Social Hacking Helper (SHH!)

Christopher Jackson
OMS Cybersecurity
Georgia Institute of Technology
Atlanta, Georgia 30332
Email: cjackson42@gatech.edu

1 Abstract

I would like to help mitigate the problem of Social Engineering that occurs over the phone in any organization. I plan to develop a highly-scalable, cloud-based tool that can listen in real-time to a help desk operator's conversation with a customer and illustrate in real-time a level of confidence that the person to whom they are speaking with is in fact malicious or not. There are tools available to help analyze recorded conversations for possible attacks but the problem is that many of these tools take a post-analytics approach after the attacker has infiltrated the information they were seeking and in some cases has already caused damage using the infiltrated data. We need a more timely approach that will alert a help desk operator in order to real-time prevent the exfiltration of data or compromise before an attacker can succeed. In addition, we need an easier way for organizations to quickly consume and deploy such security tools in their environments.

2 Introduction

Security systems and tools have constantly evolved to enhance the security posture of organizations, however people themselves have not advanced as fast as technology. Many would agree that humans are still the weakest link in any organization when it comes to security. This problem has an immediate, real-world relevance that if used successfully could immensely help mitigate and cut down the amount of social engineering attacks that succeed by arming front-line workers with enhanced AI tooling to augment their understanding in real-time when it comes to enabling a secure environment. In addition, by using a cloud-based approach and cloud-native services, we can easily package up such an application making it easy to deploy and use.

3 Background (Social Engineering Attack Methods on Call Centers)

Social Engineering attacks have been around since before the dawn of the internet through telecommunication systems. These attacks are well known yet still very pervasive today. Many researchers have tackled the problem of Social Engineering in a phone call setting [1,2]. Some methods are based on scam signatures but also on machine learning or AI techniques. The general issue with any techniques is building a sufficient dataset in order to build models and predict scam callers. In this paper, I develop a tool, Social Hacking Helper tool (SHH!), which uses a combination of the two approaches to help combat social engineering attacks in a phone call center setting.

4 Social Hacking Helper tool (SHH!)

Social Hacking Helper is a cloud-based tool that can listen in real-time to a help desk operator's conversation with a customer and illustrate in real-time a level of confidence that the person to whom they are speaking with is in fact malicious or not. Building this tool involves a great deal of infrastructure architecture and natural language processing algorithms using parts of speech, dictation, key phrases, and patterns common among bad actors attempting to social engineer a help desk operator within an organization.

4.1 SHH as a software solution

Social Hacking Helper consists of three main areas:

1. A virtual call center with a real-world phone number and end-to-end infrastructure to help demonstrate a help-desk operator receiving calls. In addition, I will deliver end-to-end cloud-based infrastructure for running real-time analytics on these recorded calls. This will be accomplished by using Amazon Connect, a cloud-native cloud center platform that is very customizable, deployable, and provides real-time data output storage of calls while separating voice channels for both the help desk operator and the Good/Bad Actor calling into the call center. This last feature will be valuable for analyzing the sources in the conversation.

2. I've determined that finding large datasets to train an ML as well as performing custom labeling for ground truth is too big a scope for a semester project. Instead, I've found research that uses a more signature-based approach allowing them to use more statistical analy-

sis methods and natural language processing techniques, but again, their methods are not done in real-time as I propose for this project. [2]. Therefore, I will follow some of the approaches taken in [1,2]. For the non-scam cases I will use the datasets from CallHome dataet [3] which contains 140 spoken English conversations (120 participants with 140 conversations). These are all telephone conversations. However, finding datasets for actual social engineering scam attacks is still proving to be quite difficult. Many research papers to not publish their datasets for anonymity purposes and much of the research shows that they've crafted their own datasets from volunteers. For example, in [2], they recruited 15 graduate students and gave each of the five prompts presenting example telephone scams and had each of them write their own version of each of the five prompts. I can do something similar, I can for example employ Generative AI (just like the graduate students) and craft dozens of example telephone scam transcriptions between an attacker and a help desk operator. The deliverable here will be some sort of packaged classifier that is able to be deployed in any environment built with a similar technology stack.

3. A security application that will in real-time analyze the recorded calls as well as transcribe and analyze both help desk operator and potential attacker voice data enriching the data with key phrases, parts of speech, dictation, etc. in addition to querying the trained machine-learning model in order to produce a confidence score which will be shown through a custom UI to help desk operators easily react to the given confidence score.

