Communicating Anonymously via E-Mail

May 6, 2017

What is anonymity?

- In colloquial use, "anonymous" is used to describe situations where the acting person's name is unknown. [1]

- Anonymity is seen as a technique, or a way of realizing, certain other values, such as privacy, or liberty.

- An important example for anonymity being not only protected, but enforced by law is probably the vote in free elections. There are also various situations in which a person might choose to withhold their identity: whistleblowing, breaking the law, charity.

- Anonymity is always within a set. In mathematics, in reference to an arbitrary element (e.g., a human, an object, a computer), within a well-defined set (called the "anonymity set"), "anonymity" of that element refers to the property of that element of not being identifiable within this set. If it is not identifiable, then the element is said to be "anonymous."

- Anonymity can be expressed in degrees. There were two papers that put forth the idea of using entropy as the basis for formally measuring anonymity: "Towards an Information Theoretic Metric for Anonymity", and "Towards Measuring Anonymity".

- Anonymity is not just hiding, but also deniability.

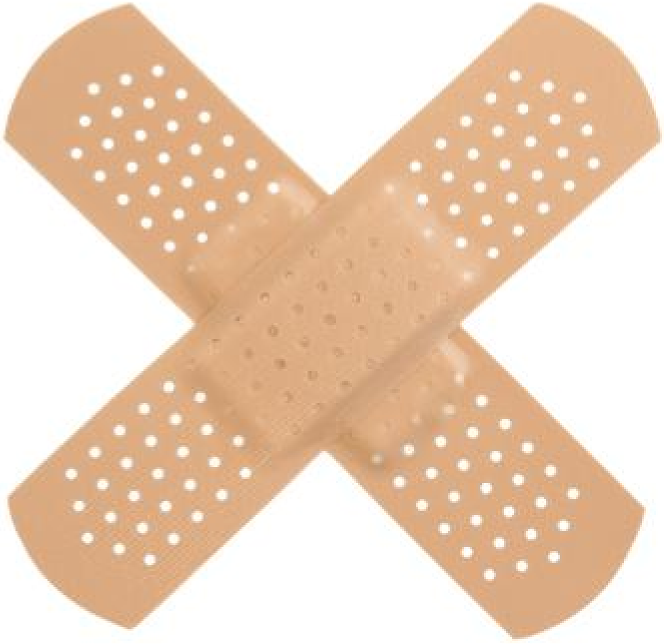
E-Mail Anonymity Issues

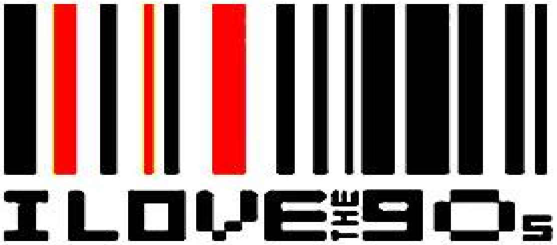
Centralized in practice

Plain text by default

Forgery possible

- GPG for **Encryption**
- Run your own server for **Decentralization**
- SPF/Sender ID/DKIM for **Forgery**



