Secure voice based authentication for mobile devices: Vaulted Voice Verification

R.C. Johnson[a,b], Walter J. Scheirer[a,c] and Terrance E. Boult[a,b]

[a]Securics, Inc, Colorado Springs, CO, USA;
[b]University of Colorado Colorado Springs, CO, USA;
[c]Harvard University, Cambridge, MA, USA;

ABSTRACT

As the use of biometrics becomes more wide-spread, the privacy concerns that stem from the use of biometrics are becoming more apparent. As the usage of mobile devices grows, so does the desire to implement biometric identification into such devices. A large majority of mobile devices being used are mobile phones. While work is being done to implement different types of biometrics into mobile phones, such as photo based biometrics, voice is a more natural choice. The idea of voice as a biometric identifier has been around a long time. One of the major concerns with using voice as an identifier is the instability of voice. We have developed a protocol that addresses those instabilities and preserves privacy. This paper describes a novel protocol that allows a user to authenticate using voice on a mobile/remote device without compromising their privacy. We first discuss the Vaulted Verification protocol, which has recently been introduced in research literature, and then describe its limitations. We then introduce a novel adaptation and extension of the Vaulted Verification protocol to voice, dubbed Vaulted Voice Verification (V^3). Following that we show a performance evaluation and then conclude with a discussion of security and future work.

Keywords: Vaulted Verification, biometrics, voice, security

1. INTRODUCTION

This paper addresses the question "Can voice be used in a remote verification protocol that also preserves privacy?" This paper shows that yes it can. We do this by borrowing techniques from the voice community and integrating them with a recently introduced protocol called Vaulted Verification from the vision community. The Vaulted Verification protocol is a vision based protocol that allows for remote verification in a privacy-preserving way.[1,2] The Vaulted Verification protocol will be discussed further in Section 3.

Biometric security is a growing field. In recent years, the popularity of mobile biometrics has increased tremendously with the proliferation of incorporated sensors into everyday technology, such fingerprint scanners in laptops and face recognition in mobile phones. The increased interest in biometric security has lead to more of a focus being placed on the security of such techniques.[3]

Because the actual biometrics are limited and therefore too easily compromised, biometric security protocols must be based on templates, or models. With this, there is much concern over how to protect these templates. Most of the state-of-the-art template protection schemes can be grouped into one of two categories; feature transformation, or biometric cryptosystems.[3] Schemes from the feature transformation category include things like hashing techniques that take the raw features and use a transformed version of them. Schemes from the cryptosystems include things such as fuzzy vaults which hide the real data among the chaff.

One of the biggest issues in biometric systems is the security of enrollment and verification processes for the biometric data. Once a biometric sample (scan, recording, picture, etc...) has been obtained by the system the information has to be not only saved securely, but the system also has to be able to verify against it, or the resulting saved data, with a second sample in order to verity the validity of the second sample. To solve the security issues with the processes, there are a few requirements that must be met.[4] These requirements are:

Further author information: (Send correspondence to R.C. Johnson)
R.C. Johnson and Terrance E. Boult: E-mail: {rjohnso9 , tboult} at uccs.edu,
Walter J. Scheirer: E-mail: wscheirer at fas.harvard.edu

(a) The Problem: Multiple users trying to access server as Bob.

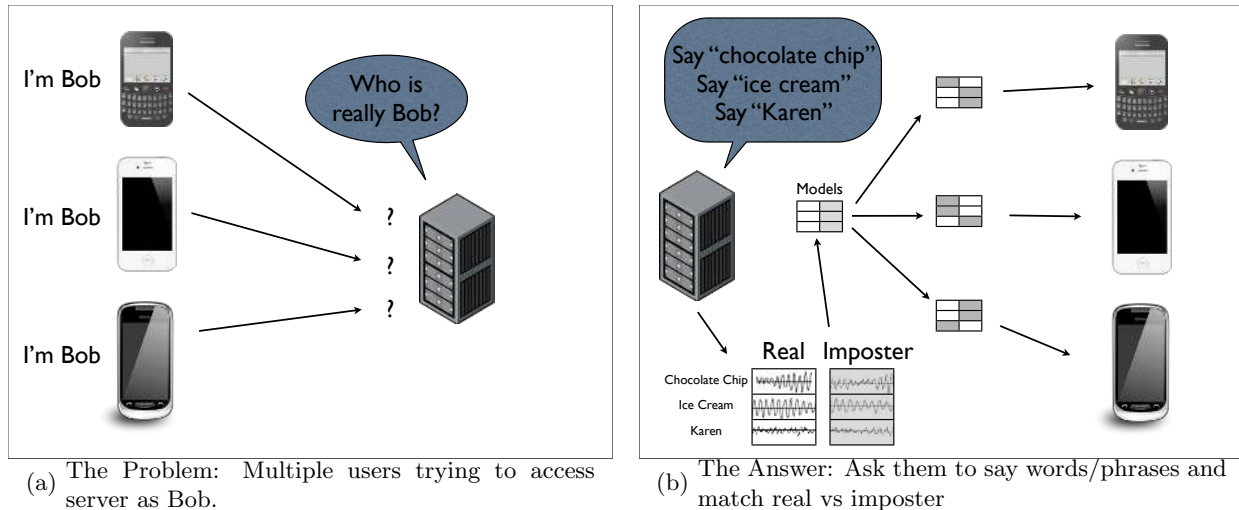(b) The Answer: Ask them to say words/phrases and match real vs imposter

Figure 1. Purpose of Vaulted Voice Verification

1. Ensure that the biometric can not be compromised if the server is compromised and the model is acquired by an attacker.