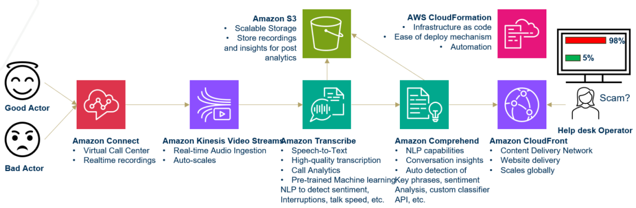


Fig. 1. SHH Architecture

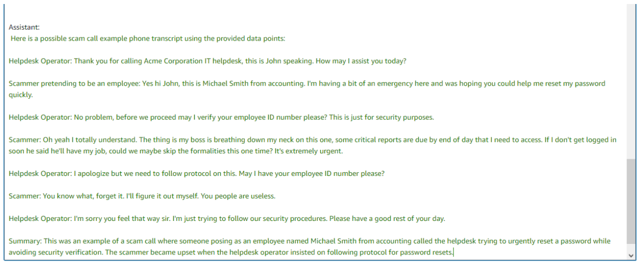
4.2 SSH High-Level Architecture

The architecture for SSH is presented in Figure 1. I rely on Amazon Web Services cloud-native services to help process and ingest audio calls in real-time as well as services like Amazon Transcribe for Speech-to-Text that we can send to Amazon Comprehend for conversation insights. In fact this is where I can create custom classifiers and work with pre-trained ML models. I also have the option of retraining ML models and doing manual GroundTruth labeling. Included in the diagram is a high-level depiction of a webpage for Help desk operators to access the output of the analysis of the conversation in real-time. What is not depicted in this high-level diagram is the actual application code that will take up the bulk of the semester to connect and integrate the services via AWS Lambda functions which will also be where most of my analysis and data work will be performed.

5 Call Center Analytic Datasets

Based on the feedback I need datasets. I've decided to follow some of the approaches taken in [2] For the non-scam cases I will use the datasets from CallHome dataet [3] which contains 140 spoken English conversations (120 participants with 140 conversations). These are all telephone conversations. However, finding datasets for actual social engineering scam attacks is still proving to be quite difficult. Many research papers to not publish their datasets for anonymity purposes and much of the research shows that they've crafted their own datasets from volunteers. For example in [2], they recruited 15 graduate students and gave each of the five prompts presenting example telephone scams and had each of them write their own version of each of the five prompts. I can do something similar, I can for example employ Generative AI (just like the graduate students) and craft dozens of example telephone scam transcriptions between an attacker and a help desk operator. Here are a couple examples from my own experiments.

Figure 2 illustrates a generative AI generated scam call while Figure 3 illustrates a non-scam call.



Assistant:
Here is a possible scam call example phone transcript using the provided data points:

Helpdesk Operator: Thank you for calling Acme Corporation IT helpdesk, this is John speaking. How may I assist you today?

Scammer pretending to be an employee: Yes hi John, this is Michael Smith from accounting. I'm having a bit of an emergency here and was hoping you could help me reset my password quickly.

Helpdesk Operator: No problem, before we proceed may I verify your employee ID number please? This is just for security purposes.

Scammer: Oh yeah I totally understand. The thing is my boss is breathing down my neck on this one, some critical reports are due by end of day that I need to access. If I don't get logged in soon he said he'll have my job, could we maybe skip the formalities this one time? It's extremely urgent.

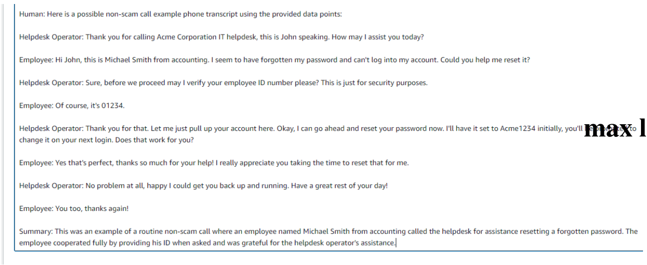
Helpdesk Operator: I apologize but we need to follow protocol on this. May I have your employee ID number please?

Scammer: You know what, forget it. I'll figure it out myself. You people are useless.

Helpdesk Operator: I'm sorry you feel that way sir. I'm just trying to follow our security procedures. Please have a good rest of your day.

