi(p)^{n \times m}$ *for a given prime* p *is the set of the permuted matrices, where each matrix includes* $\lambda + e + 1$ *row vectors* $\boldsymbol{d}_1, \ldots, \boldsymbol{d}_{\lambda+e+1}$ *and* $\lambda + e + 1$ *row vectors* $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{\lambda+e+1}$.

- *The decisional-SSH problem is: The description of this problem is the same as that of the decisional version in Definition 1.*
- *The search-SSH problem is: The description of this problem is the same as that of the search version in Definition 1. The aim is to find* $k_1, \ldots, k_{\lambda+e+1}$ *(or* $\boldsymbol{A}$*).*

According to Atallah and Frikken's opinion, the WSH assumption denotes that no polynomial-time adversary solve the decisional and search WSH problem, and the SSH assumption means that no polynomial-time adversary can solve the decisional and search SSH problem. Specifically, the decisional-WSH assumption and the decisional-SSH assumption state that the distribution $\chi(p)^{n \times m}$ is computationally indistinguishable from the uniform distribution over $\mathbb{Z}_p^{n \times m}$ for $n \in \{2 \cdot \lambda + 1, 2 \cdot \lambda + 2 \cdot e + 2\}$. Then, this implies that, for any polynomial-time adversary $\mathcal{A}$, we have

$$\mathsf{Adv}_{\mathcal{A},\mathrm{SH}}(p, n, m) \stackrel{\mathrm{def}}{=} \left|\mathsf{Suc}_{\mathcal{A},\mathrm{SH}}(p, n, m) - \tfrac{1}{2}\right| \leq \mathsf{negl}(\lambda), \tag{1}$$

where SH is either the WSH distribution or the SSH distribution, $\mathsf{Suc}_{\mathcal{A},\mathrm{SH}}(p, n, m)$ denotes the probability of $\mathcal{A}$'s successful guess employing some adversary's strategy for the distribution of the sample $\mathbf{X}$ from $\mathbb{Z}_p^{n \times m}$, and $\mathsf{Adv}_{\mathcal{A},\mathrm{SH}}(p, n, m)$ denotes the advantage of $\mathcal{A}$'s guess for the distribution. Note that, $\mathcal{A}$'s guess for the sample $\mathbf{X} \in \mathbb{Z}_p^{n \times m}$ can be based on the following experiment:

1. $b \stackrel{\$}{\leftarrow} \{0, 1\}$.
2. If $b = 1$ then $\mathbf{X} \leftarrow \chi(p)^{n \times m}$ else $\mathbf{X} \stackrel{\$}{\leftarrow} \mathbb{Z}_p^{n \times m}$.
3. If $\mathcal{A}(\mathbf{X}) = b$ then $\mathcal{A}$ wins else $\mathcal{A}$ loses.

2.2 Atallah-Frikken Theorems Related to the SH Assumption

For the above WSH assumption and SSH assumption, Atallah and Frikken introduced some associated consequences below.

Lemma 1 ([2], Lemma 1). *Given a set of ℓ special row vectors $\boldsymbol{d}_1, \ldots, \boldsymbol{d}_\ell$, where $\ell < \lambda + 1$, this set of row vectors is distributed identically to a set of ℓ uniformly random row vectors from $\mathbb{Z}_p^m$.*

Theorem 1 ([2], Corollary 2). *Consider a $(\lambda + 1) \times m$ matrix that includes $\lambda + 1$ randomly permuted rows consisting of ℓ special row vectors $\boldsymbol{d}_1, \ldots, \boldsymbol{d}_\ell$ and $\lambda + 1 - \ell$ uniformly random row vectors from $\mathbb{Z}_p^m$, where $\ell < \lambda + 1$. This type of matrix is distributed identically to the uniformly sampled matrix from $\mathbb{Z}_p^{(\lambda+1)\times m}$.*

Theorem 2 ([2], Theorem 6). *Consider an $n \times m$ matrix sampled from $\chi(p)^{n\times m}$, where $n = 2 \cdot \lambda + 2 \cdot e + 2$. Choose a set of $\lambda + 1$ row vectors from this matrix uniformly at random. The probability that all the $\lambda + 1$ row vectors come from the $\lambda + e + 1$ special row vectors $\boldsymbol{d}_1, \ldots, \boldsymbol{d}_{\lambda+e+1}$ is negligible in λ.*

In [2], Atallah and Frikken did not prove the WSH assumption and SSH assumption from first principles, but the authors confirmed the hardness of the WSH problem and proposed the following theorem.

Theorem 3 ([2], Theorem 9). *Assume that the decisional-WSH assumption holds, the function that outputs an $n \times m$ matrix by invoking the generation steps of the distribution $\chi(p)^{n\times m}$ is a one-way function, where $n = 2 \cdot \lambda + 1$.*

Actually, from Theorem 3, the difficulty for distinguishing between the uniform distribution over $\mathbb{Z}_p^{n\times m}$ and the distribution $\chi(p)^{n\times m}$ shows the lower bound of the difficulty for finding $k_1, \ldots, k_{\lambda+1}$ (or $\mathbf{A}$). This means that the hardness of the decisional-WSH problem implies the hardness of the search-WSH problem. We refer to [2] for more details.

2.3 Atallah-Frikken PVC Protocols for Matrix Multiplication

Since decisional version is more handy for applications, Atallah and Frikken proposed two PVC protocols $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$ based on the plausible hardnesses of the decisional-WSH problem and decisional-SSH problem, respectively. Specifically, these protocols consist of a tuple of Probabilistic Polynomial-Time (PPT) algorithms $\mathsf{PVC} = (\mathsf{KeyGen}, \mathsf{ProbGen}, \mathsf{Compute}, \mathsf{ResuGen})$, where KeyGen is a private-key generation algorithm, $\mathsf{ProbGen}$ a problem generation algorithm that produces some ciphertext inputs for an outsourced function, $\mathsf{Compute}$ a function computation algorithm that is run by the server to produce some ciphertext outputs of the outsourced function, and $\mathsf{ResuGen}$ a result generation algorithm that produces the real output. The details of these two protocols are as follows:

The Two-Server Case: Given a security parameter λ, the matrix size $v = \mathsf{poly}(\lambda)$, the size of the message space $p = \mathsf{poly}(\lambda)$ and the degree of a polynomial

$h = \lambda$. For two $v \times v$ matrices $\mathbf{M}_1, \mathbf{M}_2 \in \mathbb{Z}_p^{v \times v}$, a quadruple of PPT algorithms $\mathsf{AF\text{-}PVC}_{two}$ is defined by

1. $\mathsf{AFT.KeyGen}(1^\lambda)$: Choose a uniformly random matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_p^{2 \cdot v^2 \times h}$, $2 \cdot \lambda + 1$ distinct values $k_1, \ldots, k_{2 \cdot \lambda + 1} \xleftarrow{\$} \mathbb{Z}_p^*$ and a random permutation θ of the set $[2 \cdot \lambda + 1]$. Output a fresh key $\mathsf{sk} = (\mathbf{A}, \{k_1, \ldots, k_{2 \cdot \lambda + 1}\}, \theta)$.
2. $\mathsf{AFT.ProbGen}(\mathsf{sk}, \mathbf{M}_1, \mathbf{M}_2)$: Run $\mathbf{A} \cdot \mathbf{k}$ to obtain a vector $\mathbf{d}$ that involves $2 \cdot v^2$ h-degree polynomials, where $\mathbf{k} = [k\ k^2 \ldots k^h]^T$, where k is an indeterminate. Use these h-degree polynomials to mask each element of $\mathbf{M}_1$ and $\mathbf{M}_2$, and generate two ciphertexts $\mathbf{C}_1$ and $\mathbf{C}_2$. Specifically, $\forall i,\ j \in [v]$, $i' \in [2 \cdot v^2]$ $c_{i,j} = \Sigma_{s=1}^{h} a_{i',s} \cdot k^s + m_{i,j}$. For $r \in [2 \cdot \lambda + 1]$, let $k = k_r$ and compute $\mathbf{C}_1(k_r)$ and $\mathbf{C}_2(k_r)$. This implies that $c_{i,j}(k_r) = \Sigma_{s=1}^{h} a_{i',s} \cdot k_r^s + m_{i,j}$. Choose $2 \cdot \lambda$ uniformly random matrices $\mathbf{B}_1, \ldots, \mathbf{B}_{2 \cdot \lambda} \xleftarrow{\$} \mathbb{Z}_p^{v \times v}$ and create λ pairs $(\mathbf{B}_1, \mathbf{B}_2), \ldots, (\mathbf{B}_{2 \cdot \lambda - 1}, \mathbf{B}_{2 \cdot \lambda})$. A client sends a set of matrix pairs $U^{(1)} = \{(\mathbf{C}_1(k_1), \mathbf{C}_2(k_1)), \ldots, (\mathbf{C}_1(k_\lambda), \mathbf{C}_2(k_\lambda))\}$ to the first server. Moreover, the client permutes the $2 \cdot \lambda + 1$ matrix pairs of the set $U^{(2)} = \{(\mathbf{C}_1(k_{\lambda+1}), \mathbf{C}_2(k_{\lambda+1})), \ldots, (\mathbf{C}_1(k_{2 \cdot \lambda + 1}), \mathbf{C}_2(k_{2 \cdot \lambda + 1})), (\mathbf{B}_1, \mathbf{B}_2), \ldots, (\mathbf{B}_{2 \cdot \lambda - 1}, \mathbf{B}_{2 \cdot \lambda})\}$ using θ, and sends the permuted set $U^{(2)}$ to the second server.
3. $\mathsf{AFT.Compute}(U^{(1)}, U^{(2)})$: The products of all matrix pairs in $U^{(1)}$ and $U^{(2)}$ are computed by those two servers and put in two sets $Q^{(1)}$ and $Q^{(2)}$, respectively. These two sets $Q^{(1)}$ and $Q^{(2)}$ are sent back to the client.
4. $\mathsf{AFT.ResuGen}(\mathsf{sk}, Q^{(1)}, Q^{(2)})$: Based on θ, choose some matrices from $Q^{(1)}$ and $Q^{(2)}$, which correspond to $\mathbf{M}_1$ and $\mathbf{M}_2$. Interpolate these matrices to find the real result of $\mathbf{M}_1 \cdot \mathbf{M}_2$.

