Using Human Interactive Proofs to Secure Human-Machine Interactions via Untrusted Intermediaries

Chris J. Mitchell

Information Security Group, Royal Holloway, University of London,
Egham, Surrey TW20 0EX, UK
c.mitchell@rhul.ac.uk
http://www.isg.rhul.ac.uk/~cjm

Abstract. This paper explores ways in which Human Interactive Proofs (HIPs), *i.e.* problems which are easy for humans to solve but are intractable for computers, can be used to improve the security of human-machine interactions. The particular focus of this paper is the case where these interactions take place via an untrusted intermediary device, and where the use of HIPs can be used to establish a secure channel between the human and target machine. A number of application scenarios of this general type are considered, and in each case the possible use of HIPs to improve interaction security is explored.

1 Introduction

There are many situations in which a user is forced to conduct transactions employing a user interface which he/she does not entirely trust. Examples include the following.

- Many smart card applications require the user to insert his/her card into a terminal which does not belong to the user. In such a case, the user must trust the terminal both to correctly convey his/her instructions to the card, and not to make a record of any confidential information that is transferred. For example, if the card is being used to authorise a transaction, then the user must trust the terminal to instruct the card to authorise a transaction of the correct value.
- When a user employs a 'public' PC (*e.g.* in an Internet café or an airport terminal) the user has no guarantees that the terminal is not manipulating his/her instructions. For example, if such a terminal is used for a purchase, the user has no way of verifying that the terminal is correctly ordering the goods required.

This is a well-known and important practical problem, but one which is also difficult to address. Previous work has considered various aspects of this problem — see, for example, [1,2,3].

B. Christianson et al. (Eds.): Security Protocols 2006, LNCS 5087, pp. 164–170, 2009.

In this paper we consider a new approach to this problems, based on so-called Human Interactive Proofs (HIPs). In particular, we consider a number of different scenarios in which humans and machines (*e.g.* smart cards, remote PCs, etc.) communicate via untrusted intermediaries. In each case we also consider how HIPs might be be used to improve security for such an interaction.

2 The Main Idea

2.1 Human Interactive Proofs

HIPs, as discussed, for example, by Dhamija and Tygar [4] 'allow a computer to distinguish a specific class of humans over a network'. We are interested here in the simplest case, *i.e.* where humans are distinguished from computers. Again, as specified in [4], 'to do this, the computer presents a challenge that must be easy for ...humans to pass, yet hard for non-members [*i.e.* computers] to pass'.

Probably the most common example of a HIP in use today is where numbers or letters are displayed in a distorted fashion so that only humans can read them. However, many other techniques have been proposed, *e.g.* to choose a pair of images showing the same person (from amongst a larger set), or to count objects of a particular type in a scene [5].

Many applications for HIPs have been proposed. Probably the best known is the use by Yahoo (`www.yahoo.com`) to restrict access to free email accounts. Here a user must solve a HIP before obtaining such an account; this prevents automatic harvesting of email addresses for use by spammers.

2.2 Establishment of a Secure Channel

We now consider the possibility that HIPs could be used to establish a secure channel between a human user and a trusted device when all communications take place via an untrusted intermediary. Of course, this will not be a perfect solution to all the security problems, but it might significantly reduce threats in some situations.

It is not hard to see that, in some sense, a HIP can provide a low bandwidth communications channel between the computer and the human, which no other *computer* can intercept. In the case of a HIP based on distorted characters, the computer could display a message which the human can read but no other computer can understand. As a result, the HIP provides a confidentiality-protected but unauthenticated communications channel from the computer to the human.

We can also use HIPs to provide a confidential but unauthenticated reverse channel, *i.e.* a communications channel from the human to the computer. One such approach would involve the computer displaying a sequence of distorted messages (unreadable to another computer), of which the user chooses one (or more). This could, for example, and as discussed below, enable a user to enter a secret password or PIN into a computer in a fashion unobservable to a computer, even if can monitor all the data transfers. More generally, for secure password entry, the computer could ask a password-related question (in some distorted form) and receive an answer via the user selection of one of a series of distorted images.

The use of HIPs therefore allows the establishment of a two-way confidential channel between the user and a computer, even when communications pass via an untrusted intermediary. Some examples of possible uses of such a HIP-protected communications channel are sketched in the following sections, although these ideas need further analysis. There may also be further scenarios in which these ideas have value.

Before proceeding we also note two fundamental limitations of the communications channel established using a HIP.

