Bandersnatch VRF-AD Specification

Davide Galassi Seyed Hosseini

9 Mar 2025 - Draft 26

Abstract

This specification delineates the framework for a Verifiable Random Function with Additional Data (VRF-AD), a cryptographic construct that augments a standard VRF by incorporating auxiliary information into its signature. We're going to first provide a specification to extend IETF's ECVRF as outlined in RFC-9381 [1], then we describe a variant of the Pedersen VRF originally introduced by BCHSV23 [2], which serves as a fundamental component for implementing anonymized ring signatures as further elaborated by VG24 [3]. This specification provides detailed insights into the usage of these primitives with Bandersnatch, an elliptic curve constructed over the BLS12-381 scalar field specified in MSZ21 [4].

1. Preliminaries

1.1. Common definitions

- $\mathbf{G}$: Bandersnatch curve cyclic group of prime order r.
- $\mathbf{F}$: Scalar field of prime order r (i.e. $\mathbb{Z}_r$).
- $\mathbf{\Sigma}^k$: Octets strings with length $k \in \mathbb{N}$ ($*$ for arbitrary length).
- $G \in \mathbf{G}$: Prime order group generator.
- $x \in \mathbf{F}$: Secret key scalar.
- $Y \in \mathbf{G}$: Public key point defined as $x \cdot G$.
- $i \in \mathbf{\Sigma}^*$: VRF input data.
- $I \in \mathbf{G}$: VRF input point.
- $O \in \mathbf{G}$: VRF output point.
- $o \in \mathbf{\Sigma}^k$: VRF output hash.

1.2. VRF Input

An arbitrary length octet-string provided by the user to generate some unbiasable verifiable random output.

1.3. VRF Input Point

A point in $\mathbf{G}$ generated from VRF input octet-string using the *Elligator 2 hash-to-curve* algorithm as described by section 6.8.2 of RFC-9380 [5].

$$I \leftarrow \texttt{hash\_to\_curve}(i)$$

1.4. VRF Output Point

A point generated from VRF input point and secret key scalar.

$$O \leftarrow x \cdot I$$

1.4. VRF Output

A fixed-length octet string produced from the VRF output point using the *output-to-hash* procedure, which is *proof-to-hash* method described in Section 5.2 of RFC-9381, but with a specific focus on the output point component (referred to as *Gamma* in RFC-9381) extracted from the output point proof bundle.

$$o \leftarrow \texttt{output\_to\_hash}(O)$$

1.5 Additional Data

An arbitrary length octet-string provided by the user to be signed together with the generated VRF output. This data doesn't influence the produced VRF output.

1.6. VRF-AD

Regardless of the specific scheme, a *Verifiable Random Function with Additional Data (VRF-AD)* can be concisely represented by three primary functions:

- $prove(x, i, ad) \mapsto \Pi$
- $verify(y, i, ad, \Pi) \mapsto (\top \mid \bot)$
- $output(\Pi) \mapsto o$

Here:

- $y \leftarrow \texttt{serialize\_compressed}(Y)$, where Y is the public key corresponding to the private key used for proving.

- $\Pi \leftarrow$ `encode_compressed`$((O, \pi))$, where π is the proof specific to the underlying scheme, and O represents the VRF output point.

1.7. Challenge Procedure

Challenge construction mostly follows the procedure given in section 5.4.3 of RFC-9381 [1] with some tweaks to add additional data.

Input:

- $\bar{P} \in \mathbf{G}^n$: Sequence of n points.
- $ad \in \mathbf{\Sigma}^*$: Additional data octet-string.

Output:

- $c \in \mathbf{F}$: Challenge scalar.

Steps:

1. $str_0 \leftarrow$ `suite_string` $\| \; 0x02$
2. $str_i \leftarrow str_{i-1} \; \|$ `point_to_string`$(P_{i-1}),\ i = 1 \dots n$
3. $h \leftarrow$ `hash`$(str_n \; \| \; ad \; \| \; 0x00)$
4. $c \leftarrow$ `string_to_int`$(h_{0 \dots cLen-1})$

With `point_to_string`, `string_to_int` and `hash` as defined in section 2.1.

2. IETF VRF

Based on IETF RFC-9381 which is extended with the capability to sign additional user data (ad).

2.1. Configuration

Configuration is given by following the *"cipher suite"* guidelines defined in section 5.5 of RFC-9381.

- `suite_string` = `"Bandersnatch_SHA-512_ELL2"`.
- The EC group $\mathbf{G}$ is the prime subgroup of the Bandersnatch elliptic curve, in Twisted Edwards form, with finite field and curve parameters as specified in MSZ21. For this group, `fLen` = `qLen` = 32 and `cofactor` = 4.
- The prime subgroup generator $G \in \mathbf{G}$ is defined as follows:

  $$G_x = 18886178867200960497001835917649091219057080094937609519140440539760939937304$$

  $$G_y = 19188667384257783945677642223292697773471335439753913231509108946878080696678$$

  - Compressed: `0x664197ccb667315e6064e4ee81ad8c3586d5dcba508b7d150f3e12da9e666c2a`

- `cLen` = 32.
- The public key generation primitive is $pk = sk \cdot G$, with sk the secret key scalar and G the group generator. In this cipher suite, the secret scalar `x` is equal to the secret key `sk`.
- `encode_to_curve_salt` = `""` (empty - no salt)
- The `ECVRF_nonce_generation` function is specified in section 5.4.2.2 of RFC-9381.
- The `int_to_string` function encodes into the 32 octets little endian representation.
- The `string_to_int` function decodes from the 32 octets little endian representation eventually reducing modulo the prime field order.
- The `point_to_string` function converts a point in $\mathbf{G}$ to an octet-string using compressed form. The y coordinate is encoded using `int_to_string` function and the most significant bit of the last octet is used to keep track of x sign. This implies that `ptLen = flen = 32`.
- The `string_to_point` function converts an octet-string to a point on $\mathbf{G}$. The string most significant bit is removed to recover the x coordinate as function of y, which is first decoded from the rest of the string using `int_to_string` procedure. This function MUST outputs "INVALID" if the octet-string does not decode to a point on the prime subgroup $\mathbf{G}$.
- The hash function `hash` is SHA-512 as specified in RFC-6234 [6], with `hLen` = 64.
- The `ECVRF_encode_to_curve` function uses *Elligator2* method as described in section 6.8.2 of RFC-9380 and in section 5.4.1.2 of RFC-9381, with parametrized with `h2c_suite_ID_string` = `"Bandersnatch_XMD:SHA-512_ELL2_RO_"` and domain separation tag `DST = "ECVRF_"` ‖ `h2c_suite_ID_string` ‖ `suite_string`.

2.2. Prove

Input:

- $x \in \mathbf{F}$: Secret key
- $I \in \mathbf{G}$: VRF input point
- $ad \in \mathbf{\Sigma}^*$: Additional data octet-string.

Output:

- $O \in \mathbf{G}$: VRF output point
- $\pi \in (\mathbf{F}, \mathbf{F})$: Schnorr-like proof

Steps:

1. $O \leftarrow x \cdot I$
2. $Y \leftarrow x \cdot G$
3. $k \leftarrow \texttt{nonce}(x, I)$
4. $c \leftarrow \texttt{challenge}(Y, I, O, k \cdot G, k \cdot I, ad)$
5. $s \leftarrow k + c \cdot x$
6. $\pi \leftarrow (c, s)$

Externals:

- `nonce`: refer to section 5.4.2.2 of RFC-9381.
- `challenge`: refer to section 1.6 of this specification.

2.3. Verify

Input:

- $Y \in \mathbf{G}$: Public key
- $I \in \mathbf{G}$: VRF input point
- $ad \in \mathbf{\Sigma}^*$: Additional data octet-string.
- $O \in \mathbf{G}$: VRF output point
- $\pi \in (\mathbf{F}, \mathbf{F})$: Schnorr-like proof

Output:

- $\theta \in \{\top, \bot\}$: $\top$ if proof is valid, $\bot$ otherwise.

Steps:

1. $(c, s) \leftarrow \pi$
2. $U \leftarrow s \cdot G - c \cdot Y$
3. $V \leftarrow s \cdot I - c \cdot O$
4. $c' \leftarrow \texttt{challenge}(Y, I, O, U, V, ad)$
5. $\theta \leftarrow \top$ if $c = c'$ else $\bot$

Externals:

- `challenge`: as defined for *Prove*

3. Pedersen VRF

Pedersen VRF resembles IETF EC-VRF but replaces the public key with a Pedersen commitment to the secret key, which makes this VRF useful in anonymized ring proofs.

