SCIENCE CHINA
Information Sciences

• LETTER •

March 2018, Vol. 61 039105:1–039105:3
https://doi.org/10.1007/s11432-017-9228-6

Verifiable random functions with Boolean function constraints

Qianwen WANG[1], Rongquan FENG[1] & Yan ZHU[2*]

[1]*School of Mathematical Sciences, Peking University, Beijing* 100871*, China;*
[2]*School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing* 100083*, China*

Received 10 February 2017/Accepted 19 July 2017/Published online 11 January 2018

Citation Wang Q W, Feng R Q, Zhu Y. Verifiable random functions with Boolean function constraints. Sci China Inf Sci, 2018, 61(3): 039105, https://doi.org/10.1007/s11432-017-9228-6

Dear editor,

Verifiable random functions (VRFs) [1] are cryptographic functions that behaves like a random function for producing a random string y for any variant x, but also allows for efficient verification of uniqueness of y for a valid witness z. In this work, we extend the concept of VRFs to a notion of a conditionally verifiable random function (CVRF), which is a new kind of VRFs over conditional constraints for input domain of VRFs. As a general example of CVRF, we present a practical construction, called a conditionally verifiable random function on Boolean function (CVRF-BF), to explore the common structure and method of CVRFs. Our CVRF-BF construction is designed on a two-layer structure based on full disjunctive normal form (full-DNF): the first layer consists of a collection of VRFs, one of which realizes the verification of correctness of each input component, perhaps with a simple constraint; and the second layer consists of a VRF with multi-input, which takes as input the first layer's evaluations, and verifies whether these evaluations adhere to a given access constraint. Moreover, we prove that our CVRF-BF construction satisfy three security properties: conditional provability, uniqueness, and pseudorandomness, under the hardness assumption of DBDHI problem. Our construction of CVRF-BF indicated that it is feasible to implement the proposed CVRF notion with various access constraints.

The decisional bilinear diffie-Hellman inversion (DBDHI) problem, used to prove the security of our construction, is described as follows:

Definition 1 (DBDHI problem [2])**.** Let $\mathbb{G}$ be a bilinear group of prime order p with G as a generator of it. Given the elements $G, G^a, G^{a^2}, \ldots, G^{a^\ell}$ as the input of ℓ-DBDHI problem, it distinguish $e(G,G)^{\frac{1}{a}}$ from random.

The ℓ-DBDHI assumption is that there is no such an algorithm which has advantage better than negligible in solving the ℓ-DBDHI problem in $\mathbb{G}$. We describe this assumption as follows:

Definition 2 ((ℓ,ϵ,t)-DBDHI assumption)**.** We say that (ℓ,ϵ,t)-DBDHI assumption holds in $\mathbb{G}$ if there is no t-time algorithm $\mathcal{A}$ solving the ℓ-DBDHI problem with at least ϵ advantage.

$$\Pr\left[b = b' \left| \begin{array}{c} G \leftarrow \mathbb{G}; a \leftarrow \mathbb{Z}_p^*; y_0 = e(G,G)^{\frac{1}{a}}; \\ y_1 \leftarrow \mathbb{G}_T; b \leftarrow_R \{0,1\}; \\ b' \leftarrow \mathcal{A}(G, G^a, G^{a^2}, \ldots, G^{a^\ell}, y_b); \end{array} \right.\right] \leqslant \frac{1}{2} + \epsilon.$$

The definition of CVRF. We now extend the concept of VRF [3] to the CVRF, which is a new kind of VRFs over conditional constraints for input domain of VRFs with conditional provability, uniqueness, and pseudorandomness properties. Roughly speaking, given the domain $\mathbb{X}$ of VRFs and a computable decision function $f : \mathbb{X} \to \{0,1\}$, there exists an efficient VRF scheme, $\mathcal{S} =$

* Corresponding author (email: zhuyan@ustb.edu.cn)

(F, G, Verify), such that for any input $x \in \mathbb{X}$, the verification algorithm holds $\text{Verify}(\text{pk}, x, y, z) = 1$ if and only if $f(x) = 1$, where $y \leftarrow F(\text{sk}, x)$ and $z \leftarrow G(\text{sk}, x)$. Here, we call the subset of $\mathbb{X}$ induced $f(x) = 1$ as access constraints or conditions. In nature, this kind of CVRFs is a family of verifiable random functions with multi-input, perhaps with large input sizes [4]. Exactly, our CVRF is defined as follows:

Definition 3 (CVRF)**.** Given a computable decision function $f : \{0,1\}^n \to W$ for a security parameter κ, we say that a VRF over f is a CVRF if there exist algorithms $(\text{Setup}, \text{GenFun}, \text{Prove}, \text{Verify})$ such that

