CRYPTONOTE STANDARD 002 Nicolas van Saberhagen
Obsoletes: CNS001 Johannes Meier
Category: Main Track Antonio M. Juarez
Max Jameson
Seigen
CryptoNote
May 2012

# CryptoNote Signatures

Abstract

This document is part of the CryptoNote Standards describing a peer-to-peer anonymous payment system. It defines the exact method of zero-knowledge proof to establish the ownership of an asset. The standard CryptoNote approach is the one-time ring signature scheme: it allows a user to sign a message on behalf of the group while keeping his identity indistinguishable from others. Only one signature is allowed under the same key ("one-time" property), as any other signatures will be unambiguously linked with the first one, which prevents double-spending.



Table of Contents







## 1. Introduction

A distinguishing feature of cryptocurrencies is that they establish the money ownership and the authenticity of money transfers by cryptographic means, specifically through the use of digital signatures. A digital signature certifies that a particular transaction is authorized by the owner of a particular key. However, with regular digital signatures, the key which was used to authenticate each transaction can be precisely identified. While the keys are not directly linked to their owners' identities, they can be used to link transactions. This means that it is possible to start from a transaction whose sender or recipient is known, and trace the funds either forward or backward to deduce how they are spent, or where they were obtained.

CryptoNote uses advanced cryptography to obscure some of this information. An ideal scheme would leave attackers completely oblivious to a transaction's funding sources, but the scheme used in CryptoNote is more limited. Firstly, a transaction input can only obtain its funds from an output of the same amount. Secondly, each input lists specific outputs, and it is known that the output spent by the input is in the list. However, no information is disclosed beyond this fact; a signature does not reveal any information about how likely it is that an input spends a particular output in the list.

## 2. Definitions

digital signature: a cryptographic method of showing that a particular message was authorized by a particular peer

public key: a datum used to identify a peer for the purpose of digital signature verification

ring signature: a class of schemes that allow a user to sign a message on behalf of a group, making his identity indistinguishable from the other members of the group

secret key: data known to a peer only, which enables him to create digital signatures under his identity

## 3. One-Time Ring Signatures

Like regular digital signatures, keys for CryptoNote signatures come in pairs. Public keys are included in transaction outputs and are used to check the signatures, while the corresponding secret keys are





used to create them. However, CryptoNote utilizes ring signatures instead of regular, and these signatures may need multiple keys for verification. A single secret key is needed to create a signature, and this key must correspond to one of the public keys the signature is verified against. Moreover, it is computationally infeasible to recover the secret key used to create the signature. This means that the sender of a transaction can link his inputs to multiple outputs, making the task of tracking the funds more complex.

However, if CryptoNote used standard ring signatures, it would not be known which outputs are spent and which are not. Therefore, it would be possible to spend the same output multiple times. This would lead to a double-spend problem, nullifying the scarcity of the currency. To solve this issue, CryptoNote signatures were made one-time, which means that it is possible to detect when multiple signatures are made with the same key without revealing the key. This means that the double-spend protection does not break the privacy properties.

The properties of CryptoNote signatures can be summarized as follows:

1. Usability: any person with a secret key can create a signature on any message under that key and any other public keys. The list

of keys under which the signature is created, including the public key which corresponds to the secret key used, is called a ring.

2. Security: it is not possible to create a signature without possessing a secret key corresponding to one of the public keys in the ring.

3. Anonymity: the signature does not convey any information beyond being created by one of the keys in the ring. Signatures created using different keys are indistinguishable, with a small exception:

4. Linkability: it is possible to tell if two signatures were made with the same secret key. No information is revealed beyond that, therefore it could be any of the keys present in both rings.

## 4. Theoretical Description

The construction of the One-Time Ring Signatures used in CryptoNote is a simplified version of Traceable Ring Signatures by Fujisaki and Suzuki [TRS]. It is based on the framework of Camenisch and Stadler [DLOG]. It uses a group with an element G of prime order l. A secret key is an integer modulo l, and the corresponding public key is the value A=a*G.





A signature includes a key image, which is a value that corresponds to a key that can't be derived without the knowledge of the respective secret key. The key image included in a signature must correspond to the key used to create it. This facilitates the detection of signatures made with the same key: all such signatures will include the same key image. The key image corresponding to public key A is a*H(A), where a is the corresponding secret key and H is a hash function which maps public keys to group elements.

