Alexa Data Monitoring: Virtual Personal Assistants and Information Privacy Protection

Chase Lemley, Tu Le, Yuan Tian
University of Virginia
{Crl2dk, tnl6wk, yt2e}@virginia.edu

***Abstract*—With the introduction of the Amazon Alexa and other Virtual Personal Assistants (VAPs) many households have begun adopting this technology. This technology has begun taking the conversations and storing the data, and with the storage of this data Amazon offers methods to see what conversations have been saved with numerous metrics tied to those conversations. With this it has become essential to study VPAs and the method of displaying data. This project has sought out a method to utilize a google extension to better take the data being stored by Amazon for a users Alexa, and give better ways of visualizing what is being stored. We found methods that give a clearer and easier diagram that will allow users to better understand what is being stored, and whether they want to keep or delete the data.**

***Index Terms*—VPA**

I. Introduction

As Voice Personal Assistant (VPA) systems have increased in popularity they have been introduced to more home across the world to assist in everyday life. The devices have been notable for their ability to assit with basic tasks aroudn the house, whether it be for searching for answers to questions, to setting alarms and other mundane tasks. VPAs are able to do so much by communicating with third parties to give and receive information. By the year 2024 it is estimated that there 8.4 billion VPAs globally, a number exceeding our current population, [5].

A big trend in recent years has been for the ability for users to monitor what data is being stored about them, and the ability to find that data, and furthermore delete the data if they deem it to sensitive. This has been seen in numerous fields and have led to numerous technology companies giving more options about usage of cookies and giving users far more control of their own technology [1].

With so many VPA devices circulating, the question became how was the data stored, and how could the user access their own conversations. The current website is shown below (figure 1) and can be seen to be difficult to parse, and fully comprehend exactly what is being given to the user. While the website can work, it does make finding all conversations difficult, and hard to detect when the Alexa has picked up a conversation that it should not have stored. Another reason for concern is that this website and way of tracking data is not readily known by most Amazon users, and therefore users are unable to find where there conversations are being stored by Amazon.

Problem and Motivation - The Amazon Alexa has the ability to record every conversation that has ever been had by it, and store it. Users have the ability to see their previous conversations, but the ways of accessing this data is rather hidden, and once accessed it is not always clear what information may be sensitive, and where the conversation is coming from. With this comes a need to give users the ability to see clearly where their data is coming from, and more importantly how to best decide whether they want to keep or delete their data.

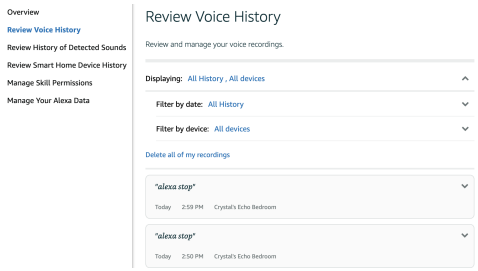

Fig. 1. Alexa conversation history

This lead to a need to create a data dashboard driven by the data stored on the users Alexa account. A requirement was the data must be processed on the users own machine as to not further mitigate the issue of having data stored. Furthermore the data must be presented in a way that would allow for the user to easily look over all the data that was pulled, and to give a clear idea of all aspects recorded by the Alexa to allow users to have the clearest picture of what was being stored and used by Amazon.

Approach -

- Build a script to clearly show the user data being used by Amazon
- Test for ability to properly show the data and give visibility to the user
- Analyze additional pieces of data to show where questionable conversations may have occurred

The goal of this project will be to show a clear improvement of the current technology at hand to visualize the data that is being stored by amazon for users to make informed decisions about data storage preferences.

II. Related Work

Huang and Feamster, a team coming out of princeton, have explored improving data visibility by giving a data visualization dashboard. The goal was to give users the ability

to show what a users IoT devices are communicating with and give a clear graph showcasing exactly this [2].

Huang and Feamster, looked at where these deices were submitting their data, and how frequently they were submitting it. In their lab they gave two examples. The first device outlined was a google chromecast that constantly contacted the google servers shown in Fig. 2. The second device was a smart bulb by Geeni. This device consistently was hitting the cloud using a constantly sending and receiving from tuyaus.com a Chinese own company seen in Fig. 3. These results were achieved in their lab where they were able to generate multiple tests.

With this technology they were able to produce clear graphs that would relay to the users where the data was going, how many KBps were being sent at a time, and when the data was being sent. This allowed users to know exactly how their IoT devices were communicating with other applications, and could help users feel more comfortable in the long run. attached below are example charts for the previously mentioned products.

Further related works on IoT were conducted by schools out of Northeastern university and Imperial College London. There they were able to map data exposures from IoT devices [4]. They did this through 34,586 manually controlled experiments for the devices to help classify the various data leaks from the devices.