The scheme proves that the output has been generated with a secret key associated with a blinded public key (instead of the public key). The blinded public key is a cryptographic commitment to the public key, and it can be unblinded to prove that the output of the VRF corresponds to the public key of the signer.

This specification mostly follows the design proposed by BCHSV23 [2] in section 4 with some details about blinding base point value and challenge generation procedure.

3.1. Configuration

Pedersen VRF is configured for prime subgroup $\mathbf{G}$ of Bandersnatch elliptic curve E, in Twisted Edwards form, defined in MSZ21 [4] with *blinding base* $B \in \mathbf{G}$ defined as follows:

$$B_x = 6150229251051246713677296363717454238956877613358614224171740096471278798312$$

$$B_y = 28442734166467795856797249030329035618871580593056783094884474814923353898473$$

- Compressed: `0xe93da06b869766b158d20b843ec648cc68e0b7ba2f7083acf0f154205d04e23e`

For all the other configurable parameters and external functions we adhere as much as possible to the Bandersnatch cipher suite for IETF VRF described in section 2.1 of this specification.

3.2. Prove

Input:

- $x \in \mathbf{F}$: Secret key
- $b \in \mathbf{F}$: Secret blinding factor
- $I \in \mathbf{G}$: VRF input point
- $ad \in \mathbf{\Sigma}^*$: Additional data octet-string.

Output:

- $O \in \mathbf{G}$: VRF output point
- $\pi \in (\mathbf{G}, \mathbf{G}, \mathbf{G}, \mathbf{F}, \mathbf{F})$: Pedersen proof

Steps:

1. $O \leftarrow x \cdot I$
2. $k \leftarrow \texttt{nonce}(x, I)$
3. $k_b \leftarrow \texttt{nonce}(b, I)$
4. $\bar{Y} \leftarrow x \cdot G + b \cdot B$
5. $R \leftarrow k \cdot G + k_b \cdot B$

6. $O_k \leftarrow k \cdot I$
7. $c \leftarrow \texttt{challenge}(\bar{Y}, I, O, R, O_k, ad)$
8. $s \leftarrow k + c \cdot x$
9. $s_b \leftarrow k_b + c \cdot b$
10. $\pi \leftarrow (\bar{Y}, R, O_k, s, s_b)$

3.3. Verify

Input:

- $I \in \mathbf{G}$: VRF input point
- $ad \in \mathbf{\Sigma}^*$: Additional data octet-string.
- $O \in \mathbf{G}$: VRF output point
- $\pi \in (\mathbf{G}, \mathbf{G}, \mathbf{G}, \mathbf{F}, \mathbf{F})$: Pedersen proof

Output:

- $\theta \in \{\top, \bot\}$: $\top$ if proof is valid, $\bot$ otherwise.

Steps:

1. $(\bar{Y}, R, O_k, s, s_b) \leftarrow \pi$
2. $c \leftarrow \texttt{challenge}(\bar{Y}, I, O, R, O_k, ad)$
3. $\theta_0 \leftarrow \top$ if $O_k + c \cdot O = I \cdot s$ else $\bot$
4. $\theta_1 \leftarrow \top$ if $R + c \cdot \bar{Y} = s \cdot G + s_b \cdot B$ else $\bot$
5. $\theta = \theta_0 \wedge \theta_1$

4. Ring VRF

Anonymized ring VRF based of [Pedersen VRF] and Ring Proof as proposed in VG24.

4.1. Configuration

Ring proof is configured to work together with Pedersen VRF as presented in this specification.

The following configuration should be applied to specialize VG24 in order to instance the concrete scheme.

- **Groups and Fields**:
  - $\mathbb{G}_{\mathbb{K}}$: BLS12-381 prime order subgroup.

  - $\mathbb{F}$: BLS12-381 scalar field.
  - J: Bandersnatch curve defined over $\mathbb{F}$.
- **Polynomial Commitment Scheme**
  - KZG with SRS derived from Zcash powers of tau ceremony.
- **Fiat-Shamir Transform**
  - `ark-transcript`.
  - Begin with empty transcript and "ring-proof" label.
  - Push R to the transcript after instancing.
  - TODO: Specify the order and how parameters are added to the transcript as we progress the protocol.
- Accumulator seed point in Twisted Edwards form:

  $$\mathrm{S}_x = 37805570861274048643170021838972902516980894313648523898085159469000338764576$$

  $$\mathrm{S}_y = 14738305321141000190236674389841754997202271418876976886494444739226156422510$$

  - Compressed: `0x6e5574f9077fb76c885c36196a832dbadd64142d305be5487724967acf9595a0`
- Padding point in Twisted Edwards form:

  $$\square_x = 26287722405578650394504321825321286533153045350760430979437739593351290020913$$

  $$\square_y = 19058981610000167534379068105702216971787064146691007947119244515951752366738$$

  - Compressed: `0x92ca79e61dd90c1573a8693f199bf6e1e86835cc715cdcf93f5ef222560023aa`

  A point with unknown discrete logarithm derived using the `ECVRF_encode_to_curve` function as described in IETF suite [Configuration] section with input the string: `"ring-proof-pad"`.
- Polynomials domain ($\langle\omega\rangle = \mathbb{D}$) generator:

  $$\omega = 49307615728544765012166121802278658070711169839041683575071795236746050763237$$

- $|\mathbb{D}| = 2048$

4.2. Prove

Input:

- $x \in \mathbf{F}$: Secret key
- $P \in ?$: Ring prover
- $k \in \mathbb{N}_k$: prover public key position within the ring
- $b \in \mathbf{F}$: Secret blinding factor

- $I \in \mathbf{G}$: VRF input point
- $ad \in \mathbf{\Sigma}^*$: Additional data octet-string.

Output:

- $O \in \mathbf{G}$: VRF output point
- $\pi_p \in (\mathbf{G}, \mathbf{G}, \mathbf{G}, \mathbf{F}, \mathbf{F})$: Pedersen proof
- $\pi_r \in ((G_1)^4, (\mathbf{F})^7, G_1, \mathbf{F}, G_1, G_1)$: Ring proof

Steps:

1. $(O, \pi_p) \leftarrow Pedersen.prove(x, b, k, I, ad)$
2. $\pi_r \leftarrow Ring.prove(P, b)$

4.3. Verify

Input:

- $V \in (G_1)^3$: Ring verifier (pre-processed commitment).
- $I \in \mathbf{G}$: VRF input point.
- $O \in G$: VRF output point.
- $ad \in \mathbf{\Sigma}^*$: Additional data octet-string.
- $\pi_p \in (\mathbf{G}, \mathbf{G}, \mathbf{G}, \mathbf{F}, \mathbf{F})$: Pedersen proof
- $\pi_r \in ((G_1)^4, (\mathbf{F})^7, G_1, \mathbf{F}, G_1, G_1)$: Ring proof

Output:

- $\theta \in \{\top, \bot\}$: $\top$ if proof is valid, $\bot$ otherwise.

Steps:

1. $\theta_0 = Pedersen.verify(I, ad, O, \pi_p)$
2. $(\bar{Y}, R, O_k, s, s_b) \leftarrow \pi_p$
3. $\theta_1 = Ring.verify(V, \pi_r, \bar{Y})$
4. $\theta \leftarrow \theta_0 \wedge \theta_1$

Appendix A

The test vectors in this section were generated using `ark-ec-vrfs` libraries revision `d90e180`.