(1) $\text{Setup}(1^\kappa, n)$ outputs a pair of keys (pk, sk) for a security parameter κ and a number n;

(2) $\text{GenFun}(\text{pk}, f)$ outputs a public key pk_f for a special $f(\cdot)$ and lets $\text{pk}_f \in \text{pk}$;

(3) $\text{Prove}(\text{sk}, x)$ outputs a pair $(F(\text{sk}, x), G(\text{sk}, x))$, where $F(\text{sk}, x)$ is the function value and $G(\text{sk}, x)$ is the proof of correctness; and

(4) $\text{Verify}(\text{pk}, x, y, z)$ verifies that $y = F(\text{sk}, x)$ using the proof $z = G(\text{sk}, x)$.

Formally, we require that the CVRF satisfies the following properties:

(1) Conditional provability. For all $(\text{pk}, \text{sk}) \in \text{Setup}(1^\kappa, n)$, $\text{pk}_f \in \text{GenFun}(\text{pk}, f)$ and all $x \in \{0,1\}^n$, if $(y, z) = \text{Prove}(\text{sk}, x)$, there exists a negligible polynomial μ such that

$$\begin{aligned}&\Pr[\text{Verify}(\text{pk}, x, y, z) = b \mid \exists\, b \in \{0,1\}, f(x) = b]\\ &\quad > 1 - u(\kappa). \end{aligned} \tag{1}$$

(2) Uniqueness. For all $(\text{pk}, \text{sk}) \in \text{Setup}(1^\kappa, n)$ and inputs $x \in \{0,1\}^{\text{in}(\kappa)}$, there does not exist a tuple (y_1, y_2, z_1, z_2) such that

$$\Pr\left[y_1 \neq y_2 \;\middle|\; \begin{matrix}\text{Verify}(\text{pk}, x, y_1, z_1) = 1,\\ \text{Verify}(\text{pk}, x, y_2, z_2) = 1\end{matrix}\right] \leqslant \mu(\kappa). \tag{2}$$

(3) Pseudorandomness. For all PPT distinguishers D, there exists a negligible polynomial μ such that

$$\begin{aligned}&\Pr[D(1^\kappa, F(\text{sk}, x)) = 1] - \Pr[D(1^\kappa, \{0,1\}^{\text{out}(\kappa)}) = 1]\\ &\quad \leqslant \frac{1}{2} + \mu(\kappa). \end{aligned} \tag{3}$$

To explore the feasible methods of CVRF, we proposed an instance of CVRF, called CVRF-BF, in which the computable decision function f is described as Boolean function.

Definition 4 (CVRF-BF)**.** Given a full-DNF Boolean function $f(x) = f(x_1 x_2 \cdots x_n)$ for $x \in \{0,1\}^n$, the system $\mathcal{S} = (\text{Setup}, \text{GenFun}, \text{Prove}, \text{Verify})$ is called as CVRF over Boolean function f, if $(\text{pk}, \text{sk}) \leftarrow \text{Setup}(1^\kappa, n)$ and $\text{pk}_f \leftarrow \text{GenFun}(\text{pk}, f)$ and $(y, z) \leftarrow \text{Prove}(\text{sk}, x) = (F(\text{sk}, x), G(\text{sk}, x))$. Then there exists the algorithm Verify such that for any input x, the equation (4) holds.

$$\text{Verify}(\text{pk}_f, x, y, z) = \begin{cases} 1, & f(x) = 1,\\ 0, & f(x) = 0. \end{cases} \tag{4}$$

Our construction of CVRF-BF. Let $\mathbb{S} = (p, \mathbb{G}, \mathbb{G}_T, e, G, H)$ is a bilinear map group system under the security parameter κ, where $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ is a bilinear map over the groups $\mathbb{G}, \mathbb{G}_T$ of order p, and $\mathbb{G}$ contains two generators $G, H \in \mathbb{G}$. Given a monotone Boolean function $f(x)$, we construct a VRF-BF on $\mathbb{S}$ as follows:

• $\text{Setup}(1^\kappa, n)$. It generates the public key and the secret key, as follows:

(1) Chooses $2n$ integers $\lambda_1, \ldots, \lambda_n, \psi_1, \ldots, \psi_n \in_R \mathbb{Z}_p$ to define $G_{i,0} = G^{\lambda_i}$ and $G_{i,1} = G^{\psi_i} \in \mathbb{G}$ for $i = 1, 2, \ldots, n$, each of which corresponds to $x_i = 0$ or 1;