The Single-Server Case: Given a security parameter λ, the matrix size $v = \mathsf{poly}(\lambda)$, the size of the message space $p = \mathsf{poly}(\lambda)$ and the degree of a polynomial $h = \lambda$. For two $v \times v$ matrices $\mathbf{M}_1, \mathbf{M}_2 \in \mathbb{Z}_p^{v \times v}$, a quadruple of PPT algorithms $\mathsf{AF\text{-}PVC}_{single}$ is defined as

1. $\mathsf{AFS.KeyGen}(1^\lambda)$: Choose a uniformly random matrix $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_p^{2 \cdot v^2 \times h}$, $2 \cdot \lambda + 1$ distinct values $k_1, \ldots, k_{2 \cdot \lambda + 1} \xleftarrow{\$} \mathbb{Z}_p^*$ and a random permutation θ of the set $[4 \cdot \lambda + 2]$. Output a fresh key $\mathsf{sk} = (\mathbf{A}, \{k_1, \ldots, k_{2 \cdot \lambda + 1}\}, \theta)$.
2. $\mathsf{AFS.ProbGen}(\mathsf{sk}, \mathbf{M}_1, \mathbf{M}_2)$: Run $\mathbf{A} \cdot \mathbf{k}$ to obtain a vector $\mathbf{d}$ that includes $2 \cdot v^2$ h-degree polynomials, where $\mathbf{k} = [k\ k^2\ \ldots\ k^h]^T$, where k is an indeterminate. Use these h-degree polynomials to mask each element of $\mathbf{M}_1$ and $\mathbf{M}_2$, and generate two ciphertexts $\mathbf{C}_1$ and $\mathbf{C}_2$. Specifically, $\forall i,\ j \in [v]$, $i' \in [2 \cdot v^2] c_{i,j} = \Sigma_{s=1}^{h} a_{i',s} \cdot k^s + m_{i,j}$. For $r \in [2 \cdot \lambda + 1]$, let $k = k_r$ and compute $\mathbf{C}_1(k_r)$ and $\mathbf{C}_2(k_r)$, where $\forall i, j \in [v], i' \in [2 \cdot v^2] c_{i,j}(k_r) = \Sigma_{s=1}^{h} a_{i',s} \cdot k_r^s + m_{i,j}$. Choose $4 \cdot \lambda + 2$ uniformly random matrices $\mathbf{B}_1, \ldots, \mathbf{B}_{4 \cdot \lambda + 2} \xleftarrow{\$} \mathbb{Z}_p^{v \times v}$ and create $2 \cdot \lambda + 1$ pairs $(\mathbf{B}_1, \mathbf{B}_2), \ldots, (\mathbf{B}_{4 \cdot \lambda + 1}, \mathbf{B}_{4 \cdot \lambda + 2})$. A client permutes the $4 \cdot \lambda + 2$ matrix pairs of the set $U = \{(\mathbf{C}_1(k_1), \mathbf{C}_2(k_1)), \ldots, (\mathbf{C}_1(k_{2 \cdot \lambda + 1}), \mathbf{C}_2(k_{2 \cdot \lambda + 1})), (\mathbf{B}_1, \mathbf{B}_2), \ldots, (\mathbf{B}_{4 \cdot \lambda + 1}, \mathbf{B}_{4 \cdot \lambda + 2})\}$ using θ, and sends the permuted set U to a server.

3. $\mathsf{AFS.Compute}(U)$: The products of all matrix pairs in U are computed by the server and put in a set Q. The set Q is sent back to the client.
4. $\mathsf{AFS.ResuGen}(\mathsf{sk}, Q)$: Based on θ, choose some matrices from Q, which correspond to $\mathbf{M}_1$ and $\mathbf{M}_2$. Interpolate these matrices to find the real result of $\mathbf{M}_1 \cdot \mathbf{M}_2$.

For $\mathsf{AF\text{-}PVC}_{single}$, Atallah and Frikken introduced a method to verify the result returned from a server who is lazy or malicious. This verification algorithm is a probabilistic verification process that means successfully detecting a cheating server with non-negligible probability. Since our work focuses on the privacy property of the PVC protocol, we refer to [2] for more details about the verification process.

2.4 Privacy Definition

According to [2], a property of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$, from an informal ciphertext indistinguishability statement, is that it is infeasible for any passive PPT adversary $\mathcal{A}$ to computationally distinguish the ciphertexts over two distinct inputs. Specifically, a ciphertext is a set of matrix pairs (i.e., $U^{(1)}, U^{(2)}, U$). This computational problem is linked to the notion of privacy against passive adversary. Based on different attack models, two formal definitions are given below.

Definition 3 (Privacy Against Passive Eavesdropping). *For a PVC protocol* $\mathsf{PVC} = (\mathsf{KeyGen}, \mathsf{ProbGen}, \mathsf{Compute}, \mathsf{ResuGen})$*, the following experiment associated with a* PPT eavesdropping adversary $\mathcal{A}$ *is considered:*

Experiment $\mathsf{Exp}_{\mathcal{A}}^{ind\text{-}priv^{coa}}[\mathsf{PVC}, \lambda]$*:*

$((\boldsymbol{M}_{1(0)}, \boldsymbol{M}_{2(0)}), (\boldsymbol{M}_{1(1)}, \boldsymbol{M}_{2(1)})) \leftarrow \mathcal{A}(1^{\lambda})$*;*
$\mathsf{sk} \leftarrow \mathsf{KeyGen}(1^{\lambda})$*;*
$b \xleftarrow{\$} \{0, 1\}$*;*
$U_b \leftarrow \mathsf{ProbGen}(\mathsf{sk}, \boldsymbol{M}_{1(b)}, \boldsymbol{M}_{2(b)})$*;*
$b' \leftarrow \mathcal{A}((\boldsymbol{M}_{1(0)}, \boldsymbol{M}_{2(0)}), (\boldsymbol{M}_{1(1)}, \boldsymbol{M}_{2(1)}), U_b)$*;*
If $b' = b$*, output* 1*; else, output* 0*,*

where U_b *is called a challenge ciphertext. The computation of* U_b *is done by the performer of the experiment. Then, we define the advantage of* $\mathcal{A}$ *in the experiment above as follows:*

$$\mathsf{Adv}_{\mathcal{A}}^{ind\text{-}priv^{coa}}(\mathsf{PVC}, \lambda) = \left| \Pr[\mathsf{Exp}_{\mathcal{A}}^{ind\text{-}priv^{coa}}[\mathsf{PVC}, \lambda] = 1] - \frac{1}{2} \right|.$$

PVC is IND-COA private if, for any $\mathcal{A}$*, there exists a negligible function* negl *such that*

$$\mathsf{Adv}_{\mathcal{A}}^{ind\text{-}priv^{coa}}(\mathsf{PVC}, \lambda) \leq \mathsf{negl}(\lambda).$$