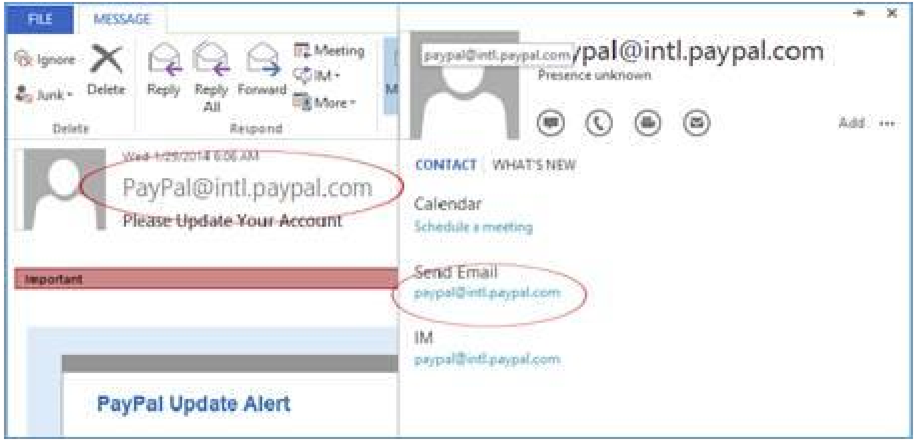


“In 1997, at the dawn of the internet’s potential, the working hypothesis for privacy enhancing technology was simple: we’d develop really flexible power tools for ourselves, and then teach everyone to be like us. *Everyone sending messages to each other would just need to understand the basic principles of cryptography…*”

“...I was excited about the future, and I dreamed of a world where *everyone* would install GPG. Now I’m still excited about the future, but I dream of a world where *I* can uninstall it.” [3]

-Moxie Marlinspike

“Although their use is increasing, estimates vary widely as to what percentage of emails have no form of domain authentication: from 8.6% to "almost half", but to effectively stop forged email being delivered, receiving mail systems also need to be configured to check this authentication.” [2]



Wed 1/29/2014 6:06 AM

PayPal@intl.paypal.com

Please Update Your Account

Important

PayPal Update Alert

paypal@intl.paypal.com

Presence unknown

Calendar

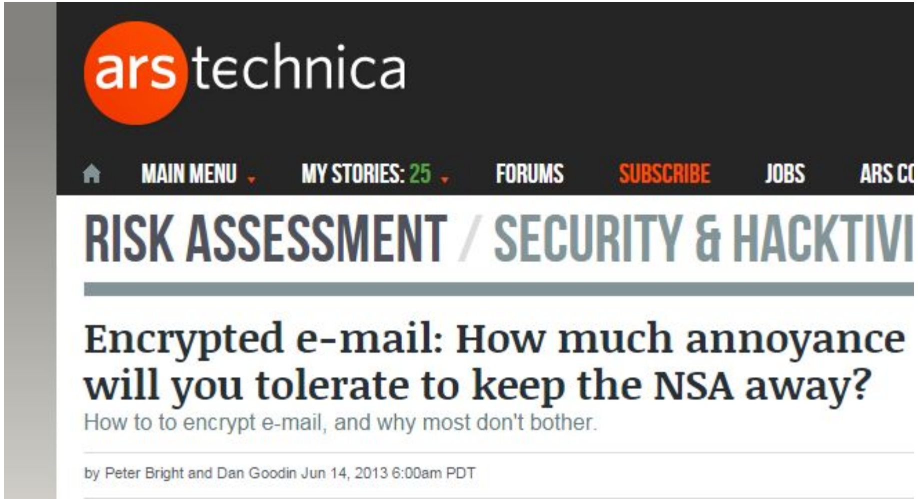
Schedule a meeting

Send Email

paypal@intl.paypal.com

IM

paypal@intl.paypal.com



Encrypted e-mail: How much annoyance will you tolerate to keep the NSA away?

How to to encrypt e-mail, and why most don't bother.

by Peter Bright and Dan Goodin Jun 14, 2013 6:00am PDT

“...e-mail isn't a very good system for secure communications. You're wholly dependent on other people doing the right thing and sending you properly encrypted mail.” [4]

-Peter Bright and Dan Goodin

"I don't have to listen to your phone calls to know what you're doing. If I know every single phone call that you made, I am able to determine every single person you talked to. I can get a pattern about your life that is very, very intrusive.”