2. Prevent man-in-the-middle and replay attacks. A "man-in-the-middle" is an attack where someone intercepts the communication from the server to the client and replaces the information on both sides. This is similar to the "replay" attack where an attacker saves the captured information to be used again later.

3. The system must have revocability. If the user needs to invalidate their model current model or something happens where the model is compromised the user should be able to invalidate their current model and issue a new model.

4. Withstand blended substitution attacks. In the blended substitution attack, an attacker combines their template with the stored template so that both the user and the attacker can authenticate using the same template. The system should be able to detect and prevent someone from substituting a users model with their own.

The prior work on the Vaulted Verification technique addresses the previously mentioned requirements. In the current work we are extending this to Vaulted Voice Verification , a novel challenge-response protocol appropriate for voice based verification. The Vaulted Voice Verification protocol is a new challenge-response approach to authentication well suited to mobile devices. The problem Vaulted Voice Verification truly solves is how can this be done in a secure way while also preserving the privacy of the user. One of the advantages of Vaulted Voice Verification is that security can be increased by combing speech content with the challenges. As an example, Figure 1(a) shows multiple people trying to access a system while all claiming to be Bob. Using the novel Vaulted Voice Verification protocol, a simplified version of which is shown in Figure 1(b), the server is able to use the challenge response pairs to ensure the authenticity of the users. This example shows the server challenging the users by having them speak three words/phrases. The server sends pairs of models to the user, with one authentic and one chaff model for each phrase. The user must decide which of each pair is authentic based on their response, 0 if they think the real is first and a 1 otherwise. In Figure 1(b) the top user should respond 100, the middle user should respond 110, and the bottom user should respond 001. The authentic user should have no difficulty choosing the correct bits; impostors have to guess. There are other layers of security, the algorithm is described in greater detail in section 4.

2. RELATED WORK

Many techniques are used in an effort to utilize voice as a biometric identifier. The idea of privacy and security using voice goes many years back as in 2001 when Monrose et al[5] created a system for biometric key generation.

However, the system created by Monrose et al was impractical because of the varying nature of voice. There has been lots of work on voice, as Jain et al[3] and Rathgeb et al[6] discussed in their survey papers which discuss the state of the art. Here, we are only going to discuss the most relevant papers to this work.

In the last couple of years, different groups have begun to focus on using template protection with speech using GMMs. Teoh and Chong[7] used Probabilistic Random Projections (PRP) to protect their speaker template while doing speaker verification. In their research, the template is hidden in a process of random projections in a subspace. The EER obtained in their experiment range between 0% and 8.94% depending on the scenario. Vaulted Voice Verification differs from this work because it is a client-server protocol that can be used on mobile devices, and the template is not stored on the server. Vaulted Voice Verification also improves over the technique discussed by Teoh and Chong in that we use multiple templates instead of a single template, this improves the certainty in the final score.

In this paper, we are using MFCCs and will highlight the similarities and difference between our work and the work of others using similar structures. For example, in the work done by Murty et al[8] the combination of residual phase and mel-frequency cepstral coefficients (MFCCs) effect a 10.5% equal error rate (EER) for speaker verification in a text-independent speaker verification system with no security or protection.

Some of recent work by Inthavisas and Lopresti[9, 10] proposes combining "a speech biometric cryptosystem with a password" . In this work, the biometric template is transformed, mapped, permuted, fuzzy committed, and more. During the transformation stage of their process, a Discrete Fourier Transform (DFT) is applied to the raw audio signal to produce the feature vectors. This is similar to, if not the same as, the process of obtaining MFCCs. The system then performs dynamic time warping to "minimize distance between the reference signal and training utterance." Next, the features are mapped to a binary string. When features that are most stable are found, a string of them, collectively called a "distinguishing descriptor" is used in the next step. To harden their template, they perturb the system by run the mapping function over and over again, each time removing one of the stable features and using this new vector as the key in the dynamic time warping. Until the resulting distinguishing descriptor has less than or equal to half of the feature vector length, this process repeats. At this point, the template gets transformed and permuted, as more thoroughly described in their work,[9] so that the last step is to combine their final binary string with a cryptographic key. In the verification phase, a template, a bunch of thresholds, and the lock data are given to the user. Then, the user's password and a new biometric submission are used to generate a new set of distinguishing descriptors, which then have the correct indexes pulled out to match the lock data. If the user's data matches the lock data, within some tolerance, then the decoding process takes place, and the system authenticates the user.

Using their technique on the MIT mobile device speaker verification corpus they report EER using three different scenarios; neither password nor biometrics compromised, password compromised, biometrics compromised. The EERs reported are 0%, 11.11%, and 0%, respectively for their two layer scheme. This shows two main things; that the technique becomes vulnerable if the password is compromised and that the dataset is not challenging enough for full evaluation of the technique.

3. Vaulted Verification

As with any client-server verification process, the Vaulted Verification process utilizes two main steps: enrollment and verification. Vaulted Voice Verification shares its base concept with the Vaulted Verification protocol, so this section will provide a general overview of the Vaulted Verification process.[1, 2]

3.1 Overview

We first define some basic terms to assist in understanding the Vaulted Verification (V^2) process. C refers to the client software running on the mobile device. U refers to a person using C. S refers to the machine/device to which U is trying to gain access. K_S and K_U refer to the encryption keys of S and U, respectively.