Further related work was done to show that users of these devices do not fully understand these devices, whether it be with privacy settings, or with what is done with their data. This group was able to show that 67.9 percent of user knew they could delete their data, but did not now how to do this [3]

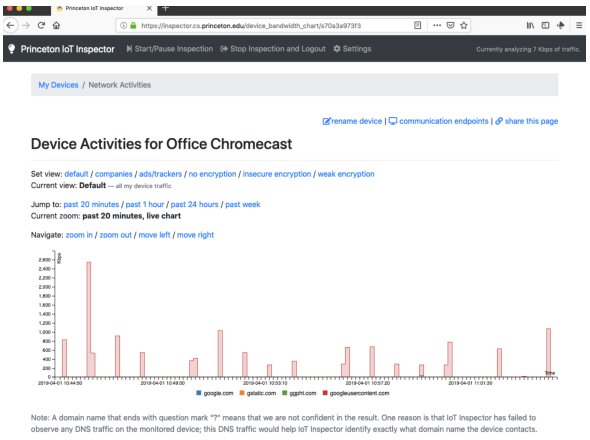

Fig. 2. The graphic related to the Chromecasts contacting google servers

III. Technical Approach

Approach -

- Build google chrome extension to scrap Alexa data
- Build a script to clearly show the user data being used by Amazon

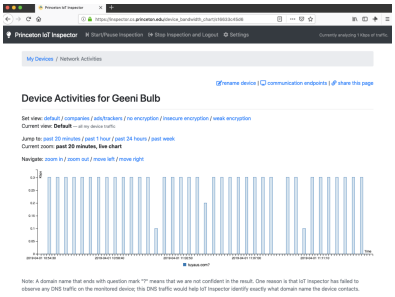

Fig. 3. The graphic related to the Geeni bulb contacting the cloud

- Test for ability to properly show the data and give visibility to the user
- Analyze additional pieces of data to show where questionable conversations may have occurred

The First step in the process was creating a google chrome extension that would hit a users Amazon account to scrape the conversations and data being stored. The method of storage for this was making these into JSON files that took all features stored with a conversation. This process requires user permissions to allow for the web extension to extract the data, and saves the JSON file locally to ensure the data is kept private for the user.

The second step is creating a dashboard for the data that is lightweight and allows all data to be kept on the users machine. The method for this was to use the package Charts in java script. This allowed for easy charts to be generated, without needing installations of other packages, and not requiring intensive processes to generate the chart. The Java script file accesses the local file and generates the chart for users to be able to see the data in an HTML file.

The third step was giving users the ability to customize their dataset. By giving a user interfacing experience through HTMl and CSS we are able to give users a quick and easy method of changing the way the data is displayed to give whatever is necessary for the user to find relevant information to them. Furthermore when an alexa is in use for long periods of time greater amounts of data may be stored, which could clutter a dashboard. This problem was solved by giving only selected nodes for the initial chart and allowing users to zoom in on specific sections to reveal more data that would be close to the zoomed in area.

The fourth and final step is giving users the ability to see what data the Alexa has stored that may be problematic. The key process here was finding key words associated with problematic statements and highlighting it on users dashboards by changing the color of the node on the chart.

IV. Evaluation

The second set of tests were looking into the ability to transform JSON data into a chart.js template, as this is an uncommon practice. The initial results proved successful as converting the JSON values into arrays and using the array values as data set variables for the chart. Once this was

established we then added the process of being able to take a the scraped file from a local repo, and generating the chart in an HTML file to prove that this can be done all locally on any users machine.

The third stage of the process was showing the ability for chart.js to detect key phrases detected as problematic and showing that they were able to be marked differently than normal conversations to quickly alert users. This was done by using the phrase, 'Audio was not intended for this device', this is recorded when the Alexa has picked up something it is not supposed to. When this was detected in future data scrapes the chart showed the ability to change the point to yellow to allow users to quickly review the conversation. This showed that the dashboard would have to power to look at a dictionary of key phrases that were deemed questionable, and be able to alert users to critical issues.

The final portion of the project was using CSS and HTML to allow users the ability to dynamically change the data being displayed by the chart, and the ability to zoom in and out of the chart to reduce cluttering of too much data. This was proven successful in the ability to go from changing the X axis of the chart, and by giving users the ability to scroll to change the size and number of objects on the chart.