-Joe Biden

Bitmessage: A Peer-to-Peer Message Authentication and Delivery System

“We propose a system that allows users to securely send and receive messages, and subscribe to broadcast messages, using ***a trustless decentralized peer-to-peer protocol***. Users need not exchange any data beyond a relatively short (around 36 character) address to ensure security and they need not have any concept of public or private keys to use the system. It is also ***designed to mask non-content data, like the sender and receiver of messages***, from those not involved in the communication.” [5]

-Jonathan Warren

Authentication Against Forgery

- Users exchange a hash of a public key that also functions as the user's address. Encoded with base58 and prepended with recognizable characters (like BM for Bitmessage), similar to Bitcoin, an example address would be:

  BM-2nTX1KchxgnmHvy9ntCN9r7sgKTraxczzyE

- This address format is superior to email in that it guarantees that a message from a particular user or organization did, in fact, come from them. ***The sender of a message cannot be spoofed.***

Anonymized Message Transfer

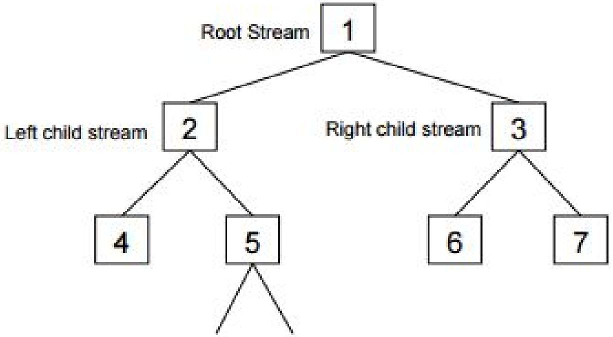
- The transfer mechanism is similar to Bitcoin’s transaction and block transfer system but with a proof-of-work for each message.

- **All users receive all messages.** They are responsible for attempting to decode each message with each of their private keys to see whether the message is bound for them.

- If you send and receive all messages, there’s no way to know which, if any, messages were meant for you!