During the enrollment processes of V^2, U gives a biometric sample to the C. C takes that sample and creates feature vectors from the sample. The feature vectors are then sectioned into blocks. C then creates a chaff block for each of the feature vector blocks in such a way that it is indistinguishable from the real feature vector blocks to anyone but a user having the same original features. The chaff blocks are created by randomly taking
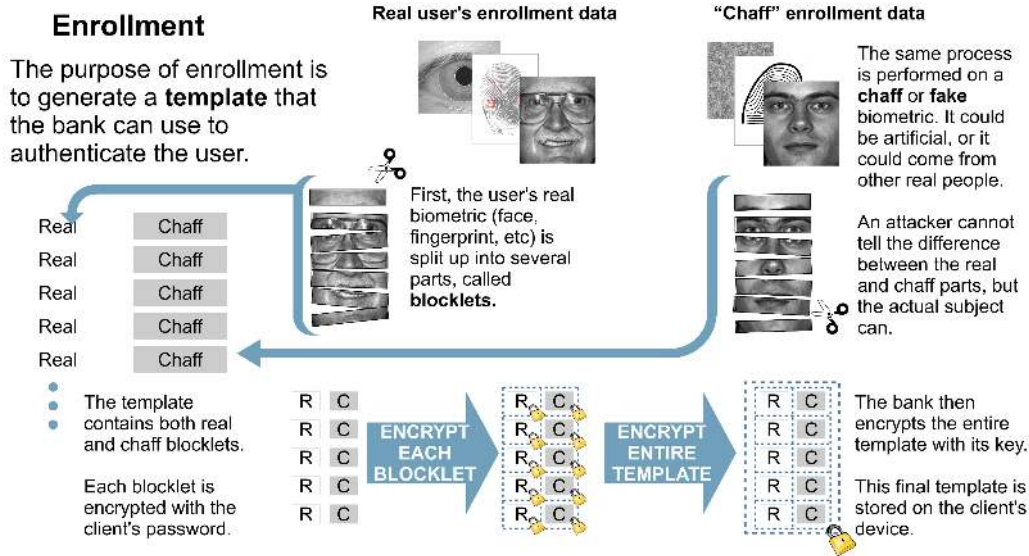
Figure 2. Visual example of Vaulted Verification enrollment.[1]

a corresponding feature vector from a different person.[1] As shown in Figure 2, each block is then encrypted, first using K_U and again using K_S. The blocks are then sent to S in such a way that S knows of the pairs which blocks are real and which blocks are chaff.

As can be seen in Figure 3, when U logs in during the verification process S will send U a locked challenge. For this, S creates a random binary bitstring of n bits such that roughly $n/2$ bits are 1's. S then sends n block pairs to C in an order that depends on the value of the binary bitstring; if the bit is a 0 the real block is sent first, if the bit is a 1 the chaff block is sent first. U then submits an image to C. C will then decrypt and analyze each block pair and respond with a 0 or 1 for each pair depending on which C believes is real and which C believes is chaff.

In this scenario, the blocks stored on S are secure even if S is compromised because S does not have the encryption keys and, therefore, does not know the contents of the blocks. The only information S has is the knowledge of real vs chaff. S does not even concern itself with how the blocks are authenticated, just whether or not the responses are correct.
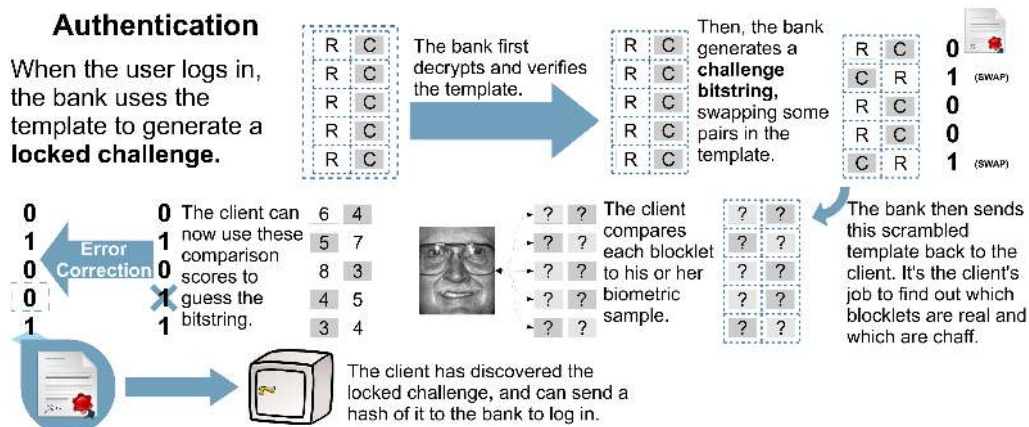

Figure 3. Visual example of Vaulted Verification authentication/verification.[1]

3.2 Limitations

This section describes the limitations of the Vaulted Verification protocol. One of the main limitations of Vaulted Verification is that with the limited amount of information available in face and iris data. Vaulted Verification is not able to vary the data in the challenge-response process. Also, because of the limited amount of data, there are only so many challenge-responses that can be generated. The data limitations of Vaulted Verification that are imposed by face and iris data are the same for both face and iris, so this explanation will refer only to face as shown in FIgures 2 and 3.

As a part of the enrollment, the image is split into blocklets. These blocklets are what get mixed with chaff. Choosing between the real and the chaff blocklets is the basis for vaulted verification. When sectioning the image into blocklets, a large enough section must be chosen so differentiation can happen. If the sections are two small the real and the chaff will blend together, defeating the purpose Vaulted Verification serves. Inversely, the larger the blocklets the smaller the number of challenge-response pairs.