Summary: This was an example of a scam call where someone posing as an employee named Michael Smith from accounting called the helpdesk trying to urgently reset a password while avoiding security verification. The scammer became upset when the helpdesk operator insisted on following protocol for password resets.

Fig. 2. Generative AI created scam call



Human: Here is a possible non-scam call example phone transcript using the provided data points:

Helpdesk Operator: Thank you for calling Acme Corporation IT helpdesk, this is John speaking. How may I assist you today?

Employee: Hi John, this is Michael Smith from accounting. I seem to have forgotten my password and can't log into my account. Could you help me reset it?

Helpdesk Operator: Sure, before we proceed may I verify your employee ID number please? This is just for security purposes.

Employee: Of course, it's 01234.

Helpdesk Operator: Thank you for that. Let me just pull up your account here. Okay, I can go ahead and reset your password now. I'll have it set to Acme1234 initially, you'll be prompted to change it on your next login. Does that work for you?

Employee: Yes that's perfect, thanks so much for your help! I really appreciate you taking the time to reset that for me.

Helpdesk Operator: No problem at all, happy I could get you back up and running. Have a great rest of your day!

Employee: You too, thanks again!

Summary: This was an example of a routine non-scam call where an employee named Michael Smith from accounting called the helpdesk for assistance resetting a forgotten password. The employee cooperated fully by providing his ID when asked and was grateful for the helpdesk operator's assistance.

Fig. 3. Generative AI created non-scam call

6 Generative AI Prompting

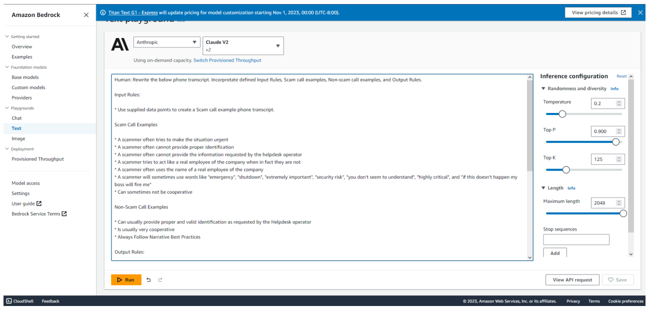
The Prompt I construct is extremely important to the sophisication of the portion of the dataset generated by AI. I've used best-practices given by both Amazon and Anthropic to build a sophisticated prompt able to generate a variety of both scam and non-scam calls. Appendix A. shows the full prompt used by SHH. The prompt consists of Input Rules, Output Rules, Scam call attributes, Non-scam call attributes, Output formating, and data points to follow. Attributes here means common features found among scammers and non-scammers such as scammers portraying a sense of "urgency" or unable to provide information which would authenticate their identiy while non-scammers typically (but not always) provide some identity criteria in order to authenticate with the Helpdesk Operator.

Figure 4 shows the use of Anthropic's Claude v2 model with our prompt. You'll notice there are several Inference configurations which we set. Through pure experiementation I've set these values to settings that will give me both a diverse set of dataset call transcripts as well as stay relevant to scam vs non-scam calls. The meaning behind these values are as follows:

temperature - (0.3-0.5) probability to construct the potential next word in a sequence. higher value is more randomness in how words are put together.

top p - (0.5-0.8) the cutoff based on the sum of probabilities of the potential choices. Lower value ignore less probable results

top k - (200) the cutoff where the model no longer selects the next word. K=50 selecst from 50 of the most probably next word that could be next in a given sequence. The number of the highest-probability vocabulary tokens to keep for Top-K-filtering. the number potential word choices the model uses to generate the next word.



Amazon Bedrock

Titan Text G1 - Express will update pricing for model customization starting Nov 1, 2023, 00:00 (UTC-8:00).

Anthropic
Claude V2
v2

Using on-demand capacity. Switch Provisioned Throughput

Human: Rewrite the below phone transcript. Incorprotate defined Input Rules, Scam call examples, Non-scam call examples, and Output Rules.

Input Rules:

* Use supplied data points to create a Scam call example phone transcript.