A.1. IETF VRF Test Vectors

Schema:

```
sk (x): Secret key,
pk (Y): Public key,
in (alpha): Input octet-string,
ad: Additional data octet-string,
h (I): VRF input point,
gamma (O): VRF output point,
out (beta): VRF output octet string,
proof_c: Proof 'c' component,
proof_s: Proof 's' component,
```

Vector 1

```
3d6406500d4009fdf26045460936659l1e753f2213570a29521fd88bc30ede18,
a1b1da71cc4682e159b7da23050d8b6261eb11a3247c89b07ef56ccd002fd38b,
-,
-,
c5eaf38334836d4b10e05d2c1021959a917e08eaf4eb46a8c4c8d1bec04e2c00,
e7aa5154103450f0a0525a36a441f827296ee489ef30ed8787cff8df1bef223f,
fdeb377a4ffd7f95ebe48e5b43a88d069ce62188e49493500315ad55ee04d744
..2b93c4c91d5475370e9380496f4bc0b838c2483bce4e133c6f18b0adbb9e4722,
439fd9495643314fa623f2581f4b3d7d6037394468084f4ad7d8031479d9d101,
828bedd2ad95380b11f67a05ea0a76f0c3fef2bee9f043f4dffdddde09f55c01,
```

Vector 2

```
8b9063872331dda4c3c282f7d813fb3c13e7339b7dc9635fdc764e32cc57cb15,
5ebfe047f421e1a3e1d9bbb163839812657bbb3e4ffe9856a725b2b405844cf3,
0a,
-,
8c1d1425374f01d86b23bfeab770c60b58d2eeb9afc5900c8b8a918d09a6086b,
60f32f5ad3e9694b82ccc0a735edb2f940f757ab333cc5f7b0a41158b80f574f,
44f3728bc5ad550aeeb89f8db340b2fceffc946be3e2d8c5d99b47c1fce344b3
..c7fcee223a9b29a64fe4a86a9994784bc165bb0fba03ca0a493f75bee89a0946,
8aa1c755a00a6a25bdecda197ee1b60a01e50787bd10aa976133f4c39179330e,
18c74ffd67e6abc658e2d05ecd3101ddc0c33623823f2395538cf8d39e654f12,
```

Vector 3

```
6db187202f69e627e432296ae1d0f166ae6ac3c1222585b6ceae80ea07670b14,
9d97151298a5339866ddd3539d16696e19e6b68ac731562c807fe63a1ca49506,
-,
0b8c,
c5eaf38334836d4b10e05d2c1021959a917e08eaf4eb46a8c4c8d1bec04e2c00,
```

```
67a348e256d908eb695d15ee0d869efef2bcf9f0fea646e788f967abbc0464dd,
edde0178045133eb03ef4d1ad8b978a56ee80ec4eab8830d6bc6c08003138841
..6657d3c449d9398cc4385d1c8a2bb19bcf61ff086e5a6c477a0302ce270d1abf,
aec4d1cf308cb4cb400190350e69f4fb309255aa738fff5a6ac4ced7538fce03,
54e5d38a76f309ce63ca82465160abd8d75b78805a0b499e60c26436de4a8e01,
```

Vector 4

```
b56cc204f1b6c2323709012cb16c72f3021035ce935fbe69b600a88d842c7407,
dc2de7312c2850a9f6c103289c64fbd76e2ebd2fa8b5734708eb2c76c0fb2d99,
73616d706c65,
-,
672e8c7a8e6d3eca67df38f11d50f3d7dbb26fa8e27565a5424e6f8ac4555dcc,
4d3e0524fc59374f1fdad8e471c695469b45ecf69c1de85c6c1230e888dd4cbe,
36127f8aee7c61048984f0a208bf6d334db9dacbeeeef9ff2d17117e81232832
..1462eb3ef602f5911d77ab11f815eb4154ba95c934e414198ef000a61b4de31a,
b72598f235145a377911caa794ba85820173c4c49b7be3b05d847b2c753e0311,
e8e34ad3131388a88eb7f80bd874f3421c378d4ad45911c4bc16e4cdc17b5716,
```

Vector 5

```
da36359bf1bfd1694d3ed359e7340bd02a6a5e54827d94db1384df29f5bdd302,
decb0151cbeb49f76f10419ab6a96242bdc87baac8a474e5161123de4304ac29,
42616e646572736e61746368207665637466f72,
-,
4315192d2ce9e52ceb449a6b4da7f7e6636e53592c7f5e236763e21e9bac24c7,
9508104b820469687488d83f729288d9f70fc0523318beff44a47da10d490b3c,
4ee61f3c000544aa48c565e143e05c6501a623bdbf02a0a408b97433660b4907
..715f75890cc0e45cdd7116e3da15b15c3c637782e8e05d05c0d5895e5fe583d1,
ad6af59b4b84f18187c694ef374687d13517cb53508ff9dafa37d0c759e9601c,
4c1269d9d161dabd082fc606af979eca7f6c3ab68e78261dc6fb9fbbb98c9704,
```

Vector 6

```
da36359bf1bfd1694d3ed359e7340bd02a6a5e54827d94db1384df29f5bdd302,
decb0151cbeb49f76f10419ab6a96242bdc87baac8a474e5161123de4304ac29,
42616e646572736e61746368207665637466f72,
1f42,
4315192d2ce9e52ceb449a6b4da7f7e6636e53592c7f5e236763e21e9bac24c7,
9508104b820469687488d83f729288d9f70fc0523318beff44a47da10d490b3c,
4ee61f3c000544aa48c565e143e05c6501a623bdbf02a0a408b97433660b4907
..715f75890cc0e45cdd7116e3da15b15c3c637782e8e05d05c0d5895e5fe583d1,
4fa53519bd9d17acae4d1021416557d11b84dd4670b563770c14eb98161eaa08,
0f7f9bee9077427f547e69b919cf8d63823c14b20085fd9516768e0f5e3d3f0e,
```

Vector 7

```
35b877a25c394512292b82bdf8468e98eaf03c79c7fc9d53546dadc5fb75b500,
b0e1f208f9d6e5b310b92014ea7ef3011e649dab038804759f3766e01029d623,
42616e646572736e6174636820766563746f72,
1f42,
4315192d2ce9e52ceb449a6b4da7f7e6636e53592c7f5e236763e21e9bac24c7,
6d1dd583bea262323c7dc9e94e57a472e09874e435719010eeafae503c433f16,
09106f062ac07846f3f841f64765527b333575143483855d633f99ccc2e8e306
..e6239ff79a1272cff931e8d0ac6c390328486329118ad40a18b85184da1837ff,
6dbeeab9648505fa6a95de52d611acfbb2febacc58cdc7d0ca45abd8c952ef12,
ce7f4a2354a6c3f97aee6cc60c6aa4c4430b12ed0f0ef304b326c776618d7609,
```

A.2. Pedersen VRF Test Vectors

Schema:

```
sk (x): Secret key,
pk (Y): Public key,
in (alpha): Input octet-string,
ad: Additional data octet-string,
h (I): VRF input point,
gamma (O): VRF output point,
out (beta): VRF output octet string,
blinding: Blinding factor,
proof_pk_com (Y^-): Public key commitment,
proof_r: Proof 'R' component,
proof_ok: Proof 'O_k' component,
proof_s: Proof 's' component,
proof_sb: Proof 's_b' component
```

Vector 1

```
3d6406500d4009fdf2604546093665911e753f2213570a29521fd88bc30ede18,
a1b1da71cc4682e159b7da23050d8b6261eb11a3247c89b07ef56ccd002fd38b,
-,
-,
-,
c5eaf38334836d4b10e05d2c1021959a917e08eaf4eb46a8c4c8d1bec04e2c00,
e7aa5154103450f0a0525a36a441f827296ee489ef30ed8787cff8df1bef223f,
fdeb377a4ffd7f95ebe48e5b43a88d069ce62188e49493500315ad55ee04d744
..2b93c4c91d5475370e9380496f4bc0b838c2483bce4e133c6f18b0adbb9e4722,
01371ac62e04d1faaadbebaa686aaf122143e2cda23aacbaa4796d206779a501,
3b21abd58807bb6d93797001adaacd7113ec320dcf32d1226494e18a57931fc4,
8123054bfdb6918e0aa25c3337e6509eea262282fd26853bf7cd6db234583f5e,
ac57ce6a53a887fc59b6aa73d8ff0e718b49bd9407a627ae0e9b9e7c5d0d175b,
0d379b65fb1e6b2adcbf80618c08e31fd526f06c2defa159158f5de146104c0f,
```

e2ca83136143e0cac3f7ee863edd3879ed753b995b1ff8d58305d3b1f323630b,

Vector 2

8b9063872331dda4c3c282f7d813fb3c13e7339b7dc9635fdc764e32cc57cb15,
5ebfe047f421e1a3e1d9bbb163839812657bbb3e4ffe9856a725b2b405844cf3,
0a,
-,
-,
8c1d1425374f01d86b23bfeab770c60b58d2eeb9afc5900c8b8a918d09a6086b,
60f32f5ad3e9694b82ccc0a735edb2f940f757ab333cc5f7b0a41158b80f574f,
44f3728bc5ad550aeeb89f8db340b2fceffc946be3e2d8c5d99b47c1fce344b3
..c7fcee223a9b29a64fe4a86a9994784bc165bb0fba03ca0a493f75bee89a0946,
99ff52abf49d67c4303ac4a8a00984d04c06388f5f836ebd37031f0e76245815,
c1322e7a65b83996c25e37a84e36598333b0d417619242c0cb3d9d972edde848,
7a4363e0bf9cd18317287d681ab05704982b0088ce373f696dbdf3909a902b36,
fc8770c209212640742d53e2f40e5c30fffae574f90fdc670ff11a1127586c03,
93f7c9d73eec05e500b758f645a2967e62b2206e57eff5f9b99bfc71812e620d,
c864de36e0b428f6fb4ef470f94ec9601716cb26ad96f3359e4a1ec110794a0b,