Vaulted Verification will provide security with privacy. The limitation becomes notable when certain pairs are consistently guessed correctly by an attacker, rendering them useless. The more pairs an attacker can consistently guess correctly, the weaker the security of the algorithm becomes. Because of the limited data, there is a limited number of paris that can be generated, which in turn limits the effective security.

4. Vaulted Voice Verification

This section will describe the general procedure of how Vaulted Voice Verification will be utilized in the creation of this novel scheme. Vaulted Voice Verification improves upon Vaulted Verification in multiple ways. Vaulted Voice Verification can expand the available data using different words and phrases. If some models are compromised they can be discarded and other models generated using new words and phrases. An additional way is a further generalization of Vaulted Verification via its application using voice. Another is the enhanced variability and security through the question selection process. With Vaulted Voice Verification S does not know what the questions or the answers are; only that for some encrypted question, there exists a correct response among the encrypted choices. Also, Vaulted Voice Verification will be easy to use and implement on a mobile platform because there is no need for specialized hardware beyond a microphone, which all mobile phones have.

To understand the Vaulted Voice Verification protocol, let us also define the following terms and symbols. P denotes the password. q denotes a single question and Q a set of questions. r denotes a single response and R a set of responses. The symbol $=>$ denotes transmission of some form (speaking to, typing in, sending over the network). The symbol $<=>$ denotes back and forth communication. M denotes a message that is either an input or transmission. The keys, K_U and K_S, are RSA keys.

4.1 Enrollment Process

The first thing that must happen is enrollment. A simplified version of the enrollment process is shown in Figure 4. The steps mentioned in this section refer to the numbered arrows in Figure 4. For layered security, during transmission M will first be encrypted with private K_U and again with public K_S. K_S is obtained when initial contact for enrollment is made from C to U. U needs to give enough information so that they will be able to verify their identity at a later time. The process involves U supplying S with some different information.

The first step, labeled step 1, is for U to give C identifying information, so S can identify U with their data. U tells C their name and possibly other pieces of identification that S will be able to use during the verification process. U also either inputs P or a random P is automatically generated and stored on C. P is used to generate K_U. Next C gives S the identifying enrollment information. This information is also encrypted K_U, so the only thing S knows is an id and a hash.

$$U(id, P) => C \tag{1}$$

In step 2, C asks U to repeat a series of phrases/questions. These phrases are what S will use to challenge C during verification. The questions are designed to get the maximum amount of information out of the fewest
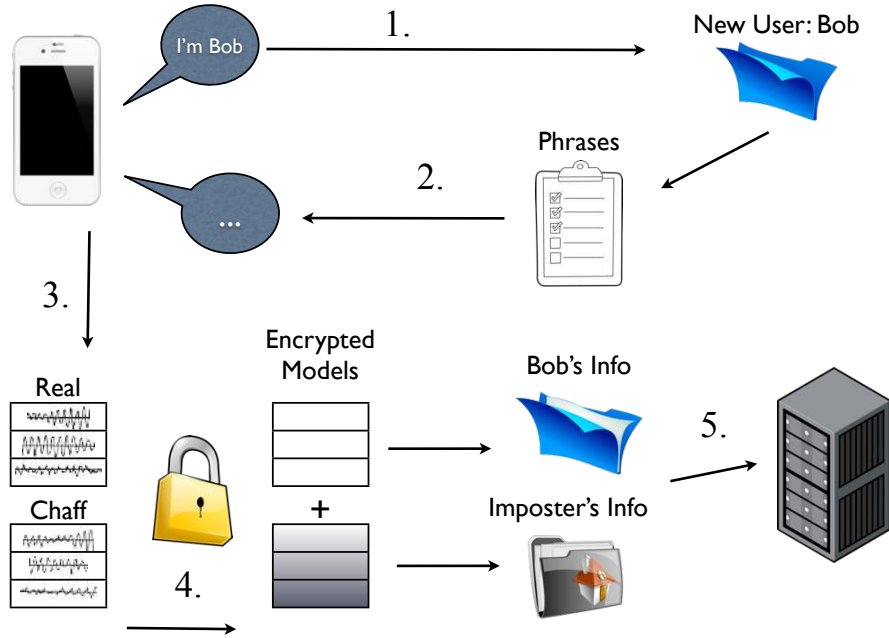
Figure 4. Vaulted Voice Verification - Enrollment Process. 1) Client device gives Server information for new User. 2) Server gives Client set of Questions/phrases for Client to give User. 3) User gives Responses to Client and models are generated and chaff is inserted. 4) Models and chaff are encrypted on Client device using password/key. 5) Encrypted models (now cyphertext) sent to Server for storage.

number of phrases. The goal of the questions is to be able to generate a prescribed number of challenge-response pairs, thus having enough bits to give proper identity security.

$$S(q) => C(q) => U \tag{2}$$

As step 3 illustrates, C will create adapted models from the responses. The models created are GMMs. Each GMM contains a number of distributions, D, and each of those distributions has a number of components, C, for a total number of components of $N = D * C$. How these models are used and compared will be detailed in section 4.2.

The models and transcriptions of the responses are stored together. Each of the model/transcription pairs are encrypted as a block of data. Step 4 shows the models as they are first encrypted with K_U, and then again with K_S. This will ensure that S has no knowledge of what the responses/questions are. The only thing S is aware of is that for some encrypted question, there exists a correct response among the encrypted responses provided.

$$U(r) => C \tag{3}$$

S then receives and stores the challenge/response pairs for future verification, illustrated in step 5.

