Hidden commands audio attacks on Alexa and Siri

GROUP - 18

Josh Hofing, Rahul Deshpande, Naveen Kalloor Vijay, Zhenfeng Qi, Yitao Zhou
jlh627 rd2589 nkv223 zq419 yz4573

Abstract – **AI assistants such as google home, Alexa, Siri etc. are widely used in our phones and homes to accomplish many tasks. But the way AI assistants are manipulated and are used the way they are, exposes many security issues. One can generally send ultrasonic waves to trigger them, without being noticed by the owner and therefore make it vulnerable to attacks. Therefore, we came up with an idea to make this system more secure by setting a voice password so that the AI assistants won't get triggered by ultrasonic waves or by any other means. Furthermore, we have discussed the design, implementation and other solutions which could help make the system attack free.**

I. INTRODUCTION

Virtual assistant is becoming more and more popular these days. It can provide services like playing music and videos, providing information of weather, setting calendars and reminders, controlling smart light and door locks, and even purchasing groceries online.

As the virtual assistant can control so many things, it would be very dangerous if it is controlled by others. Recently, researchers found virtual assistants, for example Apple Siri and Amazon Alexa, not only listen to human voice, but also "hidden commands in white noise played over loudspeakers and through YouTube videos to get smart devices to work"[1].

Here the user's property is at risk whenever other people can send hidden commands to the virtual assistant. They can send command to open the door, send money, or place orders online. Hence, the user's intergrity should be protected under such circumstances. The solution to this problem is a voice password to activate the virtual assistant for any private services. The solution ensures other people who does not know the password, are not able to activate the assistant for private services, so that they cannot request other services. Some attempts have been made to implement the virtual assistant password strategy. But most of them are utilized on mobile devices, for example iPhone and Android Phone. The virtual assistant required phones to be unlocked to process further instructions. However, on smart speakers, for example Google Home and Amazon Alexa, such password strategy haven't been implemented.

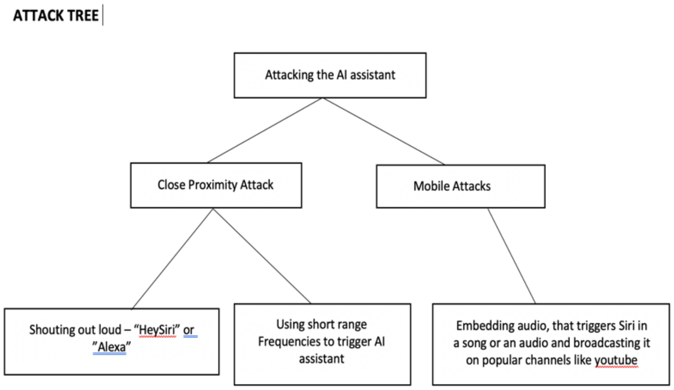
But still, we believe this technique is a key solution not only for the problem of hidden command in white noise, but also for the problem of 'Dolphin' attacks, that "voice-controlled assistants by Amazon, Apple and Google could be hijacked by ultrasonic audio commands that humans cannot hear"[2]. In this paper, we discussed the implementation of the solution, which is a voice password to activate the virtual assistant for any private services.

II. BACKGROUND

A trivial and simple solution for the problem would be adding voice authentication function to the smart devices, i.e. the virtual assistant should be able to verify the speaker's identity at any time. However, this solution requires too much computation resource to keep the AI assistant on a consistent computing stage, there may be no room for the CPU to recognize the command and do other things, so it's not a good idea. A much more practical solution would be recording the voice password, and the virtual assistant will only require it when encountering sensitive commands, such as sending money, making purchases on Amazon and going to an uncommon website. When facing such commands, the AI assistant would ask for the password to move on. On one hand, the password verification would only be triggered under certain circumstance, which saves a lot of computing resources; On the other hand, this setting would not prevent your friend or family member from using the AI assistant, it's embarrassing that if the assistant can only serve you and refuse every others' command. A lot of work has been done on the field of voice authentication. I.V. Vasyltsov [7] proposed the structure of a voice authentication system. He and his team consist in the usage of a probabilistic approach to generate the authentication system. This will increase the resistance of the voice authentication tool to attacks with pre-recorded voice messages. Sattar B. Sadkhan[8] found that the template of stored biometric members of the biometric authentication system is a dangerous issue, while it can be stolen or breached. They tried to make an auto-evaluation system which can evaluate the biometric voice template automatically and providing a report about expected accuracy and security. Chenguang Yang [9] found that the entropy of the text-independent human voice is limited. Under this condition one needs to be very careful when they want to use the human voice for the authentication task.