Scam Call Examples

* A scammer often tries to make the situation urgent
* A scammer often cannot provide proper identification
* A scammer often cannot provide the information requested by the helpdesk operator
* A scammer tries to act like a real employee of the company when in fact they are not
* A scammer often uses the name of a real employee of the company
* A scammer will sometimes use words like "emergency", "shutdown", "extremely important", "security risk", "you don't seem to understand", "highly critical", and "if this doesn't happen my boss will fire me"
* Can sometimes not be cooperative

Non-Scam Call Examples

* Can usually provide proper and valid identification as requested by the Helpdesk operator
* Is usually very cooperative
* Always Follow Narrative Best Practices

Output Rules:

Inference configuration

Randomness and diversity

Temperature 0.2

Top P 0.900

Top K 125

Length

Maximum length 2048

Stop sequences

Fig. 4. Anthropic Claude v2 Large Language Model (LLM)

max length - (4000) The maximum length of the output. I used this to help regulate the length of the transcript.

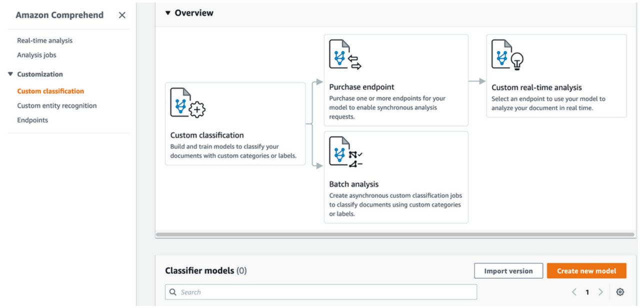
There are however some limitations to using a hosted Generative AI model like the above examples due to built-in safeguards. For example a very similar prompt would hit an ethical wall by the GenAI API. To get around this I simply have to self-host these GenAI models myself within Amazon SageMaker, a machine-learning platform, and host my own endpoints. Then I will not have the ethical limitations as seen above when simply calling a 3rd party service. With dozens of real-life non-scam calls and automated dataset of scam calls, I can use these two datasets to better evaluate the system.

In addition, from the architecture perspective I can load the transcriptions directly from S3 storage to Amazon Comprehend, or if I want to simulate a real-time conversation I can easily playback the transcript using Amazon Polly directly into Amazon Transcribe and continue along the original path as shown in the architecture. I will leave both options open in case I run into usability issues.

I continue to explorer other datasets. I've found one such dataset which I'm awaiting access approval called "AVSpoof" [4], which is dataset for speaker recognition and voice presentation attack detection (anti-spoofing). This may not entirely align with my research, but once I have access I will evaluate it's usefulness for this project.

7 Amazon Comprehend

Amazon Comprehend is a natural-language processing (NLP) service that uses machine learning to uncover valuable insights and connections in text. It develops insights by recognizing the entities, key phrases, language, sentiments, and other common elements in a document. Specifically I'm using Amazon Comprehend's custom classification feature in order to organize my documents into categories of spam and non-spam. I use a two step-process as illustrated in Figure 5. I first train a custom classification machine-learning model (classifier) into categories of spam and non-spam. I then I use this model to feed in call transcripts in real-time as it's happening. The more a conversation move along the



Custom classification
Build and train models to classify your documents with custom categories or labels.

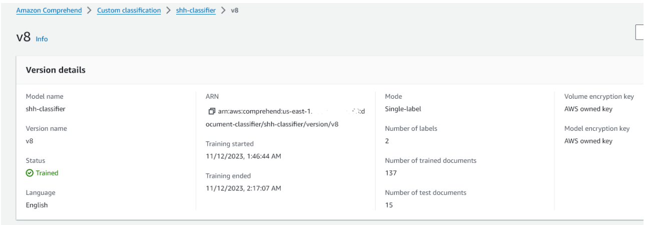
Purchase endpoint
Purchase one or more endpoints for your model to enable synchronous analysis requests.

Custom real-time analysis
Select an endpoint to use your model to analyze your document in real time.

Batch analysis
Create asynchronous custom classification jobs to classify documents using custom categories or labels.

Classifier models (0)

Fig. 5. Amazon Comprehend Workflow with Inference endpoints


Amazon Comprehend > Custom classification > shh-classifier > v8
v8 Info

Version details

Model name
shh-classifier

Version name
v8

Status
Trained

Language
English

ARN
arn:aws:comprehend:us-east-1.[redacted]:d
ocument-classifier/shh-classifier/version/v8