(2) Picks a random integer $\xi \in_R \mathbb{Z}_p$ and sets $H' = H^\xi$;

(3) Chooses n random $r_1, \ldots, r_n \in_R \mathbb{Z}_p$, sets $H_i = H^{r_i}$ and $H_i' = (H')^{r_i} = H^{r_i \xi} = (H^\xi)^{r_i}$, for $i = 1, 2, \ldots, n$;

(4) Defines the secret key as $\text{sk} = (\xi)$ and the public key as

$$\text{pk} = (\mathbb{S}, H', \{H_i, H_i', (G_{i,0}, G_{i,1})\}_{i \in \{1,2,\ldots,n\}}).$$

• $\text{GenFun}(\text{pk}, f)$. It proceeds three steps:

(1) Given a Boolean function $f(x)$ and $x = (x_1, x_2, \ldots, x_n) \in \{0,1\}^n$, transfers it into a disjunctive normal form (DNF) $f(x) = \bigvee_{k=1}^m \text{conj}_k$, where any $\text{conj}_k = \nabla x_1 \wedge \cdots \wedge \nabla x_n$ and ∇ denotes $\neg$ or empty.

(2) For each clause conj_k, where $k \in \{1, 2, \ldots, m\}$, it sets a random polynomial $g_k(X) = \sum_{i=0}^{m_k} a_{i,k} X^i$, where $a_{0,k} = \xi$ and $a_{i,k} \in_R \mathbb{Z}_p^*$ for $i \in \{1, 2, \ldots, m_k\}$. Furthermore, only when $\text{conj}_k = 1$, we compute the value of $g_k(X)$. And for each conj_k we have different $g_k(X)$ with different k. In order to mark simply, we sometimes use $g(X)$ instead of $g_k(X)$.

(3) This algorithm chooses $v_1, \ldots, v_n \in_R \mathbb{Z}_p^*$ and computes $\{(\tilde{H}_1^{(k)}, \tilde{G}_{1,x_1}^{(k)}), \ldots, (\tilde{H}_n^{(k)}, \tilde{G}_{n,x_n}^{(k)})\}$, where $\tilde{H}_i^{(k)} = H^{g_k(v_i)}$ and $\tilde{G}_{i,x_i}^{(k)} = G_{i,x_i}^{g_k(v_i)}$ for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, m$.

(4) After processing all $\text{conj}_1, \ldots, \text{conj}_m$, this algorithm outputs

$$\text{pk}_f = (v_i, \{\tilde{H}_i^{(k)}, \tilde{G}_{i,x_i}^{(k)}\}_{k \in \{1,2,\ldots,m\}})_{i \in \{1,2,\ldots,n\}}.$$

Note that, $\text{pk}_f \in \text{pk}$, and there exist m random polynomials $g_k(\cdot)$ for m clauses in $f(\cdot)$.

• $\text{Prove}(\text{sk}, x)$. For any $x = (x_1, x_2, \ldots, x_n) \in \{0,1\}^n$, the result (y, z) is outputted by computing the following equations:

$$\begin{cases} y = F(\mathrm{sk}, x) = e(G, H)^{\frac{\xi}{\xi+x}}, \\ z = G(\mathrm{sk}, x) = (z_1, z_2, \ldots, z_n) \\ \quad = G^{\frac{1}{\xi+x}} \cdot (G_{1,x_1}^{r_1}, G_{2,x_2}^{r_2}, \ldots, G_{n,x_n}^{r_n}), \end{cases} \tag{5}$$

as the algorithm's outputs and witness of correctness transferred to the verifier. Finally, this algorithm outputs (y, z).

• Verify(pk, x, y, z). This algorithm is divided into these processes:

(1) In terms of $f(x) = \bigvee_{k=1}^{m} \mathrm{conj}_k$, this algorithm finds a clause conj_k to satisfy $\mathrm{conj}_k = 1$ for $x = (x_1, x_2, \ldots, x_n)$; otherwise it outputs 0.

(2) Next, in order to verify whether $z = (z_1, \ldots, z_n)$ was computed correctly, it needs to check

$$e(z_i, H' \cdot H^x) = e(G, H) \cdot e(G_{i,x_i}, H_i' \cdot H_i^x), \tag{6}$$

for all $i \in \{1, 2, \ldots, n\}$. Given the valid $(z_1, z_2, \ldots, z_n)$, it makes use of the public key pk and compute the corresponding y_i by

$$y_i = e(z_i, \tilde{H}_i) \cdot e(\tilde{G}_{i,x_i}, H_i)^{-1}. \tag{7}$$

(3) To verify whether $y = H(\mathrm{sk}, x)$ was computed correctly, the algorithm sets Lagrangian interpolation coefficient $\gamma_i = \prod_{1 \leqslant j \leqslant n, j \neq i} \frac{v_j}{v_j - v_i} \pmod p$. Then it needs to check the equations

$$y = \prod_{i=1}^{n} y_i^{\gamma_i} = e(G, H)^{\frac{\xi}{\xi+x}}. \tag{8}$$

(4) Finally, it outputs 1, if and only if the equation above has been checked.