Vector 3

6db187202f69e627e432296ae1d0f166ae6ac3c1222585b6ceae80ea07670b14,
9d97151298a5339866ddd3539d16696e19e6b68ac731562c807fe63a1ca49506,
-,
-,
0b8c,
c5eaf38334836d4b10e05d2c1021959a917e08eaf4eb46a8c4c8d1bec04e2c00,
67a348e256d908eb695d15ee0d869efef2bcf9f0fea646e788f967abbc0464dd,
edde0178045133eb03ef4d1ad8b978a56ee80ec4eab8830d6bc6c08003138841
..6657d3c449d9398cc4385d1c8a2bb19bcf61ff086e5a6c477a0302ce270d1abf,
e22ec3e4a2a4132237eb8a62bcc5ed864593cfde08e53b1632ecd3245761c808,
54c04f259f9e40ee086031d29960b12b6b6407e9de14985001c7265587941831,
9200b650a0c20b0ef73ccd7651ffc7af154e5e02879dc8666025c245aa547f01,
35f8dc0f744d1850513c46b6b4640716cbb4643da26cfe67f8c701486e0b4cae,
5faa89369589174f4202d6e53e8b4ef10a49b2ad8face60d7cb28bfc8f43bf0e,
017093ff8d22ba2f3852141365a1452fbb5ab8cf6f20cb04555e3163f8d88f13,

Vector 4

b56cc204f1b6c2323709012cb16c72f3021035ce935fbe69b600a88d842c7407,
dc2de7312c2850a9f6c103289c64fbd76e2ebd2fa8b5734708eb2c76c0fb2d99,
73616d706c65,
-,
-,
672e8c7a8e6d3eca67df38f11d50f3d7dbb26fa8e27565a5424e6f8ac4555dcc,
4d3e0524fc59374f1fdad8e471c695469b45ecf69c1de85c6c1230e888dd4cbe,

```
36127f8aee7c61048984f0a208bf6d334db9dacbeeeef9ff2d17117e81232832
..1462eb3ef602f5911d77ab11f815eb4154ba95c934e414198ef000a61b4de31a,
755610da34cc224fbe60ce5e42add2ea6b272ef466aef18c13497363116d1c03,
d26274e014ebfc19a9c1a951193858b972eae3360ed35635e89f1f9dbe432be5,
26202144ba4c4cb7ecde831c9e9662bec519493b29a098dd5803a8b4d261fc12,
b9fa51c75d278d95f2ccace9609b28ec137b244c8b7d1523b16ed07c8e24b8e4,
e42423127a2ca12d4f199287c8fa07784eacf9fc9b86a6bd56ee364cc352c009,
5371d6f9c76b560b4e42b9154a395bed60924d8de31284e926d06af382f5ad1b,
```

Vector 5

```
da36359bf1bfd1694d3ed359e7340bd02a6a5e54827d94db1384df29f5bdd302,
decb0151cbeb49f76f10419ab6a96242bdc87baac8a474e5161123de4304ac29,
42616e646572736e6174636820766563746f72,
-,
-,
4315192d2ce9e52ceb449a6b4da7f7e6636e53592c7f5e236763e21e9bac24c7,
9508104b820469687488d83f729288d9f70fc0523318beff44a47da10d490b3c,
4ee61f3c000544aa48c565e143e05c6501a623bdbf02a0a408b97433660b4907
..715f75890cc0e45cdd7116e3da15b15c3c637782e8e05d05c0d5895e5fe583d1,
fb0123dd6317dbd379afccded247f75b3c1c2e32b86eaa9d6c9d0eb5bef07919,
a91807f0ee57d2344a8942808bf35c65b5bd4fde16752a98f3e3dc67be8c103d,
dbc69ea4dd299deec2f29845e98a17700f07842cf2f6c5e9d88f388fa3a2831c,
311f94e886825c80a30fd44535be37218501bd072afcbc1298f8fba6c3e3c96d,
15f1562046078a7c0d152ef1b56bf3078c763089bf08790f10ff1cf3b9a5030b,
f962cb2598032cf5b0a21b4c253514d75c91ace15acde9a3f716e40f70f06804,
```

Vector 6

```
da36359bf1bfd1694d3ed359e7340bd02a6a5e54827d94db1384df29f5bdd302,
decb0151cbeb49f76f10419ab6a96242bdc87baac8a474e5161123de4304ac29,
42616e646572736e6174636820766563746f72,
-,
1f42,
4315192d2ce9e52ceb449a6b4da7f7e6636e53592c7f5e236763e21e9bac24c7,
9508104b820469687488d83f729288d9f70fc0523318beff44a47da10d490b3c,
4ee61f3c000544aa48c565e143e05c6501a623bdbf02a0a408b97433660b4907
..715f75890cc0e45cdd7116e3da15b15c3c637782e8e05d05c0d5895e5fe583d1,
0752c5b639dffedf9a66ac111a765d3e9c4cfac9c8b26cc5af6d524967afdf0a,
d03caebf8577c1d2ed30a09708683195f11883411dc170e3ea9f09a2cbf86bab,
8b16f0abb2873d6d56199280aeee9e02ce0274a9ca06a3194d6a72c25516ace8,
311f94e886825c80a30fd44535be37218501bd072afcbc1298f8fba6c3e3c96d,
9671cdae8b4cdeea640c24993ccf7e571fcfb3344d81d3cc6f36d03496777c1c,
624e25cd6eccec59b09f0893ef9eab877b55c757b9e9c81260255145bffd9a0d,
```

Vector 7

```
35b877a25c394512292b82bdf8468e98eaf03c79c7fc9d53546dadc5fb75b500,
b0e1f208f9d6e5b310b92014ea7ef3011e649dab038804759f3766e01029d623,
42616e646572736e6174636820766563746f72,
-,
1f42,
4315192d2ce9e52ceb449a6b4da7f7e6636e53592c7f5e236763e21e9bac24c7,
6d1dd583bea262323c7dc9e94e57a472e09874e435719010eeafae503c433f16,
09106f062ac07846f3f841f64765527b333575143483855d633f99ccc2e8e306
..e6239ff79a1272cff931e8d0ac6c390328486329118ad40a18b85184da1837ff,
462ae9ad651e5caf11247b989fecb5f2b1729479c33b9133388d14fa35dbbd0c,
91f1ca92eeaa0b604faf3e4811c12b44991ea33cf582a529a4bc4429a3b6cc5a,
b69946f270c46ccc59557bd40288a0a27607281da1892328fdb9da2dcb6c73cf,
5a02419120b814a5c81d67096aac728ee9bda5ddf9451cf554d871462a04831a,
bd8c0c1e5e04577c8836e45fb64131d1275309fe28e1d4334b230e3aa639da1a,
d93ccbd393ed88c8165b0a01aabe28c56a53b43e527e7927eeadff006dd22114,
```

A.3. Ring VRF Test Vectors

KZG SRS parameters are derived from Zcash BLS12-381 powers of tau ceremony.

The evaluations for the ZK domain items, specifically the evaluations of the last three items in the evaluation domain $\mathbb{D}$, are set to 0 rather than being randomly generated.

