
Arghus: automated verification defenses against scammers and identity thieves¹

Josh Stein
BlueDot Impact

With
Apart Research

Abstract

Look. I didn't set out on a Hackathon with the expectation that I'd be writing a report; I thought I was supposed to be hacking?

I built a man-in-the-middle automated defense system against deepfake scammers. Think of people who are cloning your voice and then scamming your grandma out of her pension. I take this seriously because it symbolises a breakdown of trust and authenticity in society. We need to know that we can trust each other, and that we are who we say we are.

1. Introduction

Deepfakes are scary. You can make a realistic clone of almost anyone's voice with about 30s of audio samples nowadays. I'm worried about that. I hope you are too; or at least you will be once your 'mom' calls you in a distressed fit talking about how she needs cash ASAP.

I'm focussing on this because it represents deeper societal issues. How can we build relationships if we don't know who we can trust? How can we verify the other? Verifying yourself is a solved problem; we have 2FA, MFA, magic links, etc. But verifying the other can be hard, and it's only going to get more difficult.

We need more robust solutions to authenticity.

¹ Research conducted at the def/acc Hackathon, 2025

2. Methods

Core idea: you have shared dynamic secrets between you and trusted contacts. You may have DM with your cofounder - you can scrape that information to generate recent secrets (e.g. what book did you tell me that you started reading yesterday)?

Even if someone makes a deep clone of your cofounder's voice, they won't be able to pass a test asking them questions about this information.

Technically, we route all calls from unknown callers to a private number using a combination of silencing unknown callers and conditional call forwarding. This is achievable on almost all phones across almost all carriers. The private number is handled via Twilio on a FastAPI server. Once connection is established, we route audio to the OpenAI Realtime API, with a model prompted to determine the reason for calling and if there is possible scam.

As the model evaluates the discussion real-time notifications are streamed to the user's actual phone (mobile app written in React Native, streaming via Supabase). The user is alerted that a potential scam is happening and can see the transcript as the scammer engages with the realtime model.

If the scammer is flagged as suspicious a dynamic verification question is generated and asked; if the verification is passed the call is connected to your true number. If not the call is hung up and you have been protected.

Code code code: <https://github.com/joshestein/arghus>

3. Results

Most of this is audio related. I don't think you really want screenshots of my mobile app. Feel free to listen to my recording that I submitted though.

Call +15187223932 or +447450307731 now and experience my secure voicemail for yourself!

You can pretend you are 'David'. The answer is 'purple'.

4. Discussion and Conclusion

Bad guys are bad. They are going to clone you and scam your loved ones. Let's not let that happen.

The core idea of using dynamic shared secrets can be a useful utility to thwart any deepfake (or any AI trying to fool you into thinking they are real). Having it happen completely automatically is what our future defences should look like.

5. References

Look guys. It's almost midnight on a Sunday night. I don't know why I'm still here. I wrote some cool code. Yes, I could find a reference for you that

deepfake audio could be generated in 30s. But I'm too tired for that. If you are really interested I can write about this more seriously.

6. Security Considerations

Limitations: requires you to block unknown callers. Can be slow depending on where you are calling from. Dynamically secrets may not be so easy to generate. Certain platforms are hard to scrape, or it is against their terms of service.

Honestly I think services like this should just be baked into our calling infrastructure, the world would be better off. I don't think there are any security worries. If a deepfake passes the verification you would've been equally screwed without the verification in place.

Future: basically just integrate with as many forms of communication as you can to generate good secrets. Onboarding experience to connect with your loved ones. Better control/intervention into routed call via app. Full transcription analysis. Detecting fishy audio patterns and using those to train models specifically against deep fake audio. Goodnight now.