A Secure e-Shopping Using Voice Ordering

Saroj Kumar Panigrahy, Debasish Jena, and Sanjay Kumar Jena

Department of Computer Science & Engineering,
National Institute of Technology Rourkela, 769 008, Odisha, India
skp.nitrkl@gmail.com, debasishjena@hotmail.com, skjena@nitrkl.ac.in

Abstract. Internet connectivity can offer enormous advantages, however security needs to be a major consideration when planning an Internet connection. There are significant security risks associated with the Internet that often are not obvious to new (and existing) users. In particular, intruder activities as well as vulnerabilities that could assist intruder activity are widespread. Intruder activity is difficult to predict and at times can be difficult to discover and correct. Many organizations already have lost productive time and money in dealing with intruder activity; some organizations have had their reputations suffer as a result of intruder activity at their sites being publicized. This paper proposes a secure method of e-shopping using Pocket PCs through voice ordering.

Keywords: e-shopping, security, speech recognition, RSA, .Net.

1 Introduction

With the start of design of website, the first and foremost thing the webmaster is worried is about its security. But astonishingly most of the sites, around 90% of the mass, have such a poor security system, that they can be hacked even by a novice hacker, for this reason only security analysts evolved. The first thing that takes fancy of a hacker or a security analyst is open ports. The more open ports are there in a web server, the more prone is that server to a security breach. The next thing that comes into mind is its mailing system. The kind of system used in a site lets hacker to find possible flaws and exploit them such as spamming, spoofing and the like. The next thing is to exploit the hardware flaws, like routers, gateways etc. This is a serious issue that a webmaster should look into.

Recent developments in ASP.NET where the computer uses one of the senses of human beings, the speech sense, due to which, we can make the computer speak and also a conversation can take place between a user and the computer. Another development is the Mobile development or Call management controls. This is highly useful for our Shopping management System. This is an advantage for houses where the internet connection is not available. So the user can send message from any mobile to the mobile which is placed in the Shopping Mall. The user can ask for any information she needs. This gives the complete information of the requested message to the user.

V.V Das et al. (Eds.): BAIP 2010, CCIS 70, pp. 233–237, 2010.

The rest of the paper is organized as follows. Section 2 describes the design and implementation of the proposed method. A brief description of mobile services is explained in Section 3. Finally, the Section 4 describes the concluding remarks.

2 Design and Implementation

Our paper is mainly focused on secure e-shopping. The security that we have implemented is the password encryption and credit card validation. Another special feature that have been implemented in our secure e-shopping is voice ordering components. Each of these security criteria are explained below in greater details.

2.1 Credit Card Validation

The main aim of validating credit card numbers is that no unauthorized user accesses others card and all its substitutes involved with it. This unauthorized access can lead to loss of money and even huge theft using others credit card number. A typical e-business processing system is shown in Fig. 1. As a part of this web form, we wanted to include support to check that users had entered a card number, expiration date etc., and then wanted to extend it further to include support for checking that the card number was valid before issuing a request to the payment gateway's server. This is the result, a drop-in replacement for any of the other validation controls. The card system is implemented using the Luhn's Formula. The working of Luhn's formula together with its usage in creating a valid credit card number is given below [1].

1. ***Double the value of alternating digits*:** The first step is to double each of the alternating digits in the number. But the trick is to start with the second digit from the right and work backwards. Say we have a credit card number 1234 5678 1234 5670. We'll start with the second rightmost number 7, double it, and then do the same for every other digit as follows.
   1234 5678 1234 5670

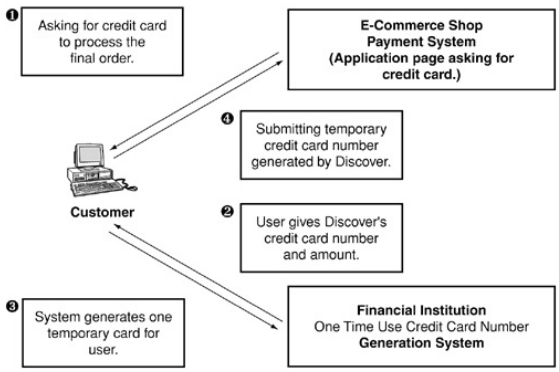

Fig. 1. A typical e-business processing system

This will give us the following values.
$7 \times 2 = 14$, $5 \times 2 = 10$, $3 \times 2 = 6$, ..., etc.

2. ***Add the separate digits of all the products*:** Now we will separate the digits of all the products, and come up with a final sum.
$(1 + 4) + (1 + 0) + 6 + 2 + (1 + 4) + (1 + 0) + 6 + 2 = 28$
3. ***Add the unaffected digits*:** Now we'll go back to the original number and add all the digits that we didn't double. We'll still start from the right, but this time we'll start from the rightmost number.
1234 5678 1234 5670
$0 + 6 + 4 + 2 + 8 + 6 + 4 + 2 = 32$
4. ***Add the results and divide by 10*:** Finally, we'll add both the results and divide the answer by 10.
$28 + 32 = 60$
60 is evenly divided by 10, so the credit card number is well formed and ready for further processing.