4.2 Verification Process

The verification process is initiated when a user attempts to gain access to the system. A simplified version of the verification processes is shown in Figure 5. All steps mentioned in this section refer to Figure 5. In this scheme, a user responds to questions asked by C. This happens in steps as follows.
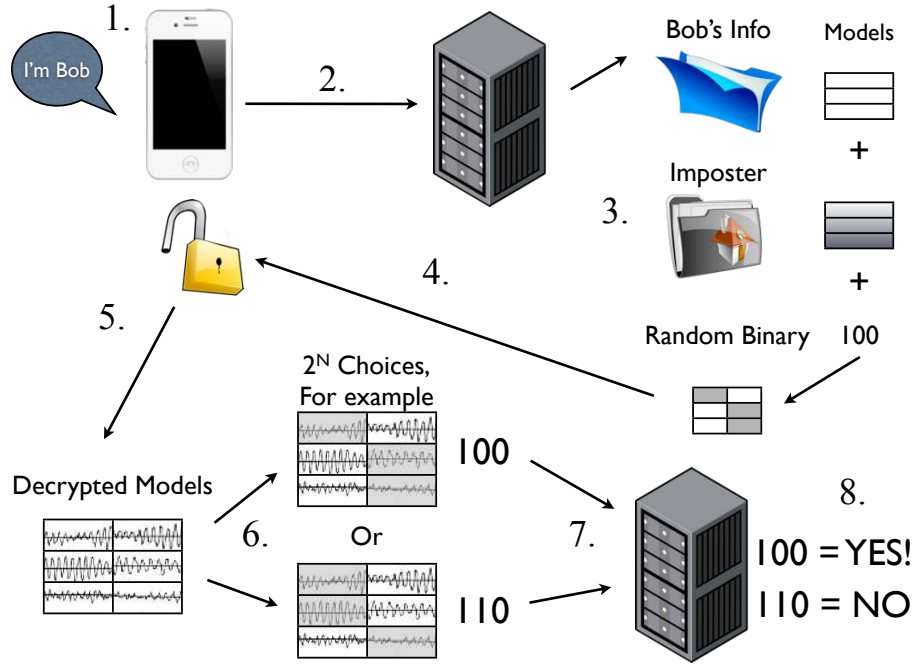
Figure 5. Vaulted Voice Verification - Verification Process. 1) User wants to verify, inputs passphrase. 2) Client sends request to Server. 3) Server retrieves stored models (in cypher text form) and randomizes them according to binary string. 4) Server sends Challenge to Client. 5) Client decrypts using password/key. 6) Client interacts with User to generate response. 7) Client sends response to Server. 8) Server responds with Yes/No.

First, U tells C that they would like to log in and access their data. At this point, C does not know who U is or even who U is claiming to be. U tells C who they are by either entering in their name or speaking their name/id. This is illustrated in step 1.

$$U(id) => C \qquad (4)$$

Then C asks for the initial password/passphrase While this happens, C also sends a request to S for U's information, as illustrated in step 2. C uses K_U, derived from P, to decrypt the transmissions from S. If C does not have the correct K_U, there will be no way to get the rest of the information for the challenge responses, ending the process.

$$U(P) => C, C(id) <=> S \qquad (5)$$

Once C is able to decrypt the information packet, M, from S, C compares the response to the information received from U.

$$K_U(M) = id \ or \ K_U(M) = 0 \qquad (6)$$

If C is able to decrypt M using K_U, the challenge response process begins. In the challenge response process, S will send the Q to C. Q is comprised of reordered pairs of (q, r) and is appended with a nonce. The pairs, as illustrated in step 3, have been reordered based on a random binary string that S generates.

$$S(Q) => C \qquad (7)$$

Step 4 illustrates S sending encrypted Q to C. Q is block encrypted with public K_U before transmission. This ensures that the blocks are different with every transmission; based on the random binary and the nonce. In step 5 C then decrypts and displays/asks Q to U, one at a time, q.

$$C(q) => U \tag{8}$$

U then responds to q. The response depends on the type of question asked. If the question is multiple choice, U would speak the correct answer. If the question is a passage to read, then U simply reads what is asked.

$$U(r) => C \tag{9}$$

C processes the response into a model. A decision is made by comparing the response from U to the options presented from S. Step 6 illustrates an example of two choices that could result from comparing real and imposter/chaff models. When comparing the similarity of two models, the probe, pr, and the gallery, g, either the variance of pr or g will be used in the calculations, shown in equation 10, depending on if the intention is to see how similar pr is to g or how similar g is to pr. Observe that for each gaussian mixture component there exists a mean, μ, and a variance, σ. In equation 10, we are seeing how similar pr is to g, which is to utilize the variance of g. As shown in equation 11, a final score is the summation of the z-scores over the total number of components, N.

$$Z_i = \frac{x_{pr} - \mu_g}{\sigma_g} \tag{10}$$

$$Score = \sum_{i}^{N} (Z_i) \tag{11}$$

Once the models are compared, C will make a decision as to which of the models from S is closest to the model generated from U.

Illustrated in step 7, C will send a binary encoded response back to the server. How this response will be encoded depends on the type of question. If the question has two options, then 0—1 are appropriate. For a multipart question, multiple bits may be required. The process will repeat until stopping condition. The stopping condition will generally be the required number of bits being reached.

After this exchange has finished, C will have generated a large bitstring to send the server. The bitstring will then be evaluated by the server. If the score is above threshold, access will be granted, illustrated in step 8.

$$C(bitstring) => S \tag{12}$$

$$S(yes\ or\ no) => C \tag{13}$$

4.3 Relation to Prior work

Comparing our work to the prior work as discussed in section 2, we can see that Vaulted Voice Verification differs from the other techniques in a few different ways. One of the ways our work is different is that in the enrollment phase; once the features are extracted and the model is generated, the process is over. Also, in our technique the server will not access the raw data in any way and therefore, needs to have no knowledge of its composition. The server has no knowledge of the data and cannot decrypt it. This is a novel and cryptographically stronger approach than simple template hardening and obfuscation.