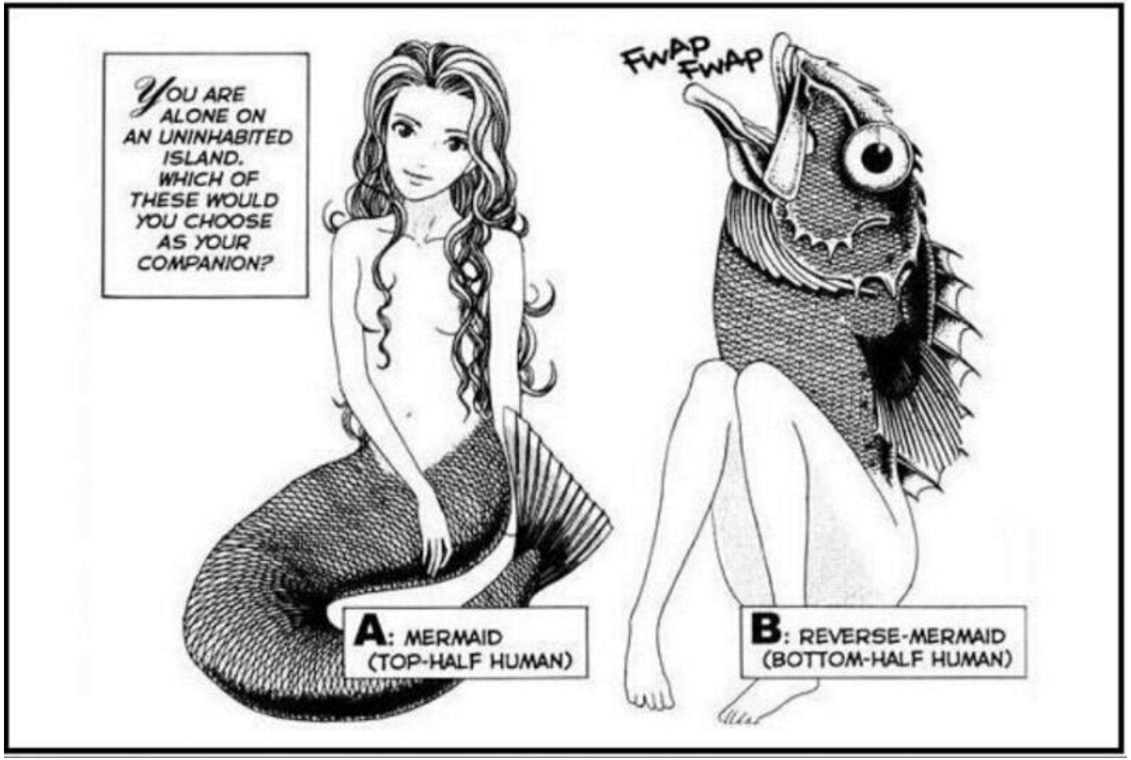
Scalability

- If all nodes receive all messages, it is natural to be concerned about the system’s scalability. After the number of messages being sent through the Bitmessage network reaches a certain threshold, **nodes begin to self-segregate into large clusters or streams**.