The design of our credit card validation first checks the credit card validity using the Luhn's Formula. It also takes into account the valid duration of the card. The amount to be credited is also taken into account. After this the details are submitted to the server which checks it using the Luhn's formula.

2.2 Password Encryption

For our password encryption we have taken use of the public key algorithm, where two keys are used for both encryption and decryption. Here we make use of the public key encryption algorithm called as RSA algorithm [2].

The password encryption has been designed as follows. It entitles the user to enter her name for checking the validity of the user by comparing the user name from the database. The encryption process takes place at the back end.

2.3 Voice Ordering Components

Voice can be used in personal communication systems [3]. In this voice ordering, any customer who logs into the website will be guided by a voice which tells them various aspects of shopping in the Shopping Mall. This voice also guides through the entire process of buying things, the various models that are available in each and every item. This voice will also tell whether an item is available or not. The Fig. 2 illustrates the process of recognizing the voice.

The voice that we speak through the microphone is captured. Then a sound wave is generated and the sound card which is available converts that particular signal into a digital signal, which again is converted into words inside the speech recognition engine. The resulting output from the speech-aware application is the output. The of design of speech recognition is shown in Fig. 3. The different buttons on the right of the figure explains the various buttons involved in generating a correct request. The semantic map button is used to compare the signal with the corresponding request. The other buttons are used in converting the words into the electrifying signal, which is finally converted to the requested word.

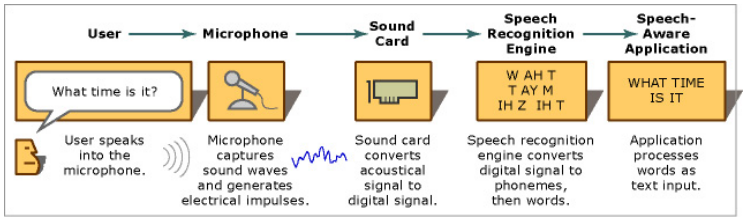

Fig. 2. Process of voice recognition

Fig. 3. Design of speech recognition system

3 Mobile Services

In this scenario, client is the Microsoft Pocket Internet Explorer with the Speech Add-in installed. ASP.NET speech-enabled Web application pages reside on the Web server, along with the application grammars, and a configuration file containing the URL to the Speech Server that performs speech processing [4, 5].

When the user enters a URL on Pocket PC, the Web server opens the application's default *.aspx* page. The Web server also sends the URL pointing to Speech Server. The page that the Web server sends contains HTML, SALT, and JScript. When the user taps a speech-enabled HTML element and talks, Pocket PC sends the audio to Speech Server. Along with the compressed audio, Pocket PC sends either an inline recognition grammar or a pointer to the location of an externally-stored recognition grammar that is bound to that speech-enabled element. If the recognition grammar is an inline grammar, Speech Server loads the grammar and performs speech recognition. If the grammar is an externally-stored grammar, Speech Server first downloads a copy of the grammar, loads the grammar, and then performs speech recognition.

After the recognition finishes, Speech Server sends Semantic Markup Language (SML) output to the Pocket PC along with audio for prompts if the application dialogue flow requires the application to play a prompt. The Pocket PC client parses the SML output, and populates the speech-enabled HTML element with the semantic value to which it is bound, and plays any prompts that Speech Server sends. The Fig. 4 shows the process of interfacing between a computer and mobile.

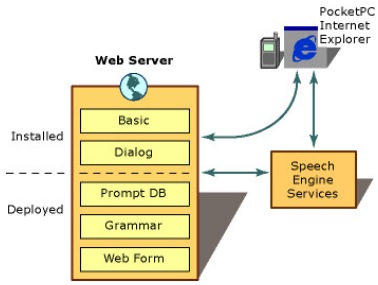

Fig. 4. The mobile and computer interfacing

4 Conclusion

Speech recognition is the latest kind in the market which will play a very important role in the future. We actually run a windows service which will scan for speech. The voice is scanned using microphone. If the voice is below a certain threshold or if a sudden voice is found, then the voice is discarded without compromising for a valid command. Since we have speaker detection, we can fix the commands for a specific speaker.

The above system can be further extended to mobile phones so that they can be accessed over a WAP enabled phone. This reduces the stress of the customer who doesn't have any access to internet. Accessing through mobile phones also enables the customer to get the same response as by accessing through internet.

References

1. Childs, L.N.: A Concrete Introduction to Higher Algebra. Springer, Heidelberg (2008)
2. Stallings, W.: Cryptography and Network Security: Principles and Practices, 3rd edn. Pearson Education, London (2003)
3. Burstein, A., Stolzle, A., Brodersen, R.: Using speech recognition in a personal communications system. In: IEEE International Conference on Communications, ICC 1992, Conference record, SUPERCOMM/ICC 1992, Discovering a New World of Communications, June 1992, vol. 3, pp. 1717–1721 (1992)
4. Evjen, B., Beres, J., et al.: Visual Basic. Net Programming Bible. Wiley, Chichester (2002)
5. Holzner, S.: Visual Basic.net Programming Little Black Book. Wiley Dreamtech, Chichester (2005)