Definition 4 (Privacy Against A Chosen-Plaintext Attack). *For a PVC protocol* $\mathsf{PVC} = (\mathsf{KeyGen}, \mathsf{ProbGen}, \mathsf{Compute}, \mathsf{ResuGen})$*, the following experiment associated with a* PPT *adversary* $\mathcal{A}$ *is considered:*

Experiment $\mathsf{Exp}_{\mathcal{A}}^{ind\text{-}priv}[\mathsf{PVC}, \lambda]$*:*

$((\boldsymbol{M}_{1(0)}, \boldsymbol{M}_{2(0)}), (\boldsymbol{M}_{1(1)}, \boldsymbol{M}_{2(1)})) \leftarrow \mathcal{A}^{\mathit{PrivProbGen}(\mathit{KeyGen}(1^\lambda),\cdot,\cdot)}(1^\lambda)$;
$\mathit{sk} \leftarrow \mathit{KeyGen}(1^\lambda)$;
$b \xleftarrow{\$} \{0,1\}$;
$U_b \leftarrow \mathit{ProbGen}(\mathit{sk}, \boldsymbol{M}_{1(b)}, \boldsymbol{M}_{2(b)})$;
$b' \leftarrow \mathcal{A}^{\mathit{PrivProbGen}(\mathit{KeyGen}(1^\lambda),\cdot,\cdot)}((\boldsymbol{M}_{1(0)}, \boldsymbol{M}_{2(0)}), (\boldsymbol{M}_{1(1)}, \boldsymbol{M}_{2(1)}), U_b)$;
If $b' = b$, *output* 1; *else, output* 0,

where the oracle $\mathit{PrivProbGen}(\mathit{KeyGen}(1^\lambda), \boldsymbol{M}_1, \boldsymbol{M}_2)$ *asks* $\mathit{ProbGen}(\mathit{KeyGen}(1^\lambda), \boldsymbol{M}_1, \boldsymbol{M}_2)$ *to obtain a set of matrix pairs* U *and send it back. The output from* $\mathit{PrivProbGen}(\mathit{KeyGen}(1^\lambda), \boldsymbol{M}_1, \boldsymbol{M}_2)$ *is probabilistic. Then, we can define the advantage of* $\mathcal{A}$ *in the experiment above as follows:*

$$\mathit{Adv}_{\mathcal{A}}^{\mathit{ind\text{-}priv}}(\mathit{PVC}, \lambda) = \left| \Pr[\mathit{Exp}_{\mathcal{A}}^{\mathit{ind\text{-}priv}}[\mathit{PVC}, \lambda] = 1] - \tfrac{1}{2} \right|.$$

PVC is IND-CPA private if, for any $\mathcal{A}$, *there exists a negligible function* negl *such that*

$$\mathit{Adv}_{\mathcal{A}}^{\mathit{ind\text{-}priv}}(\mathit{PVC}, \lambda) \leq \mathit{negl}(\lambda).$$

Remark 1. From Katti et al.'s work [11], privacy against passive eavesdropping is equivalent to IND-COA privacy. If a PVC protocol satisfies IND-CPA privacy based on Definition 4, it must also satisfy IND-COA privacy based on Definition 3. However, if a PVC protocol does not satisfy IND-COA privacy, it also does not satisfy IND-CPA privacy.

In [2], Atallah and Frikken gave the detailed proofs for privacy of AF-PVC$_{two}$ and AF-PVC$_{single}$ and the following theorems.

Theorem 4 ([2], Theorem 5). *Assume that the two servers do not collude and the decisional-WSH assumption holds. Then, AF-PVC*$_{two}$ *is IND-CPA private.*

Theorem 5 ([2], Sect. 4.5.3). *Assume that the decisional-SSH assumption holds. Then, AF-PVC*$_{single}$ *is IND-CPA private.*

3 Breaking the Decisional-SH Assumption

In this section, we first present a rigorous analysis for breaking the decisional-WSH assumption and decisional-SSH assumption. Then, we show how the analysis for solving the decisional-SH problem extends naturally to AF-PVC$_{two}$ and AF-PVC$_{single}$, thus demonstrating that both of them are not IND-COA private.

3.1 Adversary's Strategy

For the decisional-WSH problem (resp. decisional-SSH problem) in Definition 1 (resp. Definition 2), if a polynomial-time adversary $\mathcal{A}$ wants to solve this problem with non-negligible advantage, she must employ some unexpected strategy. In general, the adversary's direct strategy is that she tries to find a set that involves ℓ special row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_\ell$ efficiently and evaluate the distinction between

the set of ℓ special row vectors and a set of ℓ uniformly random vectors over $\mathbb{Z}_p^m$. As stated in Theorem 1, any set of $\lambda+1$ row vectors that include at least one uniformly random vector over $\mathbb{Z}_p^m$ is distributed identically to the set of $\lambda+1$ uniformly random vectors over $\mathbb{Z}_p^m$. This implies that $\mathcal{A}$ needs to find at least $\ell = \lambda+1$ special row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$. However, Atallah and Frikken argued that $\mathcal{A}$ is unlikely to find $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$ with significant probability (e.g., Theorem 2)[1].

Then, we take a step back and consider such a question: if we sample a matrix from a distribution which is either the WSH distribution (resp. the SSH distribution) or uniformly random, what type of factor about this matrix do we need to analyze and evaluate? We believe that one of the important factors is the rank of a matrix. This means that the adversary's strategy can be based on the analysis for the rank of a matrix. From this point of view, we propose an adversary's strategy that proceeds in two steps.

Strategy Overview: Let $\mathbf{X}$ be an $n \times m$ matrix that is sampled from a distribution which is either the WSH distribution (resp. the SSH distribution) $\chi(p)^{n\times m}$ or the uniform distribution over $\mathbb{Z}_p^{n\times m}$.

1. Compute the rank of $\mathbf{X}$, denoted by $\mathsf{rank}(\mathbf{X})$.
2. Check whether $\mathsf{rank}(\mathbf{X})$ is below some value $\varepsilon \leq \mathsf{min}(n, m)$ or not below this value. If $\mathsf{rank}(\mathbf{X})$ is below ε, $\mathbf{X}$ is sampled from $\chi(p)^{n\times m}$; otherwise, $\mathbf{X}$ is sampled from the uniform distribution over $\mathbb{Z}_p^{n\times m}$.

Why the Rank-Based Analysis Works? The idea of the proposed strategy is remarkably simple. It focuses on a distinguishing problem about the distributions of ranks of matrices from those two distributions. Specifically, the value ε can be seen as a threshold rank that is the critical factor of the proposed strategy. To motivate why computing the rank of a matrix is useful for solving the decisional-WSH problem and decisional-SSH problem, we list the following two facts:

- **Fact 1:** Consider an $n \times m$ matrix $\mathbf{X}$ over $\mathbb{Z}_p^{n\times m}$. W.l.o.g. assume that $n \leq m$. If there are $\ell < n$ linearly dependent row vectors in $\mathbf{X}$, all the n row vectors of $\mathbf{X}$ are linearly dependent. This implies that the rank of $\mathbf{X}$ must be below n (i.e., $\mathsf{rank}(\mathbf{X}) < n$).
- **Fact 2:** Consider an $n \times m$ matrix $\mathbf{X}$ sampled from the uniform distribution over $\mathbb{Z}_p^{n\times m}$. W.l.o.g. assume that $n \leq m$. With high probability, the n row vectors of $\mathbf{X}$ are linearly independent, and the rank of $\mathbf{X}$ is n (i.e., $\mathsf{rank}(\mathbf{X}) = n$).

Specifically, based on Linial and Weitz's work [13] (see Eq. (2)), we verify the **Fact 2** concretely. To implement this verification, we choose parameters $p > 4 \cdot \lambda + 2$, $n = 2 \cdot \lambda + 1$ and $m \geq n$, and compute the results on the probabilities of the full-row-rank matrices for different parameters. The verification

[1] In Sect. 5, we show that $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$ can be found (with overwhelming probability) by employing our adversary's strategy. Here, we want to show that these vectors are unlikely to be found without using our adversary's strategy.

results show that the probability of a uniformly random matrix having rank n is nearly 1, i.e., $\Pr[\mathsf{rank}(\mathbf{X}) = n] \approx 1$, which can show the rank distribution of the uniformly random matrices over $\mathbb{Z}_p^{n\times m}$. For more details about the verification results, we refer the reader to the full version of our paper.

$$\Pr[\mathsf{rank}(\mathbf{X}) = z] = \frac{1}{p^{(n-z)\cdot(m-z)}} \cdot \prod_{i=0}^{z-1} \frac{(1-p^{i-n})\cdot(1-p^{i-m})}{1-p^{i-z}} \tag{2}$$

According to **Fact 1** and **Fact 2**, if the matrices sampled from some distribution over $\mathbb{Z}_p^{n\times m}$ always have some linearly dependent row vectors, the ranks of these matrices are always below the matrix sizes, and the rank distribution is distinguished from the rank distribution of the uniformly random matrices over $\mathbb{Z}_p^{n\times m}$ with non-negligible advantage.