Schema:

```
sk (x): Secret key,
pk (Y): Public key,
in (alpha): Input octet-string,
ad: Additional data octet-string,
h (I): VRF input point,
gamma (O): VRF output point,
out (beta): VRF output octet string,
blinding: Blinding factor,
proof_pk_com (Y^-): Pedersen proof public key commitment,
proof_r: Pedersen proof 'R' component,
proof_ok: Pedersen proof 'O_k' component,
proof_s: Pedersen proof 's' component,
proof_sb: Pedersen proof 's_b' component,
ring_pks: Ring public keys,
ring_pks_com: Ring public keys commitment,
ring_proof: Ring proof
```

Vector 1

```
3d6406500d4009fdf2604546093665911e753f2213570a29521fd88bc30ede18,
```

a1b1da71cc4682e159b7da23050d8b6261eb11a3247c89b07ef56ccd002fd38b,
-,
-,
-,
c5eaf38334836d4b10e05d2c1021959a917e08eaf4eb46a8c4c8d1bec04e2c00,
e7aa5154103450f0a0525a36a441f827296ee489ef30ed8787cff8df1bef223f,
fdeb377a4ffd7f95ebe48e5b43a88d069ce62188e49493500315ad55ee04d744
..2b93c4c91d5475370e9380496f4bc0b838c2483bce4e133c6f18b0adbb9e4722,
01371ac62e04d1faaadbebaa686aaf122143e2cda23aacbaa4796d206779a501,
3b21abd58807bb6d93797001adaacd7113ec320dcf32d1226494e18a57931fc4,
8123054bfdb6918e0aa25c3337e6509eea262282fd26853bf7cd6db234583f5e,
ac57ce6a53a887fc59b6aa73d8ff0e718b49bd9407a627ae0e9b9e7c5d0d175b,
0d379b65fb1e6b2adcbf80618c08e31fd526f06c2defa159158f5de146104c0f,
e2ca83136143e0cac3f7ee863edd3879ed753b995b1ff8d58305d3b1f323630b,
7b32d917d5aa771d493c47b0e096886827cd056c82dbdba19e60baa8b2c60313
..d3b1bdb321123449c6e89d310bc6b7f654315eb471c84778353ce08b951ad471
..561fdb0dcfb8bd443718b942f82fe717238cbcf8d12b8d22861c8a09a984a3c5
..a1b1da71cc4682e159b7da23050d8b6261eb11a3247c89b07ef56ccd002fd38b
..4fd11f89c2a1aaefe856bb1c5d4a1fad73f4de5e41804ca2c17ba26d6e10050c
..86d06ee2c70da6cf2da2a828d8a9d8ef755ad6e580e838359a10accb086ae437
..ad6fdeda0dde0a57c51d3226b87e3795e6474393772da46101fd597fbd456c1b
..3f9dc0c4f67f207974123830c2d66988fb3fb44becbbba5a64143f376edc51d9,
afd34e92148ec643fbb578f0e14a1ca9369d3e96b821fcc811c745c320fe2264
..172545ca9b6b1d8a196734bc864e171484f45ba5b95d9be39f03214b59520af3
..137ea80e302730a5df8e4155003414f6dcf0523d15c6ef5089806e1e8e5782be
..92e630ae2b14e758ab0960e372172203f4c9a41777dadd529971d7ab9d23ab29
..fe0e9c85ec450505dde7f5ac038274cf,
98bc465cdf55ee0799bc25a80724d02bb2471cd7d065d9bd53a3a7e3416051f6
..e3686f7c6464c364b9f2b0f15750426a9107bd20fe94a01157764aab5f300d7e
..2fcba2178cb80851890a656d89550d0bebf60cca8c23575011d2f37cdc06dcdd
..93818c0c1c3bff5a793d026c604294d0bbd940ec5f1c652bb37dc47564d71dd1
..aa05aba41d1f0cb7f4442a88d9b533ba8e4788f711abdf7275be66d45d222dde
..988dedd0cb5b0d36b21ee64e5ef94e26017b674e387baf0f2d8bd04ac6faab05
..7510b4797248e0cb57e03db0199cd77373ee56adb7555928c391de794a07a613
..f7daac3fc77ff7e7574eaeb0e1a09743c4dae2b420ba59cf40eb0445e41ffb24
..49021976970c858153505b20ac237bfca469d8b998fc928e9db39a94e2df1740
..ae0bad6f5d8656806ba24a2f9b89f7a4a9caef4e3ff01fec5982af8731433463
..62a0eb9bb2f6375496ff9388639c7ffeb0bcee33769616e4878fc2315a3ac351
..8a9da3c4f072e0a0b583436a58524f036c3a1eeca023598682f1132485d3a570
..88b63acd86c6c72288568db71ff15b7677bfe7218acdebb144a2bf261eb4f659
..80f830e77f37c4f8d11eac9321f302a089698f3c0079c41979d278e8432405fc
..14d80aad028f79b0c4c626e4d4ac4e643692a9adfdc9ba2685a6c47eef0af5c8
..f5d776083895e3e01f1f944cd7547542b7e64b870b1423857f6362533f7cd2a0
..1d231ffed60fe26169c28b28ace1a307fdc8d4b29f0b44659402d3d455d719d8
..96f83b7ee927f0652ca883e4cfa85a2f4f7bc60dda1b068092923076893db5bd
..477fa2d26173314d7512760521d6ec9f,

Vector 2

```
8b9063872331dda4c3c282f7d813fb3c13e7339b7dc9635fdc764e32cc57cb15,
5ebfe047f421e1a3e1d9bbb163839812657bbb3e4ffe9856a725b2b405844cf3,
0a,
-,
-,
8c1d1425374f01d86b23bfeab770c60b58d2eeb9afc5900c8b8a918d09a6086b,
60f32f5ad3e9694b82ccc0a735edb2f940f757ab333cc5f7b0a41158b80f574f,
44f3728bc5ad550aeeb89f8db340b2fceffc946be3e2d8c5d99b47c1fce344b3
..c7fcee223a9b29a64fe4a86a9994784bc165bb0fba03ca0a493f75bee89a0946,
99ff52abf49d67c4303ac4a8a00984d04c06388f5f836ebd37031f0e76245815,
c1322e7a65b83996c25e37a84e36598333b0d417619242c0cb3d9d972edde848,
7a4363e0bf9cd18317287d681ab05704982b0088ce373f696dbdf3909a902b36,
fc8770c209212640742d53e2f40e5c30fffae574f90fdc670ff11a1127586c03,
93f7c9d73eec05e500b758f645a2967e62b2206e57eff5f9b99bfc71812e620d,
c864de36e0b428f6fb4ef470f94ec9601716cb26ad96f3359e4a1ec110794a0b,
7b32d917d5aa771d493c47b0e096886827cd056c82dbdba19e60baa8b2c60313
..d3b1bdb321123449c6e89d310bc6b7f654315eb471c84778353ce08b951ad471
..561fdb0dcfb8bd443718b942f82fe717238cbcf8d12b8d22861c8a09a984a3c5
..5ebfe047f421e1a3e1d9bbb163839812657bbb3e4ffe9856a725b2b405844cf3
..4fd11f89c2a1aaefe856bb1c5d4a1fad73f4de5e41804ca2c17ba26d6e10050c
..86d06ee2c70da6cf2da2a828d8a9d8ef755ad6e580e838359a10accb086ae437
..ad6fdeda0dde0a57c51d3226b87e3795e6474393772da46101fd597fbd456c1b
..3f9dc0c4f67f207974123830c2d66988fb3fb44becbbba5a64143f376edc51d9,
81ff2ae0324ba81dbc5f511fadd27d6fa23ff83d45a84ea96ed82f09ad73114a
..79349c978a86386c1a33c09f60c5362a99b73de3fe7f609d6f5f35736a6eb82c
..739943ad4a3d1fe3f1b589d5b173ad3351786b08e07a1369f82fee25b4a16001
..92e630ae2b14e758ab0960e372172203f4c9a41777dadd529971d7ab9d23ab29
..fe0e9c85ec450505dde7f5ac038274cf,
a57818b60d8fc54695a66b49a627b158a2f4141c696f0ac41b16831021e0ce56
..04aaa76fab504c106e4a50621adcbeeb9107bd20fe94a01157764aab5f300d7e
..2fcba2178cb80851890a656d89550d0bebf60cca8c23575011d2f37cdc06dcdd
..a5d0e0b9c8dfabd2a88713ea7448a6afed58c035994f52a06a37045b7fe9bc88
..00939475c3beae30ee28aabdb0932bacb7967c476a0b2aaa9bc536fd18b487a6
..5135ae128d4c6fe14dc98160c841a9183ac4a31adf99ad98c4f18368eed0733b
..7b5126d767299e72a086d9cd9fda84ef8392425404173b80a430a9b320c6cf46
..f203e0b7214333ab49b43bd68bef7db51fd7d55f3332122aa7e65dd990eb5c36
..fdef18cfe2ef8624e1372d4ae51fb115572e4a67ada192739a8eddfee2f88c53
..e9072a320d73c78176f8572b8021f5aa2bfe82834b546cd93295bf05d7b6b81a
..c56d1a3c8cdcccb575ef8e6865d0b45a2e684e3d03cdfe941d823450076a7229
..fa17e1d1a92e8a4e12672837f603e0b782235fee0b4f3f2673972730c14e224f
..0b6cd6e8a2e24358539a2cc242cf792d9b85cd784a6496192404c6ecc68ee370
..b75f373ee9d9ba48a2de51d3b3f0a923a9385444eb6396f2ec220cefe3113bf2
..08f2697fb1625da3c8d12e7ab8d405c8c05cc70074a7e2b76d73e9fc2e05e95b
..303920abf93139baaadba3911e3d2d63ae5335a8be8fb028df0052aded98f2d8
```

```
..1234fe608836b4896b2e080b9b9fa306be342e1aeb95368beb3099a97f2dd0c1
..b10e54e38efb04c2b8977da7da8dfa801d6997de31337ea2c4ca2ddc77ad4356
..6a614cc1742a24285ff9da590746aa6d,
```