Because the server has no knowledge of the data and can not decrypt it, even if the server is fully compromised, the user's biometric data will be safe. Even if the user's password were compromised, the approach uses text-dependent models and therefore, independent phrases can be revoked.

5. PERFORMANCE EVALUATION

The dataset selected for the tests is the MIT mobile device speaker verification corpus.[11] This dataset was chosen because it appears in the work of Inthavisas and Lopresti,[9] and is the state of the art benchmark against which to measure results. The speech corpus is comprised on 48 speakers; 22 female and 26 male. Short phrases, names and ice cream flavors, were recorded in 20 minute sessions. Enrollment data was created in two separate sessions. The imposter data was recorded in a separate session. Each person has a dedicated imposter, meaning all imposter files for a user come from one person instead of each file being made from a different person's voice.

In order to perform the tests, the data needed to be separated. The two enrollment sessions were split into a gallery set and a probe set. At random, 60% of the data was designated as gallery and the other 40% as probe. This ensures an appropriate balance between sessions. The imposter data was used in its entirety as the imposter data. In the tests, the data are further separated on a per-phrase basis; this way speech-dependent models could be created on which to run the tests.

The models created from the data are Gaussian Mixture Models, or GMMs. GMMs are used because of their ability to model arbitrary data in a meaningful way. The first step in modeling the voice is building the feature vectors. The feature vectors are created by analyzing the audio spectrum. Different groups[8, 12] have explained in greater detail how feature vectors are created, however, the general idea behind how the audio spectrum is broken up and turned into feature vectors can be explained in a fairly straightforward manner. The audio spectrum is divided into slightly overlapping frequency blocks and then a transform of some type, typically a DFT, is used to transform each of those blocks into cepstrum coefficients. The type of cepstrum coefficients used is referred to as Mel Frequency Cepstrum Coefficients (MFCCs). The MFCCs are what make up the feature vectors. The feature vectors are used to describe a given audio sample in a meaningful and quantifiable way.

Once the feature vectors are created, a model can be generated. The model that is created can model almost anything about the audio sample depending on how the samples are grouped, and how the feature vectors are created. Gaussian distributions have been found to be extremely useful in modeling the created feature vectors. For a deeper understanding of gaussian distributions and why they are useful for modeling based on feature vectors refer to.[13]

The number of distributions chosen is dependent of the amount of data that are being modeled. The larger the number of distributions used, the finer the granularity of the model. If only a small amount of data is being modeled then there will not be enough data to necessitate a larger number of distributions.

When looking into how to compare the different models many options were found that could have been suitable. The option finally chosen was z-scores, as shown in Equation 10. Z-scores, also known as standard scores, were chosen for a number of reasons, the best of which are the simplicity of computation and the relative proximity of the components of the formula to the model data.

Performance is measured in terms of the false accept rate (FAR) and the false reject rate (FRR). FAR measures the percentage of people that are allowed access to the system what they are posing as someone else. FRR measures the percentage of people who are falsely denied access to the system. Where these values meet when plotted against each other is called the equal error rate (EER). It is crucial to mention that the EER in Vaulted Voice Verification only applies to security after the attacker has cracked/compromised the keys (K_U and K_S). Before these keys are compromised, an attacker would get nothing.

In these experiments, we are assuming all keys have been compromised and examining the amount of security Vaulted Voice Verification adds on top of the encryption. This is what Inthavisas and Lopresti[9] calls Scenario II, for which he achieved 11% EER. A test was run to make a direct comparison to the work done by Inthavisas and Lopresti.[9, 10] This test also assumed that all keys had been compromised, as was done in Scenario II of Inthavisas and Lopresti's work. This test did not cover the scenario in which the passwords and keys were not compromised because if the data is not decrypted no valid answer can be supplied by the attacker. This test was implemented by comparing the proximity of g to pr and pr_i for each given phrase for an individual. The pr and pr_i pairs were presented in a random order and g had to decided which it was closest to and respond accordingly. In this test there existed only two choices per phrase, so each response was binary. In this test an EER of 0% was generated. Plotting a 0% EER does not illustrate anything compelling, so it was plotted as a single point
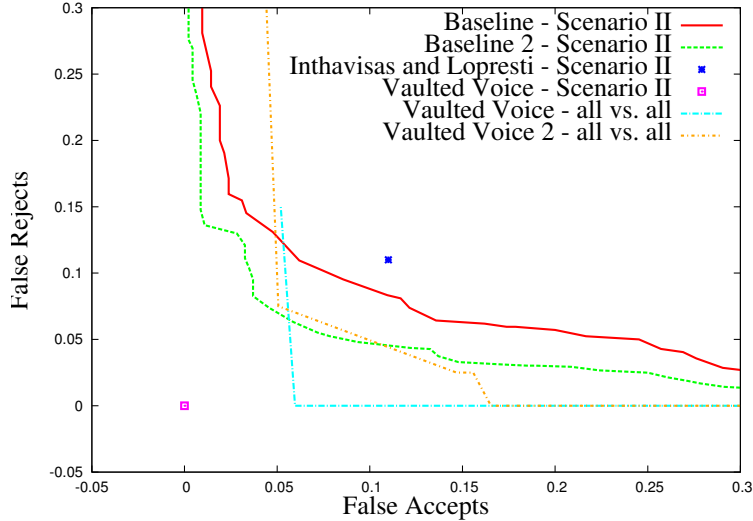
Figure 6. ROC plots from a baseline algorithms on Scenario-II as well as Vaulted Voice Verification on a larger "all-vs-all" scenario with a larger imposter set. Shown as points are Vaulted Voice Verification and Inthavisas and Lopresti's work on Scenario II, with an ERR of 0% and 11%,[9] respectively.

on Figure 6. It is necessary to note, however, that this test directly compares to the test in which Inthavisas and Lopresti reported 11%, but the ERR of 0% is unrealistic because of the small number of impostors.

