Solving Generative AI Misinformation & Other Sucky Internet Problems

This can't work. In fact, I'm certain it can't, because if it could, someone would've proposed it already and I've spent at least 5 fruitless minutes googling it. There are millions of people working on the amorphous collection of technologies we call “the internet”, a few of them are even quite clever, so the odds of me, an *intellectual*, yes, but far from a cryptography or internet architecture expert, has stumbled across a workable solution to stem the flow of both AI-generated and artisanal, small-batch misinformation, isn’t even worth finishing a sent

Despite this certainty I am haunted by not knowing *why* it can’t work. Thus, you are reading my proposal to address many of the modern internet’s key problems with misinformation by way of **cryptographically assured attribution**.

Or, put another way, **what would change if you knew the origin of any piece of content on the internet?**

Granted, even if this did work, it wouldn’t stop the world from ending. It just might end for all the *human* reasons we already know about (though continue to ignore) and not the Skynetesque uncertainty that nascent generative AI promises. The devil you know, etc, etc.

So I offer unto you this unmissable opportunity to prove me wrong and not even have to feel guilty about it afterwards since I’m practically begging for it.

Internet Suckage (The Problem)

The internet sucks. This is so well-established it was on Moses’s goddam tablets, so I'm not going to belabor this section as much as I've stretched this sentence beyond all reasonable comprehension.

You've *used* the internet lately, right? Phishing emails, social media scams, misinformation WhatsApp groups your cousin invited you to, deepfake videos of Biden raising his pinky while drinking a latte. All of these go from "be careful what you believe online" to "my god, I'm in the Marianas Trench of misinformation" when generative AI gets involved, eliminating the inconvenient and costly use of *humans* to generate all this "content".

There's just no ~~love~~ trust. Who *really* sent me this email? Is this Facebook message about a $30/hr online job *really* from the Geico gecko? How did Biden raise his pinky when I saw him lose the same digit last week on TikTok?

So what if we solved the problem of **trusting content online** with ***math***?

* If you're already familiar with cryptography, key-pair signing, and TLS communications, skip ahead a bit. But you *will* miss out on a few feeble attempts at humor.

Cryptographic Attribution (The Solution)

My desire to use (or invent) fancy technical jargon outweighs my desire to make this approachable.

Look, it’s not much different from your signature on a cheque or your fingerprint to unlock your phone. You are using elements of yourself that are unique attributes demonstrating your physical involvement in a transaction or process.

But signatures can be forged or ignored (ever sign a cheque as “Mickey Mouse” and watch the bank honor it? I have!) and fingerprints can be… relocated unwillingly. Also you have a maximum of 10 of them.

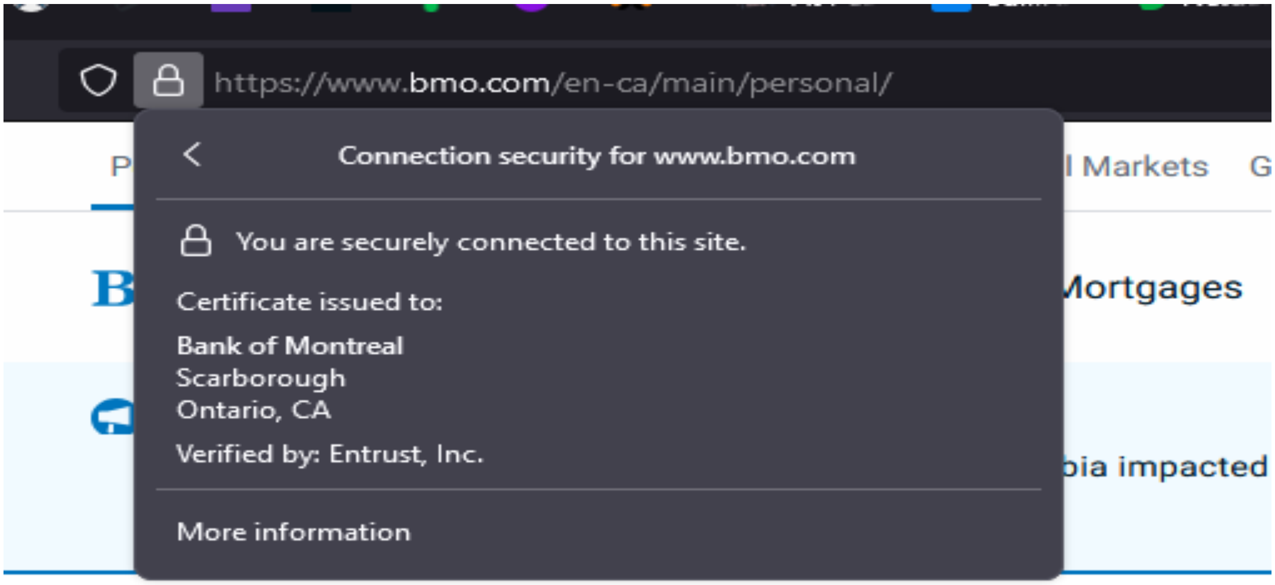
Cryptographic attribution uses extravagantly fancy math to do the same thing without the limitations. A letter signed with a sufficiently modern cryptographic cipher allows for perfect trust that it came from the only person capable of creating that signature. Any change to the letter permanently invalidates the signature, signaling to the recipient it can’t be trusted.

Allow me a self-indulgent demonstration. You can download this essay as a PDF which I have signed with my personal cryptographic key. No one else can generate this signature because only I have the key to do so and you can be certain the contents haven’t been altered since the signature is intact.*

* Yes, yes, quiet in the back, I’m well aware this is a savage simplification and your kind hate simplicity even more than you hate Hitler. Your feelings are valid but I will not respond to them.