Then, based on the above analysis, if $\mathcal{A}$ employs the proposed strategy to solve the decisional-WSH problem and decisional-SSH problem, the crux is that whether there are $\lambda+1$ linearly dependent special row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$ and what is the probability that $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$ are linearly dependent. Assume that $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$ must be linearly dependent, then the rank of a matrix sampled from $\chi(p)^{n\times m}$ can leak information about the matrix structure. In what follows, we focus on exploring the linear relation of the $\lambda+1$ special row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$ and give the answer.

3.2 Analysis for the Decisional-SH Assumption

To show the linear relation of the $\lambda+1$ special row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$, we consider a set of the transposes of the $\lambda+1$ vectors $[(\mathbf{d}_1)^T \ \ldots (\mathbf{d}_{\lambda+1})^T]$ as the product of two matrices $\mathbf{A}\cdot\mathbf{K}$, where $\mathbf{A}$ is an $m\times\lambda$ uniformly random matrix where the i^{th} column is $\mathbf{a}_i$ for $i\in[\lambda]$, and $\mathbf{K}$ is a $\lambda\times(\lambda+1)$ matrix where the r^{th} column is $\mathbf{k}_r = [k_r \ k_r^2 \ \ldots \ k_r^\lambda]^T$ for $r\in[\lambda+1]$. Specifically, according to **Fact 2**, our following analysis focuses on the case with high probability that the vectors $\mathbf{a}_1, \ldots, \mathbf{a}_\lambda$ are linearly independent[2].

Lemma 2. *Consider an $m\times(\lambda+1)$ matrix $(\boldsymbol{A}\cdot\boldsymbol{K})$. Assume that $m = \mathit{poly}(\lambda) > \lambda$, and the column vectors $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_\lambda$ are linearly independent. Then, $\mathit{rank}(\boldsymbol{A}\cdot\boldsymbol{K}) < \mathit{min}(m, \lambda+1)$, which implies that the special row vectors $\boldsymbol{d}_1, \ldots, \boldsymbol{d}_{\lambda+1}$ are linearly dependent.*

Proof. The result in this lemma is immediate, actually. For the formal proof, we refer the reader to the full version of our paper.

According to Eq. (2), the probability that the column vectors $\mathbf{a}_1, \ldots, \mathbf{a}_\lambda$ are linearly independent (i.e., $\mathsf{rank}(\mathbf{A}) = \lambda$) is $\Pi_{i=0}^{\lambda-1}(1-p^{i-m})$. Then, the probability that the row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$ are linearly dependent is also $\Pi_{i=0}^{\lambda-1}(1-p^{i-m})$. Specifically, if p is a large prime (e.g., $p > 4\cdot\lambda+2$), the row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_{\lambda+1}$ are likely to be linearly dependent.

Then, based on the proposed adversary's strategy and Lemma 2, we show our main analysis results for solving the decisional-WSH problem and decisional-SSH problem.

[2] In the full version of our paper, we will present an analysis that also considers the case that the vectors $\mathbf{a}_1, \ldots, \mathbf{a}_\lambda$ are linearly dependent.

Lemma 3. *Consider a sample* $\boldsymbol{X}$ *from either the WSH distribution (resp. the SSH distribution)* $\chi(p)^{n\times m}$ *or the uniform distribution over* $\mathbb{Z}_p^{n\times m}$*, where* $n \in \{2\cdot\lambda+1, 2\cdot\lambda+2\cdot e+2\}$*. Assume that* $m > 2\cdot\lambda$ *(resp.* $m > 2\cdot\lambda+e+1$*), and* p *is a large prime, e.g.,* $p > 4\cdot\lambda+2$*. Let* $\varphi = \Pi_{i=0}^{\lambda-1}(1-p^{i-m})$*. Let* $\eta = Pr[\mathsf{rank}(\boldsymbol{X}) = z]$*, where the probability is for the case that* $\boldsymbol{X}$ *is uniformly random, and* $z = \mathsf{min}(n,m)$*. If* $\mathsf{rank}(\boldsymbol{X}) < \mathsf{min}(n,m)$*, the probability that* $\boldsymbol{X}$ *is sampled from* $\chi(p)^{n\times m}$ *satisfies* $Pr[\boldsymbol{X} \leftarrow \chi(p)^{n\times m} | \mathsf{rank}(\boldsymbol{X}) < \mathsf{min}(n,m)] \geq \frac{1}{1+\frac{1-\eta}{\varphi}}$*, and if* $\mathsf{rank}(\boldsymbol{X}) = \mathsf{min}(n,m)$*, the probability that* $\boldsymbol{X}$ *is uniformly random satisfies* $Pr[\boldsymbol{X} \xleftarrow{\$} \mathbb{Z}_p^{n\times m} \mid \mathsf{rank}(\boldsymbol{X}) = \mathsf{min}(n,m)] \geq \frac{1}{1+\frac{1-\varphi}{\eta}}$*.*

Proof. For the detailed proof, we refer the reader to the full version of our paper.

In Lemma 3, since p is a large prime, we can obtain $\Pr[\mathbf{X} \leftarrow \chi(p)^{n\times m} | \mathsf{rank}(\mathbf{X}) < \mathsf{min}(n,m)] \approx 1$ and $\Pr[\mathbf{X} \xleftarrow{\$} \mathbb{Z}_p^{n\times m} | \mathsf{rank}(\mathbf{X}) = \mathsf{min}(n,m)] \approx 1$.

Theorem 6. *Let* $\varphi = \Pi_{i=0}^{\lambda-1}(1-p^{i-m})$*. Let* $\eta = Pr[\mathsf{rank}(\boldsymbol{X}) = z]$ *denote the probability that the rank of any* $n\times m$ *uniformly random matrix* $\boldsymbol{X}$ *is* z*, where* $z = \mathsf{min}(n,m)$*, where* $n \in \{2\cdot\lambda+1, 2\cdot\lambda+2\cdot e+2\}$*. Assume that* $m > 2\cdot\lambda$ *(resp.* $m > 2\cdot\lambda+e+1$*), and* p *is a large prime, e.g.,* $p > 4\cdot\lambda+2$*. Then there exists an adversary* $\mathcal{A}$ *running in polynomial-time* t *for solving the decisional-WSH problem (resp. decisional-SSH problem) with*

$$\mathsf{Adv}_{\mathcal{A},SH}(p,n,m) \geq \tfrac{1}{2}\cdot(\varphi+\eta) - \tfrac{1}{2}\,,$$

where t *is used to compute the rank of a matrix. Specifically, since* p *is a large prime,* $\mathcal{A}$ *has advantage* $\mathsf{Adv}_{\mathcal{A},SH}(p,n,m) \approx \frac{1}{2}$ *in solving the decisional-WSH problem (resp. decisional-SSH problem).*

Proof. Let $\mathsf{Unif}(\mathbb{Z}_p^{n\times m})$ denote the uniform distribution over $\mathbb{Z}_p^{n\times m}$. The adversary $\mathcal{A}$ has access to an oracle that is either $\chi(p)^{n\times m}$ or $\mathsf{Unif}(\mathbb{Z}_p^{n\times m})$. She calls the oracle arbitrarily many times (i.e., a polynomial number of times) to obtain samples of the form $\mathbf{X}_i$ and uses the rank-based adversary's strategy to evaluate each sample. If $\mathsf{rank}(\mathbf{X}_i) < \mathsf{min}(n,m)$, $\mathcal{A}$ outputs $\chi(p)^{n\times m}$. If $\mathsf{rank}(\mathbf{X}_i) = \mathsf{min}(n,m)$, $\mathcal{A}$ returns $\mathsf{Unif}(\mathbb{Z}_p^{n\times m})$.

We first look at the probability distribution of the rank of $\mathbf{X}_i$ when the oracle that $\mathcal{A}$ has access to is $\mathsf{Unif}(\mathbb{Z}_p^{n\times m})$. In this case it's easy to see that $\Pr[\mathsf{rank}(\mathbf{X}_i) = \mathsf{min}(n,m)] = \eta$ and $\Pr[\mathsf{rank}(\mathbf{X}_i) < \mathsf{min}(n,m)] = 1-\eta$.

If the oracle is $\chi(p)^{n\times m}$, as discussed earlier, we have $\Pr[\mathsf{rank}(\mathbf{X}_i) = \mathsf{min}(n,m)] \leq 1-\varphi$ and $\Pr[\mathsf{rank}(\mathbf{X}_i) < \mathsf{min}(n,m)] \geq \varphi$.