– Firstly, the channel is confidentiality-protected but not authenticated. This means that there is the risk of man-in-the-middle attacks. Overcoming such attacks probably means that techniques of the type we have just described may need to be combined with other techniques for human authentication of remote machines. Of course, in principle it is not difficult to transform a confidential channel into an authenticated channel, given the two parties possess a shared secret. However, the problem here is that the user does not necessarily have any means to perform cryptographic computations.
– Secondly, if a human interceptor is present, then the channel is no longer confidentiality-protected. That is, the technique only offers protection against automated attacks.

3 Scenario 1: Smart Card PIN Entry

When a smart card is inserted into a terminal for some purpose, it is often necessary to insert a PIN to authorise a transaction (this proves to the card that the cardholder is present). If the terminal is untrusted by the cardholder, as is often the case in practice, then there is a risk that the PIN will be immediately and automatically compromised. To try to reduce this risk, the card could use a HIP to prompt for the PIN, *e.g.* by displaying a distorted numeral by each button on the terminal, and inviting the user to type the button corresponding to the first PIN digit, then the second digit, and so on.

Of course, this would require the card to provide the distorted pictures to the terminal, which would then display them. The terminal would then send back to the card the identifiers for the images selected by the user. If the terminal displays the images as requested by the card, the terminal will not learn the PIN, *i.e.* this use of a HIP allows PIN entry via an untrusted terminal to take place in such a way that the PIN cannot be automatically captured. However, if a human monitors the PIN entry process, either in real time *or subsequently*, then the PIN is revealed. This nevertheless makes it more difficult for malicious entities to collect large numbers of user PINs, since the process cannot be completely automated.

Interestingly, the above described process does enable a malicious terminal to learn the PIN if it fails to follow the protocol correctly. That is, if it displays its own images (for which it knows the corresponding digits), instead of the ones provided by the smart card, then it can immediately and automatically learn the correct PIN. However, such an attack would not enable the terminal to enter the PIN it has learnt into the card (at least, unless a human was involved). Hence, the transaction would not be completed. This issue is, of course, a direct corollary of the fact that the HIP-based communications channel is not authenticated.

These latter observations raise the possibility that security might be further improved by incorporating the use of the visual authentication ideas of Dhamija and Tygar [6]. That is, the images produced by the card might be personalised to the user, and hence the user will detect when the terminal displays its own images to try to capture the user PIN.

However, even if this latter approach is followed, the PIN will still be revealed if a human monitors the procedure, either at the time of the transaction or later. That is, the main benefit of the process is in preventing automatic collection of card PINs. It would be interesting to see if the process could be further enhanced to offer a greater degree of security. Possible extensions to and generalisations of the above process include using the same process to protect the transfer of PINs (or passwords) to a remote application via an untrusted terminal. This is discussed further in Scenario 3.

4 Scenario 2: Smart Card Transaction Authorisation

As mentioned in the previous section, smart cards are often used in terminals that are not trusted by the cardholder. In particular, the cardholder may use the card to authorise a transaction, and the card will authorise the value of the transaction on behalf of the cardholder (*e.g.* by signing an authorisation message containing this value). The risk here is that a malicious terminal could change the value of the transaction, since the cardholder has no way of knowing exactly what message the card is authorising.

In a similar way to that described in the previous scenario, the card could ask the cardholder to enter the transaction value by prompting using distorted pictures of digits. That is, the terminal would not know how to enter the correct (or any specific incorrect) transaction value into the card. The card could then display a distorted image of the total transaction value for the cardholder to verify. Finally, once this interaction has completed, the card will provide the authorised message to the terminal. As in the case described in the previous scenario, the security of the process might be improved by using HIP images tailored to a particular user, so that the cardholder will be able to distinguish between images presented by a fraudulent terminal and those presented by the card. That is, prior shared secrets can be used to provide a measure of authentication for the channel.

Of course, as in all cases considered here, if a malicious human can monitor and interfere with card-cardholder communications at the time of the transaction, then the use of a HIP does not help. However, unlike the previous scenario, if a human only has access to a transaction record after the event, then there are no obvious attack strategies. That is, the use of a HIP in this scenario appears to offer greater benefits than in the previous case.

5 Scenario 3: Using an Untrusted PC for a Remote Login

The third scenario we consider here involves the use of an untrusted terminal, *e.g.* a PC in an Internet café, for remote login, *e.g.* to a server operated by the user's employer. This is a common practical scenario.

Suppose, moreover, that a user employs a hardware password-generating token when performing remote logins. Such tokens, that generate 'one-time passwords' valid for only a short time period, are in wide use, and are designed to reduce the risks of sending fixed passwords via intrusted intermediaries. However, if the terminal is fraudulent, then it could potentially copy the one-time password, and use this to login other terminals to the remote computer (as long as the logins occur within a short space of time).

The use of HIPs to secure the communications between user and remote computer might help alleviate this problem. The same strategy could be used as that described