In addition to the key image, each signature includes a zero-knowledge proof of knowledge of the secret key corresponding to one of the public keys used to validate the signature, such that the key image in the signature corresponds to that key. If the signature is validated with keys A[1], A[2], ..., A[n] and includes key image I, then the proof is as follows:

    ZKPoK[(i, a) | A[i]=a*G and I=a*H(A[i])]

The proof is made non-interactive by means of Fiat-Shamir transform [FS]. The hash of the message is added to the commitment, which binds the signature to the message.

## 5. Data Types and Accessory Functions

The CryptoNote signature scheme uses Curve25519 (see [CURVE]) as the underlying group. Group elements are encoded in the same way as in Ed25519 (see [ED25519]). Integers modulo l (henceforth named scalars) are represented in the 32-byte little-endian form. To prevent malleability, the integers encoded must lie between 0 and l-1.

The signature consists of the key image (a single group element) and 2*n scalars, where n is the number of keys used. Scalars are grouped into n pairs, the scalars in the i-th pair are the challenge and the response values c[i] and r[i] for the part of the proof concerning

A[i]. The commitment consists of the hash of the message followed by 2*n group elements grouped into n pairs. The elements in the i-th pair are respectively c[i]*A[i]+r[i]*G and c[i]*I+r[i]*H(A[i]).

The hash function used is the same Keccak function that is used throughout CryptoNote. When the value of the hash function is interpreted as a scalar, it is converted into a little-endian integer and taken modulo l. When it is interpreted as a group element, it is passed to a special function that is guaranteed to return a valid group element.





## 6. Signature Generation

In the following two procedures || denotes concatenation.

To generate a signature, the following procedure is used:

```
Procedure generate_signature(M, A[1], A[2], ..., A[n], i, a[i]):
   I <- a[i]*H(A[i])
   c[j], r[j] [j=1..n, j!=i] <- random
   k <- random
   For j <- 1..n, j!=i
      X[j] <- c[j]*A[j]+r[j]*G
      Y[j] <- c[j]*I+r[j]*H(A[j])
   End For
   X[i] <- k*G
   Y[i] <- k*H(A[i])
   c[i] <- H(H(M) || X[1] || Y[1] || X[2] || Y[2] || ... || X[n] ||
    Y[n])-Sum[j=1..n, j!=i](c[j])
   r[i] <- k-a[i]*c[i]
   Return (I, c[1] || r[1] || c[2] || r[2] || ... || c[n] || r[n])
End Procedure
```

## 7. Signature Verification

Signatures are verified using the following procedure:

```
Procedure verify_signature(M, A[1], A[2], ..., A[n], I, c[1], r[1],
 c[2], r[2], ..., c[n], r[n]):
   For i <- 1..n
      X[i] <- c[i]*A[i]+r[i]*G
      Y[i] <- c[i]*I+r[i]*H(A[i])
   End For
   If H(H(M) || X[1] || Y[1] || X[2] || Y[2] || ... || X[n] || Y[n])
    = Sum[i=1..n](c[i])
      Return "Correct"
   Else
      Return "Incorrect"
   End If
End Procedure
```

## 8. Security Considerations

It is of utmost importance that the random numbers used during the signatures generation are produced by a cryptographically secure random number generator. The distribution of the numbers must be indistinguishable from the uniform distribution. Insecure generation of the numbers c[j] and r[j] (j!=i) can be used to compromise anonymity, while insecure generation of k can compromise the secret key a[i].

Some obvious choices of hash-to-group-element function admit certain attacks that can be used to break user anonymity. The function used in CryptoNote is not susceptible to such attacks.

## 9. Differences from CNS001

These are the major differences between this document and [CNS001]:

- a new section "Theoretical Description" has been added,

- the signature algorithm has been updated to obviate the necessity for the second basepoint G2,

- a new entity "key image" has been defined instead of a group element Q,

- specific elliptic curve parameters (namely, Curve25519) have been chosen.

## 10. References

[CNS001] "CryptoNote Signatures", CryptoNote Standard 001, December 2011.

[CURVE] Bernstein, D. J., "Curve25519: new Diffie-Hellman speed records", 2006, http://cr.yp.to/ecdh/curve25519-20060209.pdf.

[DLOG] Camenisch, J., and M. Stadler, "Proof Systems for General Statements about Discrete Logarithms", 1997.

[ED25519] Bernstein, D. J., Duif, N., Lange, T., Schwabe, P., and B.-Y. Yang, "High-speed high-security signatures", 2011, http://ed25519.cr.yp.to/ed25519-20110926.pdf.

[FS] Fiat, A., and A. Shamir, "How To Prove Yourself: Practical Solutions to Identification and Signature Problems", 1987.

[TRS] Fujisaki, E., and K. Suzuki, "Traceable Ring Signature", 2007.