Thus, based on Lemma 3, we obtain the success probability (see Sect. 2.1)

$$\begin{aligned}
&\mathsf{Suc}_{\mathcal{A},\mathrm{SH}}(p,n,m)\\
&= \Pr[\mathcal{A}(\mathbf{X}) = b | \mathsf{rank}(\mathbf{X}) < \mathsf{min}(n,m)] \cdot \Pr[\mathsf{rank}(\mathbf{X}) < \mathsf{min}(n,m)] +\\
&\quad \Pr[\mathcal{A}(\mathbf{X}) = b | \mathsf{rank}(\mathbf{X}) = \mathsf{min}(n,m)] \cdot \Pr[\mathsf{rank}(\mathbf{X}) = \mathsf{min}(n,m)]\\
&= \Pr[\mathbf{X} \leftarrow \chi(p)^{n\times m} | \mathsf{rank}(\mathbf{X}) < \mathsf{min}(n,m)] \cdot \Pr[\mathsf{rank}(\mathbf{X}) < \mathsf{min}(n,m)] +\cdot\\
&\quad \Pr[\mathbf{X} \xleftarrow{\$} \mathbb{Z}_p^{n\times m} | \mathsf{rank}(\mathbf{X}) = \mathsf{min}(n,m)] \cdot \Pr[\mathsf{rank}(\mathbf{X}) = \mathsf{min}(n,m)]\\
&\geq \tfrac{\varphi}{\varphi+1-\eta} \cdot \tfrac{1-\eta+\varphi}{2} + \tfrac{1}{2}\cdot\eta = \tfrac{1}{2}\cdot\varphi + \tfrac{1}{2}\cdot\eta
\end{aligned}$$

This means that $\mathsf{Adv}_{\mathcal{A},\mathrm{SH}}(p,n,m) \geq \frac{1}{2}\cdot\varphi+\frac{1}{2}\cdot\eta-\frac{1}{2}$. Specifically, when p is a large prime, the value of φ is close to 1. Moreover, as discussed in Sect. 3.1, η is also close to 1 if p is not a small prime. Then, $\mathsf{Adv}_{\mathcal{A},\mathrm{SH}}(p,n,m)$ is close to $\frac{1}{2}$, which confirms our theorem.

Theorem 6 demonstrates that we can break the decisional-WSH assumption and decisional-SSH assumption efficiently for a wide range of parameters. The final result contradicts Atallah and Frikken's result in Eq. (1). However, this does not imply that we can solve the search-WSH problem and search-SSH problem efficiently, which shows the inaccuracy of Theorem 3.

3.3 Analysis for AF-PVC$_{two}$ and AF-PVC$_{single}$

Now we want to present the formal analysis for privacy of AF-PVC$_{two}$ and AF-PVC$_{single}$. Specifically, it is straightforward to use the idea of the analysis for the decisional-WSH assumption and decisional-SSH assumption to undermine the privacy of AF-PVC$_{two}$ and AF-PVC$_{single}$. This means that an adversary $\mathcal{A}$ employs the rank-based strategy to evaluate a given ciphertext matrix. Note that, our analysis is based on the IND-COA experiment (see Definition 3), where an eavesdropping adversary $\mathcal{A}$ running in polynomial-time has non-negligible advantage to show that both protocols are not IND-COA private (and thus not IND-CPA private).

Lemma 4. *Given a uniformly random matrix $\boldsymbol{A}\in\mathbb{Z}_p^{2\cdot v^2\times\lambda}$ where the i^{th} column is $\boldsymbol{a}_i$ for $i\in[\lambda]$, a $\lambda\times n$ matrix $\boldsymbol{K}$ where the r^{th} column is $\boldsymbol{k}_r=[k_r\ k_r^2\ \ldots\ k_r^\lambda]^T$ for $r\in[n]$, and a $2\cdot v^2\times n$ matrix $\boldsymbol{S}$ where the elements of the i^{th} column $\boldsymbol{s}_i$ are the same as the corresponding elements of the j^{th} column $\boldsymbol{s}_j$ for $i,j\in[n]$, where $\boldsymbol{s}_i\xleftarrow{\$}\mathbb{Z}_p^{2\cdot v^2}$[3]. Let p be a large prime (e.g., $p>4\cdot\lambda+2$), $v=\mathsf{poly}(\lambda)>\sqrt{\frac{\lambda}{2}}$ and $n\in\{\lambda+1,2\cdot\lambda+1\}$. Assume that the column vectors $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_\lambda,\boldsymbol{s}_i$ are linearly independent. Then, for the $2\cdot v^2\times n$ matrix $(\boldsymbol{A}\cdot\boldsymbol{K}+\boldsymbol{S})$, we have $\mathsf{rank}(\boldsymbol{A}\cdot\boldsymbol{K}+\boldsymbol{S})=\lambda+1$.*

Proof. For the formal proof, we refer the reader to the full version of our paper.

Corollary 1. *Consider two $2\cdot v^2\times n$ matrices $(\boldsymbol{A}\cdot\boldsymbol{K}+\boldsymbol{S})$ and $(\boldsymbol{A}\cdot\boldsymbol{K}+\boldsymbol{Z})$, where the definitions of $\boldsymbol{A},\boldsymbol{K}$ and $\boldsymbol{S}$ are in Lemma 4, and $\boldsymbol{Z}$ is a $2\cdot v^2\times n$ zero matrix. Let $v=\mathsf{poly}(\lambda)>\sqrt{\frac{\lambda}{2}}$ and $n\in\{\lambda+1,2\cdot\lambda+1\}$. Then the probability $Pr[\mathsf{rank}(\boldsymbol{A}\cdot\boldsymbol{K}+\boldsymbol{S})=\lambda+1]=\Pi_{i=0}^{\lambda}(1-p^{i-2\cdot v^2})$, and the probability $Pr[\mathsf{rank}(\boldsymbol{A}\cdot\boldsymbol{K}+\boldsymbol{Z})<\lambda+1]=\Pi_{i=0}^{\lambda-1}(1-p^{i-2\cdot v^2})$ for the case that the vectors $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_\lambda,\boldsymbol{s}_i$ are linearly independent.*

Proof. We again refer the reader to the full version of our paper for the detailed proof.

[3] For column vectors $\mathbf{s}_1,\mathbf{s}_2,\ldots,\mathbf{s}_n$, since $\mathbf{s}_1=\mathbf{s}_2=\cdots=\mathbf{s}_n,\mathbf{S}=[\mathbf{s}_1\ \mathbf{s}_1\ \ldots\ \mathbf{s}_1]$, where $\mathbf{s}_1\xleftarrow{\$}\mathbb{Z}_p^{2\cdot v^2}$.

Based on Lemma 4 and Corollary 1, we present the following theorems of breaking the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$.

Theorem 7. *The protocol AF-PVC$_{two}$ does not satisfy IND-COA privacy based on Definition 3 under the condition that the size of the message space p is a large prime (e.g., $p > 4 \cdot \lambda + 2$) and the matrix size $v > \sqrt{\lambda}$. Specifically, the advantage of an PPT adversary $\mathcal{A}$ for breaking the privacy of this protocol is close to $\frac{1}{2}$.*

Proof. According to Definition 3, for the experiment $\mathsf{Exp}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}[\mathsf{AF\text{-}PVC}_{two}, \lambda]$, an PPT adversary $\mathcal{A}$ chooses two pairs of $v \times v$ matrices $(\mathbf{M}_{1(0)}, \mathbf{M}_{2(0)})$, $(\mathbf{M}_{1(1)}, \mathbf{M}_{2(1)})$. Specifically, $(\mathbf{M}_{1(0)}, \mathbf{M}_{2(0)}) \xleftarrow{\$} \mathbb{Z}_p^{v \times v} \times \mathbb{Z}_p^{v \times v}$, and $(\mathbf{M}_{1(1)}, \mathbf{M}_{2(1)})$ are two zero matrices. The challenge ciphertext is the matrix set U_b that comes from the second server (i.e., $U_b^{(2)}$). $\mathcal{A}$ flattens out each pair of matrices of U_b into a list of $2 \cdot v^2$ values to generate a $(2 \cdot \lambda + 1) \times 2 \cdot v^2$ matrix $\mathbf{E}_b$. $\mathbf{E}_b$ involves either all rows of the matrix $(\mathbf{A} \cdot \mathbf{K} + \mathbf{S})^T$ or all rows of the matrix $(\mathbf{A} \cdot \mathbf{K} + \mathbf{Z})^T$, where the descriptions of the transposes of these two matrices are in Corollary 1. For winning the experiment in Definition 3, $\mathcal{A}$ employs a PPT distinguisher $\mathcal{D}$ based on the proposed adversary's strategy as follows:

Distinguisher $\mathcal{D}$:

- For the case $\mathsf{rank}(\mathbf{E}_b) = 2 \cdot \lambda + 1, \mathcal{A}$ outputs $b' = 0$.
- For the case $\mathsf{rank}(\mathbf{E}_b) < 2 \cdot \lambda + 1, \mathcal{A}$ outputs $b' = 1$.