According to the definition of CVRFs, we prove that our CVRF-BF scheme satisfies the security properties, including conditional provability, uniqueness, and pseudorandomness [1)], as follows:

Theorem 1 (Conditional provability)**.** The CVRF-BF scheme is in line with the conditional provability, that is, for all $(\mathrm{pk}, \mathrm{sk}) \in \mathrm{Setup}(1^\kappa, n)$, $\mathrm{pk}_f \in \mathrm{GenFun}(\mathrm{pk}, f)$ and all $x \in \{0,1\}^{\mathrm{in}(\kappa)}$, if $(y, z) = \mathrm{Prove}(\mathrm{sk}, x)$, there exists a negligible polynomial μ such that

$$\Pr[\mathrm{Verify}(\mathrm{pk}, x, y, z) = b \mid \exists\, b \in \{0,1\}, f(x) = b] > 1 - \mu(\kappa). \tag{9}$$

Theorem 2 (Uniqueness)**.** The CVRF-BF scheme we constructed complies with the definition of uniqueness, that is, in the groups $\mathbb{G}, \mathbb{G}_T$ of order p and a bilinear map $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$, for all $(\mathrm{pk}, \mathrm{sk}) \in \mathrm{Setup}(1^\kappa, n)$ and inputs $x \in \{0,1\}^{in(\kappa)}$, there does not exist a tuple $(y^{(1)}, y^{(2)}, z^{(1)}, z^{(2)})$ such that

$$\Pr\left[y^{(1)} \neq y^{(2)} \;\middle|\; \begin{array}{l} \mathrm{Verify}(\mathrm{pk}, x, y^{(1)}, z^{(1)}) = 1, \\ \mathrm{Verify}(\mathrm{pk}, x, y^{(2)}, z^{(2)}) = 1 \end{array} \right] \leqslant \mu(\kappa). \tag{10}$$

Theorem 3 (Pseudorandomnes)**.** Suppose the (ℓ, ϵ, t)-DBDHI assumption holds in a bilinear group $\mathbb{G}(|\mathbb{G}| = p)$, the outputs of our CVRF-BF (Setup, GenFun, Prove, Verify) is (l', ϵ', t')-indistinguishable under the chosen input attack (IND-CIA), or we say that our CVRF-BF scheme is a verifiable random function with running time t' and negligible advantage $\epsilon' = \frac{\epsilon}{2^n}$ through l'-time valid queries for Prove$(\cdot)$.

Conclusion. At present, constraint-based encryption, e.g., attribute-based encryption (ABE) and functional encryption (FE), has been an inevitable trend for conditional information sharing in terms of access constraints [5]. The proposed notion of CVRF is a basic tool for analyzing the security of constraint-based encryption. Moreover, our CVRF is a good candidate to implement more secure access constraints because the properties hold by VRFs, such as provability, uniqueness, and pseudo-randomness, are exactly what the security analysis of constraint verifications needed.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61370187, 61472032), NSFC-Genertec Joint Fund For Basic Research (Grant No. U1636104) and Joint Research Fund for Overseas Chinese Scholars and Scholars in Hong Kong and Macao (Grant No. 61628201).

Supporting information Appendix A. The supporting information is available online at info.scichina.com and link.springer.com. The supporting materials are published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

1 Micali S, Rabin M, Vadhan S. Verifiable random functions. In: Proceedings of the 40th Annual Symposium on Foundations of Computer Science. New York: IEEE, 1999. 120–130
2 Dodis Y, Yampolskiy A. A verifiable random function with short proofs and keys. In: Public Key Cryptography. New York: Springer, 2005. 416–431
3 Kuchta V, Manulis M. Unique aggregate signatures with applications to distributed verifiable random functions. In: Proceedings of the International Conference on Cryptology and Network Security. New York: Springer, 2013. 251–270
4 Hofheinz D, Jager T. Verifiable random functions from standard assumptions. In: Proceedings of the Theory of Cryptography Conference. New York: Springer, 2016. 336–362
5 Antezana J, Marzo J, Olsen J F. Zeros of random functions generated with de branges kernels. Int Math Res Not, 2016, 2016: rnw078

1) The complete proofs of the above-mentioned theorems could be found in Appendix A.