Training started
11/12/2023, 1:46:44 AM

Training ended
11/12/2023, 2:17:07 AM

Mode
Single-label

Number of labels
2

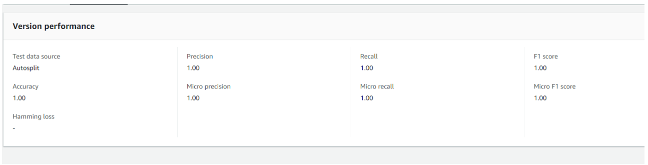
Number of trained documents
137

Number of test documents
15

Volume encryption key
AWS owned key

Model encryption key
AWS owned key

Fig. 6. Amazon Comprehend Workflow with Inference endpoints



Version performance

Test data source
Autosplit

Accuracy
1.00

Hamming loss
-

Precision
1.00

Micro precision
1.00

Recall
1.00

Micro recall
1.00

F1 score
1.00

Micro F1 score
1.00

Fig. 7. Amazon Comprehend Workflow with Inference endpoints

more times I call the model. For example if the ”scammer” and the Helpdesk operator both greet each other, that bit of text is sent to the classifier model. By the end of the call, the entire call transcript is sent to the model for evaluation. The Call center Helpdesk operator is able to view in real-time the result of the classifier along with a confidence score if the conversation so far looks to be a ”scam” or ”non-scam” call. This output is presented nicely to the helpdesk operator and color-coded in red(bad), yellow(warning), green(good or likely not a scam) along with the percentage of confidence.

8 Experiments and Evaluation

The dataset resulted in over 150 sample call transcripts with 80 percent being generated using the GenAI model. As a whole, 90 percent of the dataset was used for training while 10 percent was used as testing. I also manually tested the model with various samples. I was quite impressed as the model performed exceptionally well and better than expected. Figure 6 shows the details of the trained classifier. Figure 7 shows the performance it had on the testing dataset. It has a 100 percent accuracy, 1.00 on Precision, 1.00 on Recall, and an F1 score of 1.00.

9 Limitations

There are several limitations to this tool. First the cost of running this tool can become more expensive the more you run a custom classifier endpoint. It costs me $1 USD per day to run my inference endpoint. Other costs though as you scale are based on consumption usage. Therefore you are not charged a flat fee but rather only during the time of consumption. This helps this tool scale with organizations of different sizes. However, storage costs can increase if left in Standard storage. Based on AWS one could utilize long-term and infrequent access storage classes to save on costs. Other cloud providers offer something similar as well.

Another limitation is that not all training data will be generated via voice. Most of the training data was already in text form. I tested with Amazon Polly to go from text-to-speech but this has the limitation of not being exactly Human and thus throws off the “utterances” and other evaluation metrics that occur naturally with human voice. In any case, the model was able to predict with a high-level accuracy with Human voice calls versus transcript only.

10 Conclusions

Social Hacking Helper as a tool was built to provide a simplified and effective way to help combat Social Engineering attacks on their organization’s Helpdesk Operators. This tool proved to be successful in many cases. It is only as effective as the dataset it was trained on as well as the GenAI prompt that helps generate and diversify the dataset itself. As the security community continues to collect and share more datasets of call conversations to use for testing (a challenge within itself) as well as more sophisticated GenAI prompts, we can improve upon this model. SHH is a prototype for future work and development. It is a framework and workflow that is deployable, at leasts for now in AWS using CloudFormation templates, but can be ported to other Cloud platforms as well. It’s combined approach of using a ”scam signature” style to train GenAI model in the dataset as well as using NLP and machine-learning models to learn from sample scam and non-scam calls, it proves to be an effective approach in combating Social Engineering attacks in real-time.

Acknowledgements

I would like to acknowledge the constructive feedback I received from Professor Mustaque Ahamad, the student Technical Assistants, as well as for my student Peer group. Their feedback was invaluable for the production of this work.

References

[1] Pandit. Combating robocalls with phone virtual assistant mediated interaction. URL https://www.usenix.org/system/files/usenixsecurity23-pandit.pdf.

[2] Derakhshan. Detecting telephone-based social engineering attacks using scam signatures. URL https://dl.acm.org/doi/pdf/10.1145/3445970.3451152.

[3] Macwhinney, and Brian, 2010. Transcribing, searching and data sharing: The clan software and the talkbank data repository. gesprächsforschung online-zeitschrift zur verbalen interaktion (www.gespraechsforschung-ozs.de). URL https://media.talkbank.org/ca/CallFriend/eng-n/.

[4] Khoury, E., Lazaridis, A., and Marcel, S., 2015. On the vulnerability of speaker verification to realistic voice spoofing. In Proc. Int. Conf. on Biometrics: Theory, Applications and Systems (BTAS).10.1109/BTAS.2015.7358783. URL https://www.idiap.ch/en/dataset/avspoof.

