Dimecash: A Language Coin

John Rigler
john@rigler.org

Abstract. A fork of litecoin which puts special restriction on how proof-of-burn operates so a new token is created and awarded to an address each time the combined value of one entire coin is burned. Since blockchain transactions are inherently irreversible, an erroneous transfer to an Obviously Unspendable Crypto-Currency Address (OUCA) would normally be permanent. By creating the new token, this problem is eliminated. Advanced Satoshi Codes (ASC) are used to override this behavior when pure burning for financial regulation is required. The currency can safely begin to function as linguistic ledger pages. The mechanism of burning bitcoin is built upon the mathematical truism that an obviously readable address could never have been hashed from a valid private key.

This system greatly increases the use of unspendable addresses and normally danger exits that a large sum of money could be accidentally burned in a system that draws such attention to these addresses (especially when their readable message is anything other than a reference to the actual burn process). Once an unspendable address has been rendered onto the ledger, it will have an associated first transaction. This transaction owner might be the intended recipient of the transaction. An affinity between this address and the unspendable address could then be created and manipulated via various systems. Because Advanced Satoshi Codes are quite easy to send, a series of transactions could be presented with a revealed off-chain secret to legally argue intent. The process of creating the secrets is programmatically repeatable and is linted by the shell when it is typed in. It need not ever be written to disk, because the entire process can be done in the shell environment and printed off. For that matter, a secret could be coded on a typewriter or hand-written.

Example:

https://dogechain.info/address/DCULToFTHEDoLPHiNZZZZZZZZZZZZLkeq5

I simply use https://github.com/johnrigler/unspendable to create a human-readable address and then send it various code, like weak hashes:

```
bash-3.2$ declare -f Hi.Randy
Hi.Randy ()
{
   pwd;
   ls;
   : comments go here
bash-3.2$ . Hi.Randy-33494
```

```
bash-3.2$ declare -f Hi.Randy
Hi.Randy ()
{
    pwd;
    ls;
    : comments go here
}
bash-3.2$ declare -f Hi.Randy | sum
33494 1
```

This function could be loaded into memory and checked against the ledger before being executed, or it could be printed off and possible even used in a court case as evidence. If Randy and I had forged a contract into the comments of the function, I argue that it could be used in court. The point is that this hack used an off-chain secret to accomplish quite a bit with a smart contract and the average person would be able to read and understand this system at its most atomic level, which is significant.

Glossary:

Obviously Unspendable Crypto-currency Address (OUCA)
Advanced Satoshi Codes (ASC)
Secret Sharing (SS)