

them if necessary that causes the main drawbacks. Unfortunately, it is hard to predict when the constants may need to be changed and replacing them may lead to terrible consequences.

A good example of a hardcoded limit change leading to disastrous consequences is the block size limit set to 250kb<sup>1</sup>. This limit was sufficient to hold about 10000 standard transactions. In early 2013, this limit had almost been reached and an agreement was reached to increase the limit. The change was implemented in wallet version 0.8 and ended with a 24-blocks chain split and a successful double-spend attack [9]. While the bug was not in the Bitcoin protocol, but rather in the database engine it could have been easily caught by a simple stress test if there was no artificially introduced block size limit.

Constants also act as a form of centralization point. Despite the peer-to-peer nature of Bitcoin, an overwhelming majority of nodes use the official reference client [10] developed by a small group of people. This group makes the decision to implement changes to the protocol and most people accept these changes irrespective of their “correctness”. Some decisions caused heated discussions and even calls for boycott [11], which indicates that the community and the developers may disagree on some important points. It therefore seems logical to have a protocol with user-configurable and self-adjusting variables as a possible way to avoid these problems.

## 2.5 Bulky scripts

The scripting system in Bitcoin is a heavy and complex feature. It *potentially* allows one to create sophisticated transactions [12], but some of its features are disabled due to security concerns and some have never even been used [13]. The script (including both senders’ and receivers’ parts) for the most popular transaction in Bitcoin looks like this:

```
<sig> <pubKey> OP_DUP OP_HASH160 <pubKeyHash> OP_EQUALVERIFY OP_CHECKSIG.
```

The script is 164 bytes long whereas its only purpose is to check if the receiver possess the secret key required to verify his signature.

## 3 The CryptoNote Technology

Now that we have covered the limitations of the Bitcoin technology, we will concentrate on presenting the features of CryptoNote.

## 4 Untraceable Transactions

In this section we propose a scheme of fully anonymous transactions satisfying both untraceability and unlinkability conditions. An important feature of our solution is its autonomy: the sender is not required to cooperate with other users or a trusted third party to make his transactions; hence each participant produces a cover traffic independently.

### 4.1 Literature review

Our scheme relies on the cryptographic primitive called a *group signature*. First presented by D. Chaum and E. van Heyst [19], it allows a user to sign his message on behalf of the group. After signing the message the user provides (for verification purposes) not his own single public

---

<sup>1</sup>This is so-called “soft limit” — the reference client restriction for creating new blocks. Hard maximum of *possible* blocksize was 1 MB