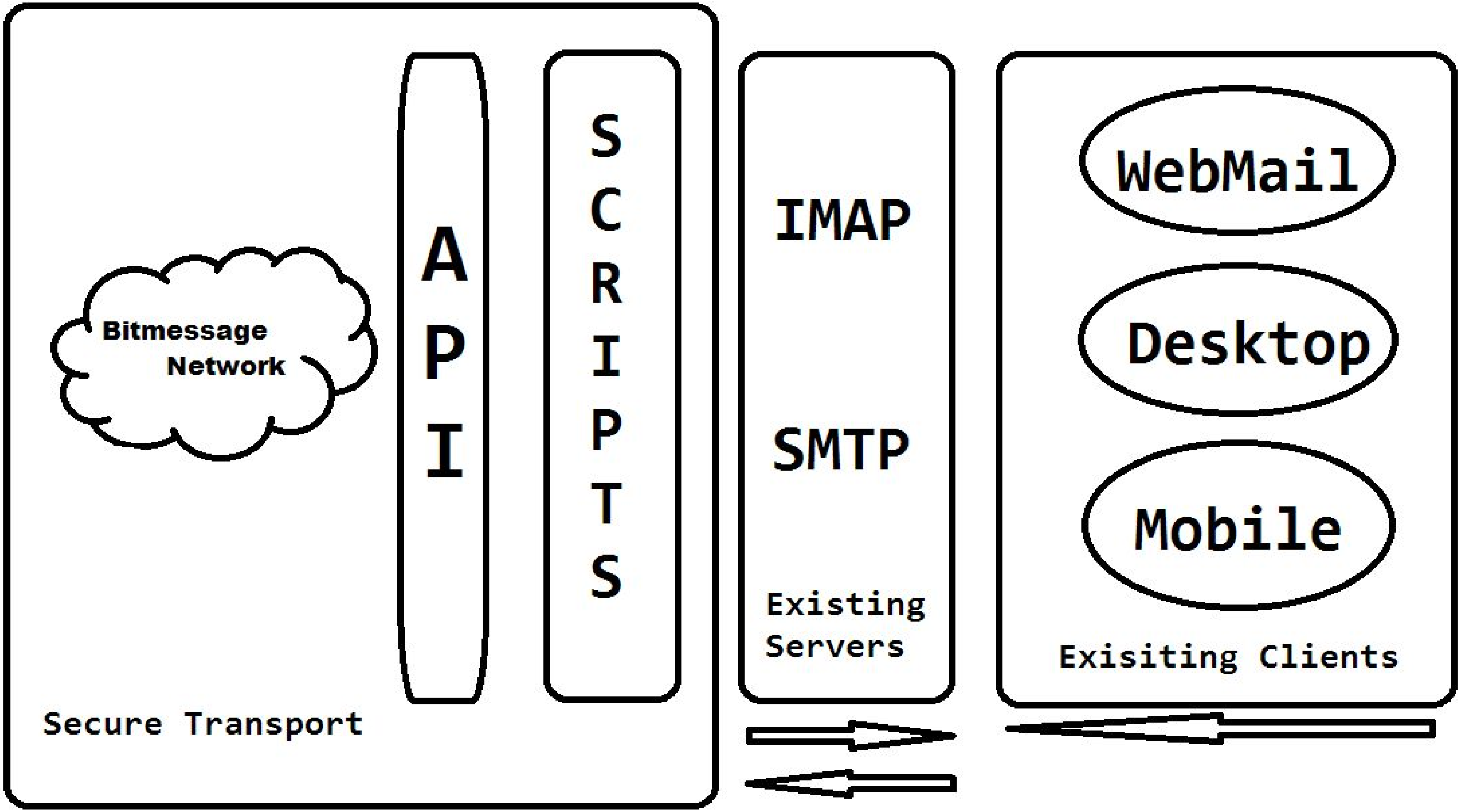


- **A Bitmessage client should use a negligible amount of hard drive space and processing power.** Once it starts exceeding comfortable thresholds, new addresses should be created in child streams and the nodes creating those addresses should consider themselves to be members of that stream and behave as such.

Usability vs Security


YOU ARE ALONE ON AN UNINHABITED ISLAND. WHICH OF THESE WOULD YOU CHOOSE AS YOUR COMPANION?
FWAP FWAP
A: MERMAID (TOP-HALF HUMAN)
B: REVERSE-MERMAID (BOTTOM-HALF HUMAN)

Let’s just take our favorite e-mail clients and bridge them to use Bitmessage as a transport!


Bitmessage Network
API
SCRIPTS
Secure Transport
IMAP
SMTP
Existing Servers
WebMail
Desktop
Mobile
Exisiting Clients

E-Mail over Bitmessage

The Bitmessage reference client already provides an api server to send and receive messages. Scripts that hit the API via RPC + cron give us an up to date maildir of messages and redirect incoming e-mails from SMTP to the Bitmessage network.

Reference Client: https://bitmessage.org/wiki/Main_Page

Daemon Mode: https://bitmessage.org/wiki/Daemon

API Reference: https://bitmessage.org/wiki/API_Reference

API Scripts: https://github.com/alexmat/bitmessage-api-scripts

Using Existing Servers

- Any existing IMAP server can sync clients with your maildir. Here’s my recommendation: http://wiki.dovecot.org/QuickConfiguration
- SMTP is a bit trickier, but thanks to EmailRelay, we can preprocess emails as they come in and feed them to our scripts. http://emailrelay.sourceforge.net/
- Here’s the command I’m using:

  -
    ```
    emailrelay --as-proxy localhost:smtp --port $EMAILRELAYPORT --server-tls $CERTPATH --remote-clients --user $LOWPRIV --pid-file $PID --filter $MAILTOBMSGSCRIPT
    ```

Issues With This Setup

- **You need a server.** However, Raspberry Pi’s and old laptops are cheapish.
- **Setup is not trivial.** This could be mitigated by distributing a live distro or container that’s mostly plug and play. Or use a hosted service, for example: https://bitmessage.ch/
- **Keys are stored on the server, which means you need to run it yourself or trust someone.**

Vice Versa: Bitmessage over E-mail

Mailchuck.com is a relay between email and Bitmessage, combining the privacy and security of a local Bitmessage client and an email account.

Mailchuck allows you to use a Bitmessage client and access the “email network” (send/receive messages with email users), and Mailchuck does not store them other than for transport purposes.

https://github.com/PeterSurda/bitmessage-email-gateway

DEMO! Msg me!

BM-2cU1wivGz8Gsc5HighLS3vbAAJ4DrDJLPR

References

[1] https://en.wikipedia.org/wiki/Anonymity

[2] https://en.wikipedia.org/wiki/Email_spoofing

[3] http://www.thoughtcrime.org/blog/gpg-and-me/

[4] http://arstechnica.com/security/2013/06/encrypted-e-mail-how-much-annoyance-will-you-tolerate-to-keep-the-nsa-away/

[5] https://bitmessage.org/bitmessage.pdf