III. THREAT MODEL

In the previous paper, we used Attack tree to show the potential risks. Generally, from the report [1] there are two kinds of attacks: Close Proximity Attack and Mobile Attacks. Let's review them and see how our solution can address them.


ATTACK TREE
Attacking the AI assistant
Close Proximity Attack
Mobile Attacks
Shouting out loud – "HeySiri" or "Alexa"
Using short range Frequencies to trigger AI assistant
Embedding audio, that triggers Siri in a song or an audio and broadcasting it on popular channels like youtube

Threat	**Solution Proposed**
Shouting out loud and using Short range frequencies	Even the attacker is close enough to trigger the voice password authentication step, the attacker who doesn't know the password and not the owner's voice cannot send sensitive command.
Embedding audio	Since everyone differs in their password and voice frequency, if the attacker want to crack one's AI assistant, he(she) requires to embed the specific password into the audio and hope the victim would watch it, which is a small-probability event.

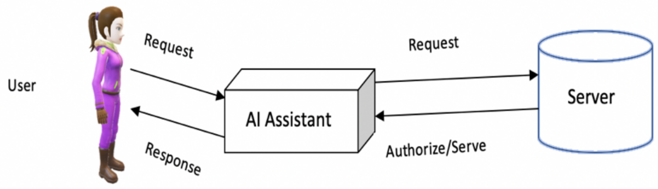
With reference to the attack tree above, solutions proposed

IV. DESIGN

The proposed solution needs to be scalable, secure and easy to use. The idea to have a voice passcode, helps overcome the AI assistant triggered by ultrasonic waves. This system involves the following components –

- *Client* – who tells the AI assistant the commands
- *Server* – which reads the users request and has some intelligence in it to distinguish between the different kinds of request to serve.

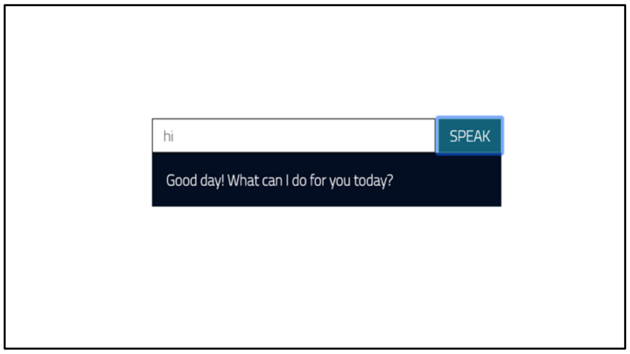
Ex – if you tell the AI assistant "Hi", "How are you", the server will reply back with the appropriate answer. If you ask the assistant "Send Money", it will further ask for authorization, where in you tell your voice passcode and then if its correct, it will allow you to send money.



V. IMPLEMENTATION

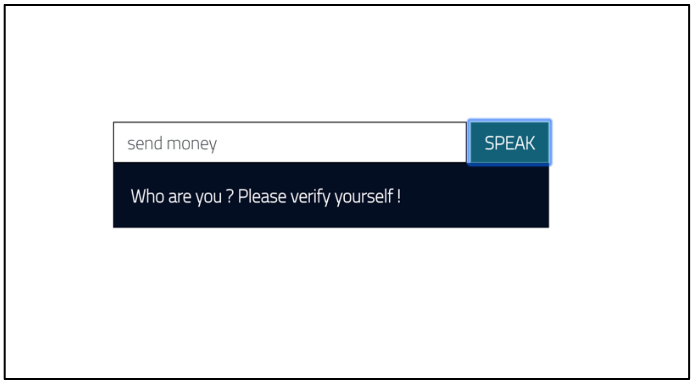
The implementation of voice passcode for the AI assistant involves, storing the voice passcode in the server. The Authorization part only comes into play when the user asks for the things which are sensitive like sending money, buying things online etc. The things that should be considered here while building this system is, the user may want to change the voice passcode at any time. So, an edit voice code functionality to change the voice code is an extra add on to the normal AI assistants.