Vector 3

```
6db187202f69e627e432296ae1d0f166ae6ac3c1222585b6ceae80ea07670b14,
9d97151298a5339866ddd3539d16696e19e6b68ac731562c807fe63a1ca49506,
-,
-,
0b8c,
c5eaf38334836d4b10e05d2c1021959a917e08eaf4eb46a8c4c8d1bec04e2c00,
67a348e256d908eb695d15ee0d869efef2bcf9f0fea646e788f967abbc0464dd,
edde0178045133eb03ef4d1ad8b978a56ee80ec4eab8830d6bc6c08003138841
..6657d3c449d9398cc4385d1c8a2bb19bcf61ff086e5a6c477a0302ce270d1abf,
e22ec3e4a2a4132237eb8a62bcc5ed864593cfde08e53b1632ecd3245761c808,
54c04f259f9e40ee086031d29960b12b6b6407e9de14985001c7265587941831,
9200b650a0c20b0ef73ccd7651ffc7af154e5e02879dc8666025c245aa547f01,
35f8dc0f744d1850513c46b6b4640716cbb4643da26cfe67f8c701486e0b4cae,
5faa89369589174f4202d6e53e8b4ef10a49b2ad8face60d7cb28bfc8f43bf0e,
017093ff8d22ba2f3852141365a1452fbb5ab8cf6f20cb04555e3163f8d88f13,
7b32d917d5aa771d493c47b0e096886827cd056c82dbdba19e60baa8b2c60313
..d3b1bdb321123449c6e89d310bc6b7f654315eb471c84778353ce08b951ad471
..561fdb0dcfb8bd443718b942f82fe717238cbcf8d12b8d22861c8a09a984a3c5
..9d97151298a5339866ddd3539d16696e19e6b68ac731562c807fe63a1ca49506
..4fd11f89c2a1aaefe856bb1c5d4a1fad73f4de5e41804ca2c17ba26d6e10050c
..86d06ee2c70da6cf2da2a828d8a9d8ef755ad6e580e838359a10accb086ae437
..ad6fdeda0dde0a57c51d3226b87e3795e6474393772da46101fd597fbd456c1b
..3f9dc0c4f67f207974123830c2d66988fb3fb44becbbba5a64143f376edc51d9,
b2c08aa40031c9f2a66353b9f3b4e30440704e46f277167fc04fdb7812953cc6
..bcbca4beb8288a2585a26d786c313c8699b0dd6b2e008ea7aa3b00c273da21ba
..02bb6d51e39c8a2caf826bc2860d5b8e07b829647b1164c53ab307b9ca0c2bf5
..92e630ae2b14e758ab0960e372172203f4c9a41777dadd529971d7ab9d23ab29
..fe0e9c85ec450505dde7f5ac038274cf,
a28c6420603f4cd2efd457092ef74585f78eeae389e2ffabf58b9f9dd14ec4db
..9ffe14be02b7376f6ae7959e11ce1e559107bd20fe94a01157764aab5f300d7e
..2fcba2178cb80851890a656d89550d0bebf60cca8c23575011d2f37cdc06dcdd
..a06a150e462634cd8e2a43f4a377bf503f37896a5ff3994e5dfe0d76d7080cab
..3c616ed081e9f13ba653904fce67a0e08476090b55153972ea2b7609d437cdee
..cce2475bfd26aede213d5878a93ace5f99620bdabe5970d32feba4b425e600a8
..f53b81887987ba0c9401bb8fd21d5219fc054674e8f6eea7824b5643e31d0d55
..da7c33c6610cb183ed9fedca5321be353303da40b63bfc00719c8d253009520f
..67a8f11107b915e404a4b3066632c858314a3c2de105cf69d4ca584ae02c3e4a
..60829ae05278cfcd8854e1eff8af8a3bda9d92797d404578011368e726052e22
..6ad658bafb6a5e5a51d1dd9ae7dc88598420f2d17146abd268471191dff53b6a
..d7267bd8bd716c004d0984b4bdb4c5d5295abe978f53cd716a43777658fe9140
```

```
..7c4e6f164ab9eb72050b7ab4845068ec48c4ef4933da4eb9f7bda8cbaee06d45
..90d328064880d590e9cf3933d96a6c15d2f2cb1bb2d6247d522089587646f8de
..386429e448d962155b5ebfcb4962e1ea3186608ce8a0928ae0eb90ebc38fac9e
..0a78cf12d236efb73ecb943636ec911a8aed1c634ab079758eb1b68aaf765857
..27dc23bc29c5e05797d7e1c0fc30a5eb00b0a323d8fe728072821dbf1c903394
..932472e99a82bb70e69505a27ed9ace5862c1e247530904bef58d4ae28f9d8d8
..2d61a5b7f8c92efeaaaccbe1ba0c9c97,
```

Vector 4

```
b56cc204f1b6c2323709012cb16c72f3021035ce935fbe69b600a88d842c7407,
dc2de7312c2850a9f6c103289c64fbd76e2ebd2fa8b5734708eb2c76c0fb2d99,
73616d706c65,
-,
-,
672e8c7a8e6d3eca67df38f11d50f3d7dbb26fa8e27565a5424e6f8ac4555dcc,
4d3e0524fc59374f1fdad8e471c695469b45ecf69c1de85c6c1230e888dd4cbe,
36127f8aee7c61048984f0a208bf6d334db9dacbeeeef9ff2d17117e81232832
..1462eb3ef602f5911d77ab11f815eb4154ba95c934e414198ef000a61b4de31a,
755610da34cc224fbe60ce5e42add2ea6b272ef466aef18c13497363116d1c03,
d26274e014ebfc19a9c1a951193858b972eae3360ed35635e89f1f9dbe432be5,
26202144ba4c4cb7ecde831c9e9662bec519493b29a098dd5803a8b4d261fc12,
b9fa51c75d278d95f2ccace9609b28ec137b244c8b7d1523b16ed07c8e24b8e4,
e42423127a2ca12d4f199287c8fa07784eacf9fc9b86a6bd56ee364cc352c009,
5371d6f9c76b560b4e42b9154a395bed60924d8de31284e926d06af382f5ad1b,
7b32d917d5aa771d493c47b0e096886827cd056c82dbdba19e60baa8b2c60313
..d3b1bdb321123449c6e89d310bc6b7f654315eb471c84778353ce08b951ad471
..561fdb0dcfb8bd443718b942f82fe717238cbcf8d12b8d22861c8a09a984a3c5
..dc2de7312c2850a9f6c103289c64fbd76e2ebd2fa8b5734708eb2c76c0fb2d99
..4fd11f89c2a1aaefe856bb1c5d4a1fad73f4de5e41804ca2c17ba26d6e10050c
..86d06ee2c70da6cf2da2a828d8a9d8ef755ad6e580e838359a10accb086ae437
..ad6fdeda0dde0a57c51d3226b87e3795e6474393772da46101fd597fbd456c1b
..3f9dc0c4f67f207974123830c2d66988fb3fb44becbbba5a64143f376edc51d9,
89804d665064e57f7de11fc3eeb50fcc7432972e344d14e9fe0f1c4a62ecb1a8
..0bf0927ac19553787590c3a834feec1083c8f1104bb55779216dd5051aefe928
..73b36f331309eaa429bb1bc14f8b31270964bdc061ae6c856e05b19117508112
..92e630ae2b14e758ab0960e372172203f4c9a41777dadd529971d7ab9d23ab29
..fe0e9c85ec450505dde7f5ac038274cf,
84c37dff677bda19d7ce202500196edeabb794b0e0970b52a76061d9fc9c396f
..5d6671db8da091886f4f894775b49a549107bd20fe94a01157764aab5f300d7e
..2fcba2178cb80851890a656d89550d0bebf60cca8c23575011d2f37cdc06dcdd
..a8c53db306c4d0d30a509b2d3b48706279629cd604a3e220fdcd49c44876b6a0
..455419b418f22b18969cec9423e540dd89dd715839826e4aede1002bb651de89
..68c528cf5d2eb41f17e10494c89f3b475aedfbfd5b5de126f7f22c2fe83e2da0
..0dc1af96ea799fe2dcbd9f0985bc69dc6ce5d23c8c51075af7aa2062927f4d73
..b76d01e3ca5b67377c420afac1850d6e053114594e7680222b06c6d1ce239d13
```