I was able to implement these three pieces and put them together in a way that would show all user data in one chart, that was customizable, and had the ability to alert users to suspicious data points that they may want to inspect further. All of which can be seen in the diagrams (Fig 4. and Fig 5.) below

Fig. 4. The chart fully zoomed out

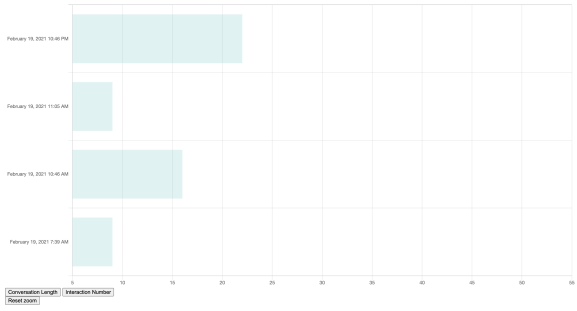

Fig. 5. The chart zoomed in

V. Discussion

The data visualization on the current Alexa website made by Amazon is lacking in ability to clearly show data that may be questionable, and the knowledge of the page is not widespread enough to sufficiently assure that all users can access their own data. This can lead to problems as not guaranteeing that users can protect their own privacies, and can detract from the overall product with users believing it has too much information readily available on them. Amazon's ability to give users the ability to see their data, but in a less than robust design, and without proper alerts to let users know where to find this information allows for more data to be saved. One potential way to help mitigate this issue is to broaden the population that knows of this feature. Amazon has a duty to give users the ability to fully choose their own privacy by deleting data they deem too sensitive to be saved. This may take away from full optimization, but will give users far more safety in regards to their own data.

With the recent emergence of VPAs there are risks that are not fully understood. The Amazon Alexa has the capability of storing every conversation had by the user, that will be used to help tailor a custom user experience, but as seen the ways users can access the data is rather limited, and not well known. This means that most users of VPAs have all of their conversations consistently monitored, and do not know of methods that will help them curate their data so it is up to their own standards. This means that users could be constantly giving personal identifiable information, and have no way of protecting themselves if something nefarious were to be done with this data. This project has taken a look at the ability to give users a quicker way of accessing their data, and quickly seeing what data may be questionable.

Limitations and Future Work - The biggest step that can be taken with this project is implementing language trees to give a better user experience in finding dangerous data. This would be critical in that it would mean that the program could adapt to its users and find what is deemed sensitive information, and from there give a better way of consistently showing users what data may want to be deleted. This would help in the fact that the developers would no longer no need to create their own dictionary of key phrases, and instead constantly refine the process for better checking. This project was limited by necessity and processing power, guaranteeing all data processing is done locally created issues with constantly updating on user preferences.

Another improvement that may help create a better and stronger dashboard is giving the dashboard the ability to hit the Amazon website after the chart has been generated. The major benefit of this would be allowing users to delete data from the dashboard once they have deemed it unsafe to be left stored. This would limit the risk of deleting improper data from storage, and reducing use error with alternating processes.

VI. Conclusion

In this work, we investigated the extent of skill squatting in children's skills and found that Amazon recently added an update to Alexa. This update will ask users if this is the skill they initially intended to invoke only there are skills with similar names. We also looked for skills that may violate COPPA by analyzing the description and reviews of a skill and found an instance of a skill asking for PII. A challenge of this project was the lack of automated testing for Alexa skills which will be beneficial to developers and researchers in the future. As VPAs are becoming more popular and integrated into our daily lives, data privacy and security will become a top priority.

This project led the team to explore different ways of clearly displaying conversational data stored by the Amazon Alexa device. This dashboard will allow users to take all conversations currently stored by Amazon, and quickly check to see where problematic data lives, and check to make sure all conversations stored make sense. We also looked into the ability to predict what data may or may not be sensitive, and created methods to give a clean user experience so they are able to decide how best to view their data. A key issue with development was the necessity to keep all data local, as this limited the ability for true predictive processes at the current time. With more VPAs expected to be on the market, and home across the globe adopting it more frequently it will be crucial to give users a way of curating what data is stored on them, and to easily remove conversation that might be too revelaing.

References

[1] W. Einat, "Data privacy rules are changing. how can marketers keep up?" Feb 2021. [Online]. Available: https://hbr.org/2020/08/data-privacy-rules-are-changing-how-can-marketers-keep-up

[2] D. Y. Huang and N. Feamster, Apr 2019. [Online]. Available: https://iotinspector.org/blog/

[3] N. Malkin, J. Deatrick, A. Tong, P. Wijesekera, S. Egelman, and D. Wagner, "Privacy attitudes of smart speaker users," *Proceedings on Privacy Enhancing Technologies*, vol. 2019, no. 4, p. 250–271, 2019.

[4] J. Ren, D. J. Dubois, D. Choffnes, A. M. Mandalari, R. Kolcun, and H. Haddadi, "Iot information exposure (imc '19)," 2019. [Online]. Available: https://moniotrlab.ccis.neu.edu/imc19/

[5] L. S. Vailshery, "Number of voice assistants in use worldwide 2019-2024," Jan 2021. [Online]. Available: https://www.statista.com/statistics/973815/worldwide-digital-voice-assistant-in-use/