To provide a stronger experimental validation, we implemented a new baseline algorithm, which is not privacy/security enhanced, and then expanded the testing to a larger test set. Figure 6 shows the ROC curves for these test results. There are results from the two baseline tests and results from the two Vaulted Voice Verification expanded tests. The baseline tests were created by comparing the scores between pr and g in the same manner as Vaulted Voice Verification . However, in the baseline tests only raw scores were recorded, removing the inherent pairwise thresholding that is added by Vaulted Voice Verification .

To generate the data for the first baseline curve, each person in the gallery was tested against pr and their dedicated imposter, pr_i. In these tests, the variance of g is used. This gives an approximate equal error rate of 8%. A second baseline was also generated (labeled Baseline 2). This baseline was created to show a similar separation of data exists when using the variance of pr as opposed to using the variance of g. The EER of this baseline test is 6%. Both of these results show an improvement over the state of the art benchmark of 11%. We note, however, the baseline has no privacy enhancements.

Vaulted Voice Verification performed significantly better than the baseline because Vaulted Voice Verification has an inherent pairwise thresholding step which allows it to make every decision on a case by case bases. For demonstration purposes, we give the following scenario of two phrases. In this scenario, let us assume that a score can range from 0 to 10. For the first phrase, ph_0, g generated a score of 5 for pr_0 and a score of 3 for pr_{0i}. For the second phrase, ph_1, g generated a score of 9 for pr_1 and 6 for pr_{1i}. According to the baseline scores, no clear threshold exists that would separate probes and their impostors. However, because Vaulted Voice Verification thresholds on a case by case bases, it is able to identify and discern between the probes and impostors for both the first and second phrases.

As previously stated, the size of the dataset limits its ability to fully estimate the EER of the Vaulted Voice Verification protocol. To better estimate the EER of the Vaulted Voice Verification protocol, a second expanded set of tests were performed. These tests are labeled in Figure 6 as all vs all. In the first all vs all test, labeled Voice Verification -all vs all, we tested all people in the gallery against all pr and pr_i pairs. By doing so, we have expanded the negative matches to more thoroughly examine the false match rate of the algorithm. In keeping with the tests performed to generate a baseline, a second all vs all test was performed. This test, labeled Voice Verification 2 - all vs all, was done using the variance of the probe and imposter against the gallery data. As Figure 6 shows, these two tests have an EER of approximately 6%.

6. SECURITY ANALYSIS

How is Vaulted Voice Verification able to solve the security issues mentioned in the Introduction? As with most secure biometric systems, this protocol contains many layers of security. As mentioned previously, the base security of this protocol is similar to the work done by Wilber, et al.[1,2]

After enrollment, the client never sends out non-encrypted data, the matching is done on the client after decryption with the user supplied password. From enrollment to verification it is assumed that the communication is happening over a secure encryption protocol, SSL or TLS for example, and an attacker cannot ease drop on the communication. If the encryption is maintained, the system is secure, so we consider various levels of compromise.

If an attack is somehow able to break the encryption on the communication and try to impersonate S and stage a man in the middle attack, the attacker would not be able to gain any additional information about the biometric to give them access. This is because even if the attacker is able to see the data from C to S, the data is still encrypted with K_U, as illustrated in step 4 of Figure 4.

If an attacker attempted a man in the middle attack by recording the encrypted and randomly ordered data pairs and responses, this would also prove pointless. This is because Q is randomly ordered, has a nonce added, and then is block-encrypted with K_S for every session, making the transmitted blocks different for every session. Thus any reordering would fail to decrypt and no information is gained by examining it since the recorded information would be different for the next session. With this, there is, effectively, no man in the middle.

If the device was lost, it provides little information as it just has an encrypted block of data that the client cannot decode. If an attacker were to obtain both the server's private key and the user's device, the attacker might be able to authenticate on that particular server. This is because they would be able to decrypt the models and, through trial and error, return the correct response to the challenges via examining the differences when models are swapped. However, because this is a phrase based scheme, this would not allow an attacker access the user's data on a different server. The reasons are two fold. The first is because the models are encrypted using the user's private key. Without being able to access the models the attacker would not be able to discern any information about it. The second is that without recordings of the correct phrases, the attacker would not have the necessary information to discern between real and chaff data. If the device is lost, the enrolled model can and should be revoked changing the encryption keys and phrases.

The worse case scenario for this protocol would be for the attacker to have access to K_S and K_U. The attacker would then be able to authenticate as U on S, but would still not know which of the N blocks are real and which are chaff. On average, an attacker with all the keys must still make 2^N guesses to correctly identify the biometrics. At 56 blocks, that is 2^{56} attempts; not too difficult for computers today. If the number of blocks were increased to 256, with current computational models, this is sufficiently secure. And again, the user can just revoke and reissue with new phrases and keys.