The positive integer $2 \cdot \lambda + 1$ is regarded as the threshold rank. If $\mathsf{Adv}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}(\mathsf{AF\text{-}PVC}_{two}, \lambda)$ is non-negligible, then $\mathsf{AF\text{-}PVC}_{two}$ is not IND-COA private. In what follows, we show this result by considering a large prime p (e.g., $p > 4 \cdot \lambda + 2$) and a matrix size $v > \sqrt{\lambda}$.

$$\begin{aligned}
&\Pr[\mathsf{Exp}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}[\mathsf{AF\text{-}PVC}_{two}, \lambda] = 1] \\
&= \Pr[\mathbf{E}_b = \mathbf{E}_0 | \mathsf{rank}(\mathbf{E}_b) = 2 \cdot \lambda + 1] \cdot \Pr[\mathsf{rank}(\mathbf{E}_b) = 2 \cdot \lambda + 1] \\
&\quad + \Pr[\mathbf{E}_b = \mathbf{E}_1 | \mathsf{rank}(\mathbf{E}_b) < 2 \cdot \lambda + 1] \cdot \Pr[\mathsf{rank}(\mathbf{E}_b) < 2 \cdot \lambda + 1]
\end{aligned}.$$

Specifically, from Corollary 1, we obtain

$$\begin{cases}
\Pr[\mathsf{rank}(\mathbf{E}_b) = 2 \cdot \lambda + 1 | \mathbf{E}_b = \mathbf{E}_0] = \prod_{i=0}^{\lambda} (1 - p^{i-2 \cdot v^2}) \cdot \prod_{i=0}^{2 \cdot \lambda} (1 - p^{i-2 \cdot v^2}) \\
\Pr[\mathsf{rank}(\mathbf{E}_b) < 2 \cdot \lambda + 1 | \mathbf{E}_b = \mathbf{E}_1] \geq \prod_{i=0}^{\lambda - 1} (1 - p^{i-2 \cdot v^2})
\end{cases}.$$

Thus, we have

$$\begin{aligned}
&\Pr[\mathsf{Exp}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}[\mathsf{AF\text{-}PVC}_{two}, \lambda] = 1] \\
&\geq \tfrac{1}{2} \cdot \prod_{i=0}^{\lambda} (1 - p^{i-2 \cdot v^2}) \cdot \prod_{i=0}^{2 \cdot \lambda} (1 - p^{i-2 \cdot v^2}) + \tfrac{1}{2} \cdot \prod_{i=0}^{\lambda - 1} (1 - p^{i-2 \cdot v^2}) \\
&= \tfrac{1}{2} \cdot \prod_{i=0}^{\lambda - 1} (1 - p^{i-2 \cdot v^2}) \cdot ((1 - p^{\lambda - 2 \cdot v^2}) \cdot \prod_{i=0}^{2 \cdot \lambda} (1 - p^{i-2 \cdot v^2}) + 1)
\end{aligned}.$$

Since p is a large prime, as discussed earlier, we can obtain $\Pr[\mathsf{Exp}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}[\mathsf{AF\text{-}PVC}_{two}, \lambda] = 1] \approx 1$. This means that $\mathsf{Adv}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}(\mathsf{AF\text{-}PVC}_{two}, \lambda) \approx \frac{1}{2} \not\leqslant \mathsf{negl}(\lambda)$.

Theorem 8. *The protocol AF-PVC$_{single}$ does not satisfy IND-COA privacy based on Definition 3 under the condition that the size of the message space p is a large prime (e.g., $p > 4 \cdot \lambda + 2$) and the matrix size $v > \sqrt{\frac{3 \cdot \lambda + 1}{2}}$. Specifically, the advantage of an PPT adversary $\mathcal{A}$ for breaking the privacy of this protocol is close to $\frac{1}{2}$.*

Proof. The proof follows a similar procedure to that for Theorem 7. For the experiment $\mathsf{Exp}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}[\mathsf{AF\text{-}PVC}_{single}, \lambda]$ in Definition 3, an PPT adversary $\mathcal{A}$ also chooses two pairs of $v \times v$ matrices $(\mathbf{M}_{1(0)}, \mathbf{M}_{2(0)})$ and $(\mathbf{M}_{1(1)}, \mathbf{M}_{2(1)})$, where $(\mathbf{M}_{1(0)}, \mathbf{M}_{2(0)}) \xleftarrow{\$} \mathbb{Z}_p^{v \times v} \times \mathbb{Z}_p^{v \times v}$, and $(\mathbf{M}_{1(1)}, \mathbf{M}_{2(1)})$ are two zero matrices. The challenge ciphertext is the matrix set U_b. $\mathcal{A}$ flattens out each pair of matrices of U_b into a list of $2 \cdot v^2$ values to generate a $(4 \cdot \lambda + 2) \times 2 \cdot v^2$ matrix $\mathbf{E}_b$. $\mathbf{E}_b$ includes either all rows of the matrix $(\mathbf{A} \cdot \mathbf{K} + \mathbf{S})^T$ or all rows of the matrix $(\mathbf{A} \cdot \mathbf{K} + \mathbf{Z})^T$. To win the experiment in Definition 3, $\mathcal{A}$ employs a PPT distinguisher $\widehat{\mathcal{D}}$ based on the proposed adversary's strategy as follows:

Distinguisher $\widehat{\mathcal{D}}$:

- For the case $\mathsf{rank}(\mathbf{E}_b) = 3 \cdot \lambda + 2$, $\mathcal{A}$ outputs $b' = 0$.
- For the case $\mathsf{rank}(\mathbf{E}_b) < 3 \cdot \lambda + 2$, $\mathcal{A}$ outputs $b' = 1$.

The positive integer $3 \cdot \lambda + 2$ is regarded as the threshold rank. If $\mathsf{Adv}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}(\mathsf{AF\text{-}PVC}_{single}, \lambda)$ is non-negligible, then $\mathsf{AF\text{-}PVC}_{single}$ is not IND-COA private. In what follows, we show this result by considering a large prime p (e.g., $p > 4 \cdot \lambda + 2$) and a matrix size $v > \sqrt{\frac{3 \cdot \lambda + 1}{2}}$.

$$\begin{aligned}
&\Pr[\mathsf{Exp}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}[\mathsf{AF\text{-}PVC}_{single}, \lambda] = 1] \\
&= \Pr[\mathbf{E}_b = \mathbf{E}_0 | \mathsf{rank}(\mathbf{E}_b) = 3 \cdot \lambda + 2] \cdot \Pr[\mathsf{rank}(\mathbf{E}_b) = 3 \cdot \lambda + 2] \\
&\quad + \Pr[\mathbf{E}_b = \mathbf{E}_1 | \mathsf{rank}(\mathbf{E}_b) < 3 \cdot \lambda + 2] \cdot \Pr[\mathsf{rank}(\mathbf{E}_b) < 3 \cdot \lambda + 2]
\end{aligned}.$$

Specifically, from Corollary 1, we have

$$\begin{cases}
\Pr[\mathsf{rank}(\mathbf{E}_b) = 3 \cdot \lambda + 2 | \mathbf{E}_b = \mathbf{E}_0] = \prod_{i=0}^{\lambda} (1 - p^{i-2 \cdot v^2}) \cdot \prod_{i=0}^{3 \cdot \lambda + 1} (1 - p^{i-2 \cdot v^2}) \\
\Pr[\mathsf{rank}(\mathbf{E}_b) < 3 \cdot \lambda + 2 | \mathbf{E}_b = \mathbf{E}_1] \geq \prod_{i=0}^{\lambda - 1} (1 - p^{i-2 \cdot v^2})
\end{cases}.$$

Then, we can obtain

$$\begin{aligned}
&\Pr[\mathsf{Exp}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}[\mathsf{AF\text{-}PVC}_{single}, \lambda] = 1] \\
&\geq \tfrac{1}{2} \cdot \prod_{i=0}^{\lambda} (1 - p^{i-2 \cdot v^2}) \cdot \prod_{i=0}^{3 \cdot \lambda + 1} (1 - p^{i-2 \cdot v^2}) + \tfrac{1}{2} \cdot \prod_{i=0}^{\lambda - 1} (1 - p^{i-2 \cdot v^2}) \\
&= \tfrac{1}{2} \cdot \prod_{i=0}^{\lambda - 1} (1 - p^{i-2 \cdot v^2}) \cdot ((1 - p^{\lambda - 2 \cdot v^2}) \cdot \prod_{i=0}^{3 \cdot \lambda + 1} (1 - p^{i-2 \cdot v^2}) + 1)
\end{aligned}.$$

Since p is a large prime, as discussed earlier, the above success probability $\Pr[\mathsf{Exp}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}[\mathsf{AF\text{-}PVC}_{single}, \lambda] = 1] \approx 1$. This implies that the adversary's advantage $\mathsf{Adv}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}(\mathsf{AF\text{-}PVC}_{single}, \lambda) \approx \frac{1}{2} \nleq \mathsf{negl}(\lambda)$.