..e69c933daa0b6671896b92456a4856b9bf92dcca0329bd1743ee6dbf0912b347
..7b1b03294a9b1553d7c79e6bc82448ee76f9dc8e9c869cca60346f0cc2700d20
..0bec0aa5f9d4f1b91acc3d0e21456779b7a0abcb1a6e52e8126410131c63756c
..2f161d49f441fdfaf719feeeedcd7b8c9f327eac0c279b42e251b71c104b280d
..8d53a5dfd6be57c646fff62151f43372f768375472ea3bf5f23a23c040b4c26f
..9460ee2b0f8b2c3946d8ce0909b80e7ee5608c1735ab06cc5bed2425fd823ce6
..8938a4b576f245ad0fcab22a6609a9ebfc5c3716d67148dd570f64f2da5cde1f
..801c0d421eb4f978026ff58aa8cb03118bd50badc10a9ac76c27579d7b074bff
..cdfe592547df5db7b7cf36de8b7ac43ea2329196d2daef4faf01cafaf016f927
..a376a1bcb5a8961c8135b8df70b98d34b6d32eade381a55595a3ba9a324d596b
..5ea2a1f01565430d7ba3965ad07450c0,

Vector 5

da36359bf1bfd1694d3ed359e7340bd02a6a5e54827d94db1384df29f5bdd302,
decb0151cbeb49f76f10419ab6a96242bdc87baac8a474e5161123de4304ac29,
42616e646572736e6174636820766563746f72,
-,
-,
4315192d2ce9e52ceb449a6b4da7f7e6636e53592c7f5e236763e21e9bac24c7,
9508104b820469687488d83f729288d9f70fc0523318beff44a47da10d490b3c,
4ee61f3c000544aa48c565e143e05c6501a623bdbf02a0a408b97433660b4907
..715f75890cc0e45cdd7116e3da15b15c3c637782e8e05d05c0d5895e5fe583d1,
fb0123dd6317dbd379afccded247f75b3c1c2e32b86eaa9d6c9d0eb5bef07919,
a91807f0ee57d2344a8942808bf35c65b5bd4fde16752a98f3e3dc67be8c103d,
dbc69ea4dd299deec2f29845e98a17700f07842cf2f6c5e9d88f388fa3a2831c,
311f94e886825c80a30fd44535be37218501bd072afcbc1298f8fba6c3e3c96d,
15f1562046078a7c0d152ef1b56bf3078c763089bf08790f10ff1cf3b9a5030b,
f962cb2598032cf5b0a21b4c253514d75c91ace15acde9a3f716e40f70f06804,
7b32d917d5aa771d493c47b0e096886827cd056c82dbdba19e60baa8b2c60313
..d3b1bdb321123449c6e89d310bc6b7f654315eb471c84778353ce08b951ad471
..561fdb0dcfb8bd443718b942f82fe717238cbcf8d12b8d22861c8a09a984a3c5
..decb0151cbeb49f76f10419ab6a96242bdc87baac8a474e5161123de4304ac29
..4fd11f89c2a1aaefe856bb1c5d4a1fad73f4de5e41804ca2c17ba26d6e10050c
..86d06ee2c70da6cf2da2a828d8a9d8ef755ad6e580e838359a10accb086ae437
..ad6fdeda0dde0a57c51d3226b87e3795e6474393772da46101fd597fbd456c1b
..3f9dc0c4f67f207974123830c2d66988fb3fb44becbbba5a64143f376edc51d9,
8a73624024659145d99046f7c25332c0427a1a419b90c9c31f11dcb43bfa5dea
..85fe0551e6e344c777470a6a75665195980c655ed0b3fe459512a901981313ce
..23003968a3487ea5b5de2f9f4372727ac137ab04fec24c03a4ebd77055851650
..92e630ae2b14e758ab0960e372172203f4c9a41777dadd529971d7ab9d23ab29
..fe0e9c85ec450505dde7f5ac038274cf,
a7185a7a63812926137b53a4776569fe2323e84689e9e2523e03d3c61beb0427
..7bde2c4a2a5e6acecbbfe1c09f16f9899107bd20fe94a01157764aab5f300d7e
..2fcba2178cb80851890a656d89550d0bebf60cca8c23575011d2f37cdc06dcdd
..b2c7a1bdeee0b96e5636a6bc0e91bf3a2309e9a396c1a079f1be004fb8f01c58

..a559c750c46b100666e10f841924c3aaafcd90d79db59ea96d9f519cd23e85e1
..498b18f2a1b5fde389db657178e77d55c946cfcc485343d2e14ed37fb5d39e7b
..bbccea9103664414fe5eb0b4224721d790674a15b386d31f1d6b4ed8ccc74536
..305372529273c12476382842eb14722352276ab05e1cdab867fe43b6da8d1149
..41f5dd46bda9b29681e6210fbfba89c09e123ae6aa788845d4f87dbc42228b66
..39a51dc61bc79ff3e53da217679648be2d6d4e4a0277d05afa98dd7378672337
..3c3d8d4932f6258f174793f9597b9ef888e0e8bcb37ce0f79669411163c51570
..e6ffd7afb89a3d8f43e1e9fdfce7c583c9a6bb32414c6787d83ad3d46812a43e
..bf758c58417d169b0a8e7f08befd8beea50699e786f54037f8859c362a7b4a17
..96c40e76ce0c05c0920f0d092c78123470cfc80e13198bca67c003bf554642fb
..1cfecc2cc5c6a418a628e2d16152c8dadd9e069e3c3ee9f6715766760d4cdfab
..d545348c67f22e576d6ab90184843e49989ff102693018ca1256aa1d20d73416
..b6a2cf93aeaf2a985b109aef311e94f1b2b9951a341496a3584d386fe998b9ea
..b014b5d452cf9489672089388ff5927e37593773ea18e316fd2bd0a8b0566f09
..f7d36173103dc41ceb82e4dc9fcfc5f9,

Vector 6

da36359bf1bfd1694d3ed359e7340bd02a6a5e54827d94db1384df29f5bdd302,
decb0151cbeb49f76f10419ab6a96242bdc87baac8a474e5161123de4304ac29,
42616e646572736e6174636820766563746f72,
-,
1f42,
4315192d2ce9e52ceb449a6b4da7f7e6636e53592c7f5e236763e21e9bac24c7,
9508104b820469687488d83f729288d9f70fc0523318beff44a47da10d490b3c,
4ee61f3c000544aa48c565e143e05c6501a623bdbf02a0a408b97433660b4907
..715f75890cc0e45cdd7116e3da15b15c3c637782e8e05d05c0d5895e5fe583d1,
0752c5b639dffedf9a66ac111a765d3e9c4cfac9c8b26cc5af6d524967afdf0a,
d03caebf8577c1d2ed30a09708683195f11883411dc170e3ea9f09a2cbf86bab,
8b16f0abb2873d6d56199280aeee9e02ce0274a9ca06a3194d6a72c25516ace8,
311f94e886825c80a30fd44535be37218501bd072afcbc1298f8fba6c3e3c96d,
9671cdae8b4cdeea640c24993ccf7e571fcfb3344d81d3cc6f36d03496777c1c,
624e25cd6eccec59b09f0893ef9eab877b55c757b9e9c81260255145bffd9a0d,
7b32d917d5aa771d493c47b0e096886827cd056c82dbdba19e60baa8b2c60313
..d3b1bdb321123449c6e89d310bc6b7f654315eb471c84778353ce08b951ad471
..561fdb0dcfb8bd443718b942f82fe717238cbcf8d12b8d22861c8a09a984a3c5
..decb0151cbeb49f76f10419ab6a96242bdc87baac8a474e5161123de4304ac29
..4fd11f89c2a1aaefe856bb1c5d4a1fad73f4de5e41804ca2c17ba26d6e10050c
..86d06ee2c70da6cf2da2a828d8a9d8ef755ad6e580e838359a10accb086ae437
..ad6fdeda0dde0a57c51d3226b87e3795e6474393772da46101fd597fbd456c1b
..3f9dc0c4f67f207974123830c2d66988fb3fb44becbbba5a64143f376edc51d9,
8a73624024659145d99046f7c25332c0427a1a419b90c9c31f11dcb43bfa5dea
..85fe0551e6e344c777470a6a75665195980c655ed0b3fe459512a901981313ce
..23003968a3487ea5b5de2f9f4372727ac137ab04fec24c03a4ebd77055851650
..92e630ae2b14e758ab0960e372172203f4c9a41777dadd529971d7ab9d23ab29
..fe0e9c85ec450505dde7f5ac038274cf,