The lists used in the MIT dataset[11] had one dedicated imposter for each phrase. Each phrase in the data set was repeated 6 to 10 times per person. There were between 8 and 12 phrases spoken per person. The tests performed were using per-phrase comparisons and where, therefore, limited to 8 to 12 bits. Though there could exist statistical differences in the models generated from a given user vs. the general models, if chaff is chosen correctly this should not be an issue. With this, we are assuming independent probabilities for each bit. That gives a probability of between 2^{-8} and 2^{-12} for an attacker with no foreknowledge, using a random chance brute force attack, to gain access. Even with this, the error rate works out to between 0.39% and 0.02%, which is better than 11% reported by Inthavisas and Lopresti[9] under the same conditions, scenario II. Our results are an improvement over straight biometrics, but we want better security.

To increase the security, we would need to increase the number of bits. One of the ways to do this would be to use multiple impostors instead of the one dedicated imposter. This would increase the number of bits based on the number of options. For example, if there were 3 impostors per question instead of 1, there would be 4 options from which to chose. The probability for each question would be 2^{-2}, effectively doubling the number of bits per question. For the same 12 phrases, there would then be a probability of 2^{-24} that an attacker could authenticate as U and gain access.

Again, it is necessary to note that this security analysis of Vaulted Voice Verification is speaking in terms of how much security is added on top of encryption. Vaulted Voice Verification would be providing K-bits of

"identity verification" security in addition to the N-bits of security offered by encryption. Since the equal error rate provides a lower bound on identification accuracy, because of the associated biometric dictionary attack, the identity security offered by Vaulted Voice Verification is already much better than the prior state of the art.

7. CONCLUSION AND FUTURE WORK

To conclude, with this work we have developed a verification protocol that addresses the instabilities inherent in voice in a way that still preserves privacy. We described a novel protocol that allows a user to authenticate using voice on a mobile/remote device without compromising their privacy. We showed that this protocol, dubbed Vaulted Voice Verification (or V^3), is able to achieve a 0% equal error rate in on the same tests where Inthavisas and Lopresti[9] achieved 11%. The new protocol achieves 6% ERR on larger scale test.

In future work, we will extend the challenge response protocol to further demonstrate the abilities of this protocol with voice. We will look into expanding the challenge response protocol, improving security and robustness, by examining the types of questions than can be asked.

One of the ways this will be accomplished is by mixing the text-dependent work done so far with text-independent modeling; allowing us to extend beyond basic words and phrases. Adding speech detection to the system also increases the number of bits. If, during the verification process, the system is also able to determine what the spoken phrase is, more bits would be added. For example, the work done by Just[14] showed that it could take as many as 2^{48} answers to correctly guess a word. If asked an open-ended question, the number of bits would be increased significantly. In addition to questions, we could also shuffle the order of phrases to increase the number of bits.

Another aspect we will look into is using location based questions and answers to decide which questions are asked. This will also help to defend against replay attacks. We also look to extend the security requirements of biometric systems by examining the ability and implications of detecting verification attempts made under duress.

ACKNOWLEDGMENTS

REFERENCES

[1] Wilber, M. J. and Boult, T. E., "Secure remote matching with privacy: Scrambled support vector vaulted verification," *Workshops on the Applications of Computer Vision (WACV)* (2012).

[2] Michael J. Wilber, Walter J. Scheirer, T. E. B., "Privv: Privae remote iris-authentication with vaulted verification," *Computer Vision and Pattern Recognition (CVPR)* (2012).

[3] Anil K. Jain, K. N. and Nagar, A., "Biometric template security," *EURASIP Journal on Advances in Signal Processing* **Special Issue on Biometrics** (2008).

[4] Scherier, W. and Boult, T., "Cracking fuzzy vaults and biometric encryption," *In Proc. of the 2007 Biometrics Symposium, held in conjunction with the Biometrics Consortium Conference (BCC 2007)* (2007).

[5] Monrose, F., Reiter, M. K., Li, Q., and Wetzel, S., "Cryptographic key generation from voice," *2001 IEEE Symposium on Security and Privacy* (May 2001).

[6] Rathgeb, C. and Uhl, A., "A survey on biometric cryptosystems and cancelable biometrics," *EURASIP Journal on Information Security 2011* (2011).

[7] Andrew Beng Jin Teoh, L.-Y. C., "Secure speech template protection in speaker verification system," *Speech Communication* **52**(150-163) (2010).

[8] Murty, K. S. R. and Yegnanarayana, "Combining evidence from residual phase and mfcc features for speaker recognition," *IEEE Signal Processing Letters* **13** (January 2006).

[9] Inthavisas, K. and Lopresti, D., "Speech cryptographic key regeneration based on password," *Proc. Int. Joint Conf. on Biometrics (IJCB 2011)* (October 2011).

[10] Inthavisas, K. and Lopresti, D., "Secure speech biometric templates for user authentication," *The institution of Engineering and Technology Biometrics* (February 2012).

[11] Ram H. Woo, A. P. and Hazen, T. J., "The mit mobile device speaker verification corpus: Data collection and preliminary experiments," *IEEE Odyssey - The Speaker and Language Recognition Workshop* (2006).

[12] F. Zheng, G. Z. and Song, Z., "Comparison of different implementations of mfcc," *Journal of Computer Science andTechnology* **16**, 582–589 (2001).

[13] R. Rajeswara Rao, V. Kamakshi Prasad, A. N., "Performance evaluation of statistical approaches for text-independent speaker recognition using source feature," *InterJRI Computer Science and Networking* **2** (August 2010).

[14] Just, M. and Aspinall, D., "Personal choice and challenge questions: A security and usability assessment," *Proceedings of the 5th Symposium on Usable Privacy and Security, 8* (2009).