The Good News (We Have the Technology!)

Thankfully, none of this is new technology. This isn't the Web 4.0 Blockhectachain, it's fundamentally the same technology you use to make sure your bank website is *actually* being sent from your bank.


https://www.bmo.com/en-ca/main/personal/

Connection security for www.bmo.com

You are securely connected to this site.

Certificate issued to:

Bank of Montreal
Scarborough
Ontario, CA

Verified by: Entrust, Inc.

More information

When you visit an HTTPS website, which at this point is virtually all of them, your browser will display some sort of lock icon next to the address bar. You can click on this to verify that the content being served to you is coming from the organization it should be. Much like my PDF, you can verify that your bank has "signed" this website and it has not been altered on its way to you.

Ok, so when I see Biden defeat Trump in a Macarena dance battle as a sponsored tweet or shared with me on WhatsApp with a snide message about Trump's apparent lack of enthusiasm for the 45-degree hop, why can't I see the signature proving it did, in fact, come from CNN? After all, the AI that created it is smart enough to include the CNN logo, plausible news ticker, and insufferable tie choices closely associated with mainstream news media.

Sadly, in the scope of this discussion and common internet use, we really only use cryptographic attribution to establish trust between your web browser (or app) and the server delivering the content. In the case of our feisty Latin tweet, the content is signed by Twitter. You can be certain it came from Elon's personal meme collection but you have no way of knowing where *he* got it from (let's pretend for the sake of the example it's not already a certainty he stole it from 8chan).

Dream Bigger (By Getting More Granular)

So let's dream a bit bigger. **Why not sign individual pieces of content?** If everyone has a cryptographic key they can sign anything they distribute online, even the dankest memes. Let's go through how this would work, step-by-step:

1. Jake has a bad idea for a meme so he opens Photoshop, scribbles something that is somehow both incomprehensible and deeply

offensive to Yemenis, and clicks Save

2. After choosing a location and format to save the image Jake is prompted to provide a key to sign the image.
3. He can sign it with any key in his possession, generate a brand new key, or simply leave the file un-signed
4. Jake uploads the image to Twitter and immediately Likes his own post (that last part's not cryptographically relevant)

Now, when an unfortunate soul such as yourself comes across this meme, you can inspect the content to see what's called the **chain of trust**:

1. The file was first signed by <JAKE'S DARKWEB MEME KEY>
2. The signature was verified by and then also signed by <TWITTER KEY>

You now have **trusted attribution**. Since public-facing organizations like news media want to be recognized online, they will provide their signature publicly for validation (this is called a "public key", capable of verifying the authenticity of a signature (also called a "private key") but incapable of *producing* the signature). In the case of our jiving presidential candidates, a user could inspect the video to see what key signed it and use CNN's public key to confirm it is **inauthentic**.

Over time, it's likely that, much like HTTPS and the lock icon, browsers will provide built-in tools to automatically authenticate and flag content that can be cryptographically attributed. Social media sites like Twitter could even do this themselves, potentially boosting the visibility of content attributed to legitimate organizations to discourage misinformation.

As attributable content is inherently more valuable* (especially to advertisers who continue to run the internet's economy), established internet megacorps will be incentivized to adopt and support trusted attribution and content creators also benefit from their content being attributable*.

* For example, Google boosts the search rankings of HTTPS websites *because* they're trusted since SesameStreet.com is unlikely to distribute malware

* Mailing yourself your manuscript to establish copyright may not be a real thing, but cryptographic signatures could contain the signing date, providing proof of who created a piece of content and when, which will sure speed up copyright lawsuits

The Best Part (Be True to Yourself, Dear Internet)

This proposal is, in all the most important ways, truly an *extension* of the internet that already exists. This isn't a radical reinvention of how information moves around the internet and no new fiber cables need to be laid. The government doesn't have to know which keys are yours and your real identity is only tied to your key(s) if you proactively make it so. If you wish to have no association at all with your content, simply don't sign your content or use a randomized key.

This approach is based on established concepts like **chain of trust** which we already use for many forms of secure communication and the history of HTTPS laid the groundwork for browsers recognizing, authenticating, and highlighting trust signals to give users confidence in the safety of their interactions.

What I'm saying is, all this has happened before (on the internet) and all of it will happen again (on the internet).

One Colossal Caveat (The Part Where the Politicians Have to Do Something)

I haven't talked about AI-generated content as much as I expected to by now.

Why wouldn't AI-generated content just be unsigned and shared freely? Or, since generating keys is effectively free, just sign every image with a random signature?

Yup. So for the sake of the stability of society, our political systems, and my rapidly deteriorating sanity, **regulations must be enacted requiring commercial generative AIs to register their keys publicly and sign *all***

generated content with one of them.*

Once in place, this regulation means that AI-generated content can be identified by anyone, especially as web browsers like Chrome and Firefox will rush to add features highlighting provably-AI-generated content to the user.

This will not be perfectly enforceable. Already there are open-source generative AI models that can be run on a home PC and would be impossible to regulate (and I would argue we shouldn't anyway). However, running these models is relatively expensive and requires specialized knowledge, dramatically reducing the outfits who could use them effectively at scale.

The Part Where I Find Out This *Is* Being Attempted

As it turns out, between writing this and publishing it, I've discovered that the C2PA standard has been created in an attempt to address this very problem in a similar way to what I've proposed. I guess I will take that as a win.

Overview - C2PA

An open technical standard providing publishers, creators, and consumers the ability to trace the origin of different…

C2PA —