a8455660f642a887ca7bced683e7c5315c6ebff1d7d047ca43f5b5c7b34c244a
..3902f6ca62346b638ed58e4aa5b2c1c29107bd20fe94a01157764aab5f300d7e
..2fcba2178cb80851890a656d89550d0bebf60cca8c23575011d2f37cdc06dcdd
..85a6e99a7824dc5f71bbe12cd251236e62139b0fd1f2f17e979f098520fb37c4
..cee0af5e1fa9f328d351bbe59044eb3f8156b5f8f7ee79d502b573f752c9d913
..fdb27c803162ca2b0711a1544679c8c40247a02d5c555d5c1f70cc6c98e82adb
..77224cedf52fe9093b6dea787a5f04c67344bae1bc8a73ec67839de08202a95f
..3156b0923bfb4de17fc2fea219af465b11f1285c32008b9ca59a88d7b085e23f
..95f99f59c457e2ecdffb9dc7fc3c8c169bc1195e70f5e7c49c56329297d0c133
..9411db06e43c66b2fd1cbe3807d1225f8eb9ca6a5b19653271a51a737224204d
..55e9e06cb7eb1748f7770d9cdcda4b0030d84c0213b76ad5fbc9c36da4c8e21e
..daa6e8dc51fa80dfb938ec531e5ef248867c9d02f3604cbd8172a80a241feb49
..61ae6a954dc81e266cff5a651b3127700848ae83e88246034d18a55f2e9eba26
..b2cb13d93cb704fff63665b87bd0ccd595c8561ae10a4484af3a070a20537df6
..79737410b648dd6c187cdc0277690b882497d53af3a8fcab1243a0bab5a49d69
..22a4e3ae917b19959a085708d26cd90898763d698a75332d1aa817534af67420
..f58c4126d7710ff7075d9a78bf4eaaf9ec30a9d4562f6db0f14ae82cf384179b
..8bd41390d2340377998f0ab8fb1547ebd4b45b36110a582db3e0d19ecb8cac6a
..8d35ba072f8176100cdde64c53a14b53,

Vector 7

35b877a25c394512292b82bdf8468e98eaf03c79c7fc9d53546dadc5fb75b500,
b0e1f208f9d6e5b310b92014ea7ef3011e649dab038804759f3766e01029d623,
42616e646572736e6174636820766563746f72,
-,
1f42,
4315192d2ce9e52ceb449a6b4da7f7e6636e53592c7f5e236763e21e9bac24c7,
6d1dd583bea262323c7dc9e94e57a472e09874e435719010eeafae503c433f16,
09106f062ac07846f3f841f64765527b333575143483855d633f99ccc2e8e306
..e6239ff79a1272cff931e8d0ac6c390328486329118ad40a18b85184da1837ff,
462ae9ad651e5caf11247b989fecb5f2b1729479c33b9133388d14fa35dbbd0c,
91f1ca92eeaa0b604faf3e4811c12b44991ea33cf582a529a4bc4429a3b6cc5a,
b69946f270c46ccc59557bd40288a0a27607281da1892328fdb9da2dcb6c73cf,
5a02419120b814a5c81d67096aac728ee9bda5ddf9451cf554d871462a04831a,
bd8c0c1e5e04577c8836e45fb64131d1275309fe28e1d4334b230e3aa639da1a,
d93ccbd393ed88c8165b0a01aabe28c56a53b43e527e7927eeadff006dd22114,
7b32d917d5aa771d493c47b0e096886827cd056c82dbdba19e60baa8b2c60313
..d3b1bdb321123449c6e89d310bc6b7f654315eb471c84778353ce08b951ad471
..561fdb0dcfb8bd443718b942f82fe717238cbcf8d12b8d22861c8a09a984a3c5
..b0e1f208f9d6e5b310b92014ea7ef3011e649dab038804759f3766e01029d623
..4fd11f89c2a1aaefe856bb1c5d4a1fad73f4de5e41804ca2c17ba26d6e10050c
..86d06ee2c70da6cf2da2a828d8a9d8ef755ad6e580e838359a10accb086ae437
..ad6fdeda0dde0a57c51d3226b87e3795e6474393772da46101fd597fbd456c1b
..3f9dc0c4f67f207974123830c2d66988fb3fb44becbbba5a64143f376edc51d9,
a359e70e307799b111ad89b162b4260fb4a96ebf4232e8b02d396033498d4216

```
..305ed9cff3584a4c68f03ab3df87243a80bfc965633efc23c82ca064afe105ba
..acccbf23e47b543d16c3c4466a83242a77acc16f79b8710051b5e97c85319cf3
..92e630ae2b14e758ab0960e372172203f4c9a41777dadd529971d7ab9d23ab29
..fe0e9c85ec450505dde7f5ac038274cf,
88a7fc8a8ae7d295bdd26553b06d298c7d7fdb3f08746aba8e3312d78254a201
..3d4cd3bea7b62156b5a5b0a42e7e45179107bd20fe94a01157764aab5f300d7e
..2fcba2178cb80851890a656d89550d0bebf60cca8c23575011d2f37cdc06dcdd
..807677ad0f02386ef4e6eb99d230aca4c9a73ee3f68397a65c98d6377bf32e1c
..e36ae082eafb81393786ab07f7a461798e0d87a09cf3d6ea801fc22d65c907dc
..b30ffb60128a307105cdd5db58f93967ec1c0497eba4db9dac64eb2dfe43baeb
..30cbe5d2184c998b9070dcda0c8170286d793d583982adb77f01f2ddd295a161
..61399bcc4b0505757b49074173aca578544d676a44d37fc01321feec35e97437
..6e2080c28a9bfd963566febe2aac12fe51326001b67abbbdc2d67a24c7605b56
..b4c12eb37e41fbace1e45bf37750057ca1bea272955a82bd2b9930d1a4c2f03d
..7dc19572954c84f2175706be08a9e7b12445f50bb4bcd6ed77c3d8f3356cc034
..6bc0a6359ca9384239bf09b160a5845710e688c4dbc7cb7e27494f0746818221
..f6b6e7af92a58c68f29415fdea5f75ab3e45b1138a7ecde2ff6eac44772f100f
..a4e6f4b97c4a401e919dbfd1386253a1dde71a3d6bbf24de9a01e1adca917d84
..e3f10888680e2b736f51a1336d7f6099000777bd3a6de5636654012b5873d109
..4994887e59f7a6565999b922cafac10782ba51d1290e344ddd3ee84a324ba241
..b8c9a32c6fb85bb7164878f2de12c9cece5ef7b92860a55fb25ccb65839a66b1
..a73d4975f06823422cbe4aac82eef13e88eb6f6fd363bb19436f34b40825438d
..c55d3812bb0485d4ae4f9cc67af7074d,
```

References

1.

Internet Engineering Task Force *Verifiable Random Functions*; RFC Editor, 2023;

2.

Burdges, J.; Ciobotaru, O.; Alper, H.K.; Stewart, A.; Vasilyev, S. Ring Verifiable Random Functions and Zero-Knowledge Continuations 2023.

3.

Vasilyev, S.; Galassi, D. Ring Proof Technical Specification 2024.

4.

Masson, S.; Sanso, A.; Zhang, Z. Bandersnatch: A Fast Elliptic Curve Built over the BLS12-381 Scalar Field 2021.

5.

Internet Engineering Task Force *Hashing to Elliptic Curves*; RFC Editor, 2023;

6.

Internet Engineering Task Force *US Secure Hash Algorithms*; RFC Editor, 2011;