The prototype of the above system was built as a website and has been deployed on the URL : **https://ai-web-c7224.firebaseapp.com/public/index.html**. Below are the screenshots and steps that can be used to secure the AI Assistants. (inspired from [10])


hi

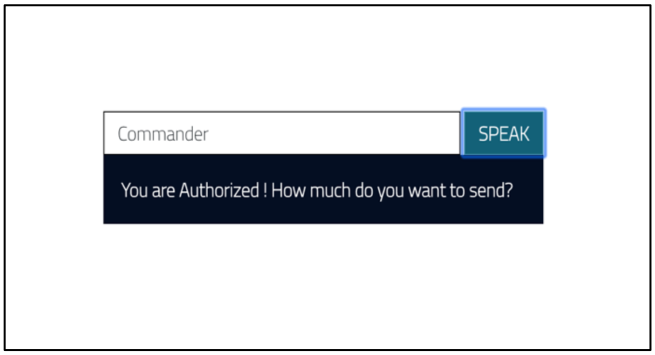
Good day! What can I do for you today?

When you click on speak and say normal things like “Hi” or “How are you “, the AI assistant responds normally.


send money

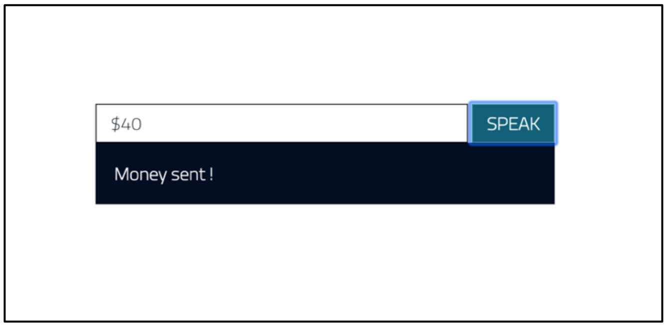
Who are you ? Please verify yourself !

When you want to send money and you ask your voice assistant to do it for you, it won’t let you do it unless you are authorized, so it asks for voice passcode, which I have set it as “Commander”.


Commander

You are Authorized ! How much do you want to send?

So when you say “commander”, it will authorize it against it’s backend server and say that you are authorized, and asks for the amount of money to be sent.


$40

Money sent !

And when you say the amount in dollars, it will say, money sent. Since it is just a prototype, the recipient name is not added, and the system needs to be trained on contacts list I have.

There are a few things that could go wrong in this implementation. The security of the voice code is of utmost importance. The Voice code should not be easily accessible: that is to say, it should not be stored as a simple text translation of the audio or just the audio. As the google assistant implements a button to turn the speaker on/off, could also be used across all the AI assistants to maintain the availability of the system.

Well, even this system is vulnerable, if the voice passcode gets compromised then the perpetrator can do anything he wants to, using the voice passcode.

VI. EVALUATION

The solution provided by us is by no means a perfect solution. There are a number security loopholes that may be exploited by an attacker. For one, the voice password is not specific to any user, which means that anyone who knows the password can perform sensitive operations regardless of whether they are authorized or not. The fact that a password is provided via voice means that there is a high possibility that the an attacker may acquire the password through eavesdropping.

Another problem with voice passwords is that the voice interpreter may not be able to interpret the password in an accent different from the owner’s native accent. This may sometimes result in a system lockout. On the flipside, an AI assistant which recognizes too many accents may result in the system giving sensitive access when the owner did not mean to say the correct password.