3.4 Discussion

In order to make the readers fully understood our rank-based analyses, we present some concrete discussions below.

Parameters: The parameter choice is significant for our analyses of solving the decisional-SH problem and breaking the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$.

First, for the size of the message space p, it should be set as a large prime that is at least larger than λ, e.g., $p > 4 \cdot \lambda + 2$. On the one hand, as shown in Sect. 3.1, if p is large enough, with high probability, the vectors from an uniformly random matrix are linearly independent. This is necessary for our analyses. On the other hand, if a client wants to outsource the multiplication of some matrix pair $(\mathbf{M}_1, \mathbf{M}_2) \in \mathbb{Z}_p^{v \times v} \times \mathbb{Z}_p^{v \times v}$ to a powerful server, then the message space of each element in these two matrices should be large, which makes the client hard to run the expensive computation. Otherwise, there is no need to do the outsourcing, and the client can carry out the computation locally. This means that the decisional-SH problem and the feasible protocols $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$ with a suitable large parameter p are the targets of our analyses.

Second, for the matrix size v (resp. the matrix size m involved in the decisional-SH problem), it should satisfy $v > \sqrt{\lambda}$ (see Theorem 7) or $v > \sqrt{\frac{3 \cdot \lambda + 1}{2}}$ (see Theorem 8) (resp. $m > 2 \cdot \lambda$ or $m > 2 \cdot \lambda + e + 1$ for $e \in \mathbb{N}_+$ (see Theorem 6)[4]). On the one hand, for an $n \times 2 \cdot v^2$ (resp. $n \times m$) matrix, where $n \in \{2 \cdot \lambda + 1, 4 \cdot \lambda + 2$ (resp. $2 \cdot \lambda + 2 \cdot e + 2)\}$, since the rank of the matrix is dependent on $\mathsf{min}(n, 2 \cdot v^2$ (resp. $m))$, if v (resp. m) does not satisfy the above condition, the rank-based adversary's strategy no longer has any effect. This means that our analyses cannot solve the decisional-SH problem and break the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$. On the other hand, if a client wants to outsource the multiplication of some matrix pair to a powerful server, a key requirement is that these two matrices should be large-scale, which makes the outsourcing practical. If the matrix size does not satisfy the above condition, e.g., $v \leq \sqrt{\lambda}$, the amount of work performed by the client for the outsourcing may be not substantially cheaper than performing the computation on its own. This implies that the outsourcing may be impractical. Therefore, this demonstrates that our analyses focus on the meaningful decisional-SH problem and protocols $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$.

Adversary's Cost: The cost of our analyses is generated by computing the rank of a matrix. We can employ any existing algorithm for obtaining the rank of a matrix. In general, for a matrix from $\mathbb{Z}_p^{n \times m}$ with rank $z \leq \mathsf{min}(n, m)$, using Gaussian elimination, we may compute the rank of the matrix in $O(z \cdot n \cdot m)$

[4] $2 \cdot v^2$ is equivalent to m.

field operations and storage of $n \cdot m$ field elements [17]. Of course, the rank of a matrix can be computed probabilistically by invoking the blackbox approaches, e.g., the Wiedemann method [10,18]. More concretely, if our analyses employ the blackbox method, for a matrix from $\mathbb{Z}_p^{(2\cdot\lambda+1)\times(2\cdot\lambda+1)}$ with rank $2\cdot\lambda$, we need to take $\tilde{O}(2\cdot\lambda\cdot(2\cdot\lambda+1)^2)$ time and use $\tilde{O}(2\cdot\lambda+1)$ storage to obtain the rank of this matrix, where we employ the "$soft-Oh$" (i.e., $\tilde{O}$) notation to suppress log factors.

4 Experimental Verifications

In order to give the reader a glance at the practical results of our analyses for solving the decisional-SH problem and breaking the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$. We implemented our analyses in Sect. 3 and reported the adversary's advantages and costs.

4.1 Setup

Hardware and Software: We conducted the real example experiments on a Lenovo ThinkStation (Intel(R) Xeon(R) E5-2620, 24 hyperthreaded cores at 2.00 GHz, 8 GB RAM at 2.00 GHz), on Windows (Windows 7, x64_64). Our implementations are single-threaded. We used the NTL library [1] version 10.5.0 for the field operations over $\mathbb{Z}_p$ and the matrix operations.

Parameters Choice: In our implementations we covered $\lambda = 80, 128, 192$ and 256 privacy. These selections lead to the parameters in Table 1, where $e = \lambda, n \in \{2\cdot\lambda+1, 2\cdot\lambda+2\cdot e+2\}$, $m \in \{2\cdot\lambda+1, 3\cdot\lambda+1\}$ for $n = 2\cdot\lambda+1$ and $m \in \{3\cdot\lambda+1, 4\cdot\lambda+2\}$ for $n = 2\cdot\lambda+2\cdot e+2, p > 4\cdot\lambda+2$, $v = \lceil\sqrt{\lambda}+1\rceil$ for $n = 2\cdot\lambda+1$ and $v = \lceil\sqrt{\frac{3\cdot\lambda+1}{2}}+1\rceil$ for $n = 2\cdot\lambda+2\cdot e+2$[5], and $h = \lambda$.

Table 1. The used parameters for our analyses

λ	e	n	m	v	p	h
80	N/A	161	161, 241	10	353, 401	80
	80	322	241, 322	12		
128	N/A	257	257, 385	13	631, 701	128
	128	514	385, 514	15		
192	N/A	385	385, 577	15	809, 907	192
	192	770	577, 770	18		
256	N/A	513	513, 769	17	1069, 1187	256
	256	1026	769, 1026	21		

[5] $n = 4\cdot\lambda+2$.

4.2 Results and Timings

The experimental results are presented in Tables 2, 3, 4 and 5. Specifically, the adversary's advantages and timings for solving the decisional-WSH problem and decisional-SSH problem are shown in Tables 2 and 3, and the adversary's advantages and timings for breaking the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$ are reported in Tables 4 and 5. To obtain these results, we compute the advantages that the adversaries answer correctly in the whole experiment process (i.e., 200 experiments). Note that, for each experiment of solving the decisional-WSH problem (resp. the decisional-SSH problem), a fresh sample from either the distribution $\chi(p)^{n\times m}$ or the uniform distribution over $\mathbb{Z}_p^{n\times m}$ is used for the guess. For each experiment of breaking the privacy of $\mathsf{AF\text{-}PVC}_{two}$ (resp. $\mathsf{AF\text{-}PVC}_{single}$), a fresh key sk used by $\mathsf{AF\text{-}PVC}_{two}$ (resp. $\mathsf{AF\text{-}PVC}_{single}$) is generated to complete the matrix masking. Moreover, for each timing in the tables, the value is the average value over 200 experiments.

