1 Hello, David

My goal for this document is for it to contain most of our thoughts and sort of iteratively converge to a good final document.

2 Goals of Anonymity

There are three main properties that a anonymization system might want to satisfy.

NOTES: Lets cross-check this against the literature.

2.1 Provider Anonymity

The server cannot determine the identity of the client.

2.2 Encrypted to Proxy

Any proxies invovled have no way of figuring out the data. This will of course require that the client get some sort of public key for the server.

2.3 Server Unawares

The server is unaware that it is being commuicated with via a proxy.

2.4 Unlinkable

An adversary with some level of knowledge of the whole system cannot determine who is comunicating with whom.

3 Scenarios

In this section, we present scenarios that would warrant the properties above.

3.1 End-to-End Connection

Two people know each other's addresses and need to stay unlinkable.

3.2 Unsecure chat room.

You want to connect to a Free Opressistan Forum. You probably don't want your password visible, and you don't want to be linkable to the site either.

4 Designing a proxy system for XIA

In this section we present several ways of organizing a proxy service.

4.1 AD-Provided Proxy

AD routes based on SID, which we trust them not to link to us.

4.2 Multi-Route Proxy Service

Your requests all take different routes.

4.3 Bound-Route Proxy Service

Your requests all take the same route.

4.4 Altruistic Onion Routing

How should Tor work in an XIA setting?

5 The trouble with SIDs

In this section we talk about the "I'll only encrypt with the adressee public key" pitfall.

6 PDRSA

Fuck this section.

6.1 Algorithm

Here we present the algorithm for Piecewise-Decrypt RSA. The main difference now is that there are now two private keys, one of which is kept and the other of which is passed to the proxy. I forgot why this distinction is important. I may have wasted a few hours on this.

6.1.1 Key generation

Let p, q be large primes like in RSA. Define $n = pq$. Note that $\phi(n) = (p-1)(q-1)$. Now select e, d_2 that are coprime with $\phi(n)$. This implies that ed_2 is also coprime with $\phi(n)$. While e can be selected to minimize the cost of encryption, d_2 must be selected randomly. Let $d_2 = (d_1 e)^{-1} \mod (p-1)(q-1)$.

Publish (e, n) as the public key. (d_2, n) is the semiprivate key. (d_1, n) is the private key.

6.1.2 Encryption

$c = m^e \mod n$

6.1.3 First Pass Decryption

$h = c^{d_2} \mod n$

6.1.4 Final Decryption

$m = h^{d_1} \mod n$

6.2 Proof of Correctness

Pretty similar to the one on wikipedia for RSA. Fermat's is easier to understand, but Euler's is shorter.