Nevertheless, the use of a voice password does mitigate a lot of security problems, and it is a step in the right direction to secure AI assistants.

VII. RELATED SOLUTIONS

Google provided a solution that called Void Match. As Google stated, Voice Match is mainly for customizing media experience, "When each person in the household uses Voice Match, they'll enjoy a more customized media experience, making the overall media experience even better."[3]. However, we believe this approach is a viable solution to many problems. Unfortunately, Google did not regard Voice Match as a security feature. "Kara Stockton on the Google Assistant team offered the following statement over email: 'Users shouldn't rely upon Voice Match as a security feature. It is possible for a user to not be identified, or for a guest to be identified as a connected user. Those cases are rare, but they do exist and we're continuing to work on making the product better.'"[4].

Similarly, Amazon provide a voice recognition feature called 'voice profile', to store the voices of each user. As Amazon stated "Once you create your voice profile, Alexa is able to recognize your voice on most Alexa-enabled devices."[5] "To create, users are asked to read aloud 10 phrases, and Alexa will then use that data to create a voice profile."[6] Although there are 10 phrases that are stored and compared, they are still far less than needed to successfully train any Machine Learning model. Both Google and Amazon's approach are both a mixture of "text-dependent" and "text-independent", because to activate the Google Home or Amazon Alexa, user need to say something like "Hey Google" or "Hey Alexa". Those phrases must be record in the training process. However the rest of the commends are "text-independent", which is a harder approach.

WeChat is a messaging, social media and mobile payment app developed by Tencent. Although it is not a virtual assistant, the approach of voice password is a viable solution. After the user log out an account, the user has an option to use voice password to log back in. "Once you've set up Voiceprint, all you have to do is read a set of digits that you see on the screen to login to WeChat."[6] The drawback of this approach is the "set of digits" is randomly generated and unchangeable. Furthermore, this approach has not been applied to any virtual assistant systems at all.

VIII. CONCLUSION

Voice passwords is an effective mitigation to the threat model of an attacker who would attempt "drive-by" attacks against voice assistants using both vocal and subvocal audio range attacks. The mitigation effectively stops the ability to "mass-attack" voice assistants, forcing attackers to customize their attack to each target. This guards against a number of threats against these assistants, but does not provide a mitigation against attackers attempting to go after a particular target. Future work could further harden the voice assistant to recognize a particular person's voice, adding a "something you are" factor in addition to the "something you know" defensive factor of a voice password.

IX. REFERENCES

[1] Alexa and Siri Can Hear This Hidden Command. You Can't. - https://www.nytimes.com/2018/05/10/technology/alexa-siri-hidden-command-audio-attacks.html

[2] 'Dolphin' attacks fool Amazon, Google voice assistants https://www.bbc.com/news/technology-41188557

[3] Voice Match and media on Google Home https://support.google.com/googlehome/answer/7342711?hl=en

[4] Fooling Amazon and Google's voice recognition isn't hard https://www.cnet.com/news/fooling-amazon-and-googles-voice-recognition-isnt-hard/

[5] About Alexa Voice Profiles https://www.amazon.com/gp/help/customer/display.html?nodeId=202199440

[6] Voiceprint: The New WeChat Password https://blog.wechat.com/2015/05/21/voiceprint-the-new-wechat-password/

[7] I. Vasyltsov, M. Karpinskyy, S. Kavka, "The structure of the voice authentication system". The Experience of Designing and Application of CAD Systems in Microelectronics, 2003. https://ieeexplore.ieee.org/document/1255129/references#references

[8] S. Sadkhan, B. Al-Shukur, A. Mattar, "Biometric voice authentication auto-evaluation system" Annual Conference on New Trends in Information & Communications Technology Applications (NTICT), 2017. https://ieeexplore.ieee.org/document/7976100

[9] Chenguang Yang, "Security in Voice Authentication". https://web.wpi.edu/Pubs/ETD/Available/etd-032714-115410/unrestricted/yang.pdf

[10] https://github.com/sitepoint-editors/Api-AI-Personal-Assistant-Demo/blob/master/README.md