Table 2. Results on the decisional-WSH problem

(n, m, p)	$\mathsf{Adv}_{\mathcal{A},\mathrm{WSH}}(p, n, m)$	Timing (second)
(161, 161, 353)	0.495	0.725
(161, 241, 353)	0.495	0.998
(161, 161, 401)	0.500	0.723
(161, 241, 401)	0.500	0.997
(257, 257, 631)	0.500	2.843
(257, 385, 631)	0.500	4.646
(257, 257, 701)	0.495	2.832
(257, 385, 701)	0.500	4.633
(385, 385, 809)	0.500	9.304
(385, 577, 809)	0.500	15.253
(385, 385, 907)	0.500	9.284
(385, 577, 907)	0.500	15.137
(513, 513, 1069)	0.500	21.558
(513, 769, 1069)	0.500	35.944
(513, 513, 1187)	0.500	21.629
(513, 769, 1187)	0.500	35.997

As reported in Tables 2, 4 and 5, all the experimental results about the adversary's advantage are in accord with the analyses of Theorems 6, 7 and 8, and demonstrate that there exists a PPT adversary algorithm that (almost) always succeeds in guessing the distribution of a given sample or the bit b in the eavesdropping indistinguishability experiment (see Definition 3). For the results in

Table 3. Results on the decisional-SSH problem

(n, m, p)	$\mathsf{Adv}_{\mathcal{A},\mathrm{SSH}}(p, n, m)$	Timing (second)
(322, 241, 353)	0.005	2.543
(322, 322, 353)	0.500	3.843
(322, 241, 401)	0.010	4.447
(322, 322, 401)	0.495	3.746
(514, 385, 631)	0.030	9.610
(514, 514, 631)	0.500	15.348
(514, 385, 701)	0.050	9.476
(514, 514, 701)	0.495	14.253
(770, 577, 809)	0.025	30.900
(770, 770, 809)	0.495	46.268
(770, 577, 907)	0.040	30.098
(770, 770, 907)	0.500	46.374
(1026, 769, 1069)	0.040	71.009
(1026, 1026, 1069)	0.500	109.091
(1026, 769, 1187)	0.020	71.436
(1026, 1026, 1187)	0.500	108.871

Table 4. Results on $\mathsf{AF\text{-}PVC}_{two}$

(n, v, p)	$\mathsf{Adv}_{\mathcal{A}}^{\text{ind-priv}^{\text{coa}}}(\mathsf{AF\text{-}PVC}_{two}, \lambda)$	Timing (second)
(161, 10, 353)	0.500	0.854
(161, 10, 401)	0.495	0.849
(257, 13, 631)	0.500	3.549
(257, 13, 701)	0.500	3.648
(385, 15, 809)	0.500	11.064
(385, 15, 907)	0.500	10.336
(513, 17, 1069)	0.500	23.037
(513, 17, 1187)	0.500	22.666

Table 3, when $n = m$, the adversary's advantages validate the analysis in Theorem 6, which is based on the fact that $m > 3 \cdot \lambda + 1$. However, when $n > m$, the adversary's advantages are close to 0. This is because the rank of a given matrix from one of those two distributions is dependent on m if $m \leq 3 \cdot \lambda + 1$. Then, in this case, the proposed rank-based strategy is invalid for distinguishing the SSH distribution from the uniform distribution over $\mathbb{Z}_p^{n \times m}$, and the adversary must guess randomly.

Moreover, the timings of all the example experiments in Tables 2, 3, 4 and 5 show that our rank-based analyses for solving the decisional-SH problem and

Table 5. Results on $\mathsf{AF\text{-}PVC}_{single}$

(n, v, p)	$\mathsf{Adv}^{\text{ind-priv}^{\text{coa}}}_{\mathcal{A}}(\mathsf{AF\text{-}PVC}_{single}, \lambda)$	Timing (second)
(322, 12, 353)	0.500	2.976
(322, 12, 401)	0.500	3.026
(514, 15, 631)	0.500	11.499
(514, 15, 701)	0.500	11.509
(770, 18, 809)	0.500	36.407
(770, 18, 907)	0.500	34.566
(1026, 21, 1069)	0.500	82.436
(1026, 21, 1187)	0.500	83.199

breaking the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$ are really efficient. Specifically, for some small matrix sizes (e.g., $(n, m) = (161, 161)$ in Table 2), our analyses take less than a second.

5 Conclusions

In this paper, we propose an efficient analysis method for solving the decisional-WSH problem and decisional-SSH problem introduced by Atallah and Frikken [2]. Specifically, the strategy of our analysis takes advantage of the rank distribution of the matrix to distinguish between the samples from the WSH distribution (resp. the SSH distribution) $\chi(p)^{n\times m}$ and the samples from the uniform distribution over $\mathbb{Z}_p^{n\times m}$. The adversary's advantage of our analysis on a wide range of parameters is close to 0.5. Moreover, we employ a similar approach to break the privacy of $\mathsf{AF\text{-}PVC}_{two}$ and $\mathsf{AF\text{-}PVC}_{single}$. The analysis results show that both protocols are not IND-COA private.

Solving the Search Variant of the SH Problem? Our rank-based analysis can break the decisional-WSH assumption and decisional-SSH assumption, but this does not implies that we can also break the search versions efficiently. Actually, for breaking the search-WSH assumption and search-SSH assumption, our rank-based analysis may be regarded as a preprocessing step. To check whether a row of a matrix sampled from $\chi(p)^{n\times m}$ is a row vector from $\mathbf{d}_1, \ldots, \mathbf{d}_\ell$ or from $\mathbf{u}_1, \ldots, \mathbf{u}_\tau$, the adversary first replaces the row that needs to be tested by a row vector chosen from $\mathbb{Z}_p^m$ uniformly at random, and then computes the rank of the matrix where the tested row has been replaced. According to our analysis in Sect. 3.2, if the obtained rank increases (compared with the rank of the matrix sampled from $\chi(p)^{n\times m}$), this implies that the tested row is from the row vectors $\mathbf{d}_1, \ldots, \mathbf{d}_\ell$. The above procedure can be run at most $n-1$ times (with overwhelming probability) to reveals all the vectors $\mathbf{d}_1, \ldots, \mathbf{d}_\ell$. However, this result is not equivalent to finding $k_1, \ldots, k_\ell$ (or $\mathbf{A}$). Therefore, how to break the search variant of the SH assumption efficiently is an interesting open problem.

Acknowledgements. The author would like to thank the anonymous reviewers of CT-RSA 2019 for providing their helpful comments. This work was supported in part by the National Natural Science Foundation of China under Grant 61302161, in part by the Doctoral Fund, Ministry of Education, China, under Grant 20130181120076.

References

1. NTL 10.5.0 (2017). http://www.shoup.net/ntl/
2. Atallah, M.J., Frikken, K.B.: Securely outsourcing linear algebra computations. In: ASIACCS (2010)
3. Benjamin, D., Atallah, M.J.: Private and cheating-free outsourcing of algebraic computations. In: PST (2008)
4. Choi, S.G., Katz, J., Kumaresan, R., Cid, C.: Multi-client non-interactive verifiable computation. In: Sahai, A. (ed.) TCC 2013. LNCS, vol. 7785, pp. 499–518. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36594-2_28
5. Chung, K.-M., Kalai, Y., Vadhan, S.: Improved delegation of computation using fully homomorphic encryption. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 483–501. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14623-7_26
6. Fiore, D., Gennaro, R., Pastro, V.: Efficiently verifiable computation on encrypted data. In: CCS (2014)
7. Goldreich, O.: Foundations of Cryptography: Volume I Basic Tools. Cambridge University Press, Cambridge (2001)
8. Gennaro, R., Gentry, C., Parno, B.: Non-interactive verifiable computing: outsourcing computation to untrusted workers. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 465–482. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14623-7_25
9. Gordon, S.D., Katz, J., Liu, F.-H., Shi, E., Zhou, H.-S.: Multi-client verifiable computation with stronger security guarantees. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015. LNCS, vol. 9015, pp. 144–168. Springer, Heidelberg (2015). https://doi.org/10.1007/978-3-662-46497-7_6
10. Kaltofen, E., David Saunders, B.: On Wiedemann's method of solving sparse linear systems. In: Mattson, H.F., Mora, T., Rao, T.R.N. (eds.) AAECC 1991. LNCS, vol. 539, pp. 29–38. Springer, Heidelberg (1991). https://doi.org/10.1007/3-540-54522-0_93
11. Katti, R.S., Srinivasan, S.K., Vosoughi, A.: On the security of randomized arithmetic codes against ciphertext-only attacks. IEEE Trans. Inf. Forensics Secur. **6**(1), 19–27 (2011)
12. Laud, P., Pankova, A.: On the (im)possibility of privately outsourcing linear programming. In: CCSW (2013)
13. Linial, N., Weitz, D.: Random vectors of bounded weight and their linear dependencies (2000). http://www.drorweitz.com/ac/pubs/rand_mat.pdf
14. Shamir, A.: How to share a secret. Commun. ACM **22**(11), 612–613 (1979)
15. Salinas, S., Luo, C., Chen, X., Li, P.: Efficient secure outsourcing of large-scale linear systems of equations. In: INFOCOM (2015)
16. Salinas, S., Luo, C., Liao, W., Li, P.: Efficient secure outsourcing of large-scale quadratic programs. In: ASIACCS (2016)
17. Saunders, B.D., Youse, B.S.: Large matrix, small rank. In: ISSAC (2009)
18. Wiedemann, D.H.: Solving sparse linear equations over finite fields. IEEE Trans. Inf. Theory **32**(1), 54–62 (1986)