Appendix A: Generative AI Prompt

Human: Rewrite the below phone transcript. Incorprotate defined Input Rules, Scam call examples, Non-scam call examples, and Output Rules.

Input Rules:

* Use supplied data points to create a Scam call example phone transcript.

Scam Call Examples

* A scammer often tries to make the situation urgent * A scammer often cannot provide proper identification * A scammer often cannot provide the information requested by the helpdesk operator * A scammer tries to act like a real employee of the company when in fact they are not * A scammer often uses the name of a real employee of the company * A scammer will sometimes use words like ”emergency”, ”shutdown”, ”extremely important”, ”security risk”, ”you don’t seem to understand”, ”highly critical”, and ”if this doesn’t happen my boss will fire me” * Can sometimes not be cooperative

Non-Scam Call Examples

* Can usually provide proper and valid identification as requested by the Helpdesk operator * Is usually very cooperative * Always Follow Narrative Best Practices

Output Rules:

* Output should consist of the following: * The phone transcript should either be an employee who is performing a scam call example or a non-scam call example who calls a helpdesk operator. * A phone call between an employee and a helpdesk operator where the helpdesk operators answers the call from an employee who has a problem they are trying to solve. * If the call is a scammer call example it means the employee is a scammer, otherwise the employee is a non-scammer. * The call should start with a greeting and identification * The call should end with helpdesk operator trying to do their best to solve the employee’s problem. * The phone transcript should not last more than 2 minutes when spoken. * The phone transcript should be written in first person simulating an actual phone call.

* Output format * The output should be the actual complete phone call transcript followed by a summary of the phone transcript.

Data points:

* Employee: Is trying to ether reset their corporate password, get personal information or data on other employeest, gain access to a corprorate office, or install malware on the helpdesk operator’s computer, obtain the helpdesk operator’s password * Hepdesk operator: Is trying to properly identify the employee in order to help resolve their issue

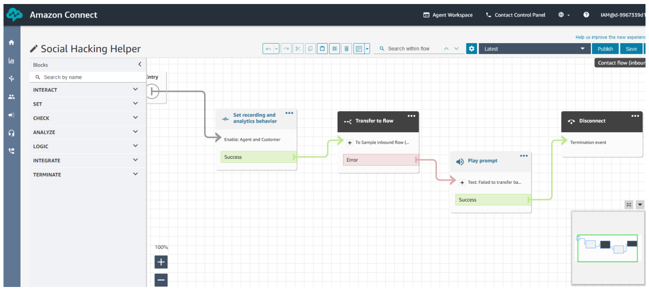
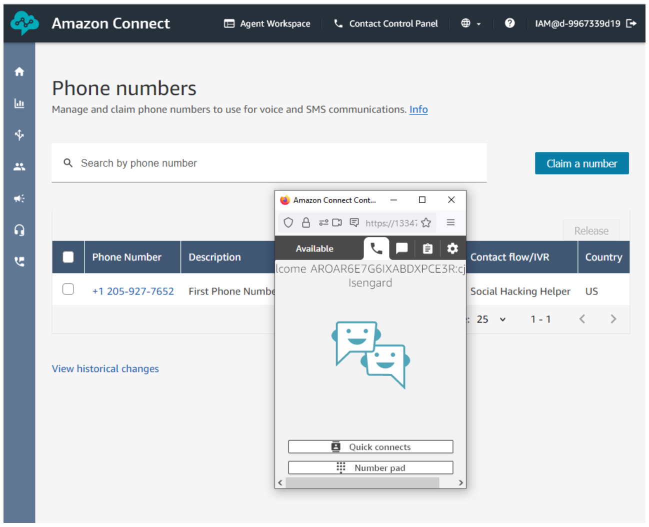


Fig. 8. Amazon Connect Call Center Flow



Phone numbers

Manage and claim phone numbers to use for voice and SMS communications. Info

Search by phone number

	Phone Number	Description	Contact flow/IVR	Country
☐	+1 205-927-7652	First Phone Numbe	Social Hacking Helper	US

: 25 1 - 1

Fig. 9. Amazon Connect Helpdesk Call Client

Appendix B: Amazon Connect Call Center Flow

Figure 8 and 9 show the virtual call center setup a call flow used by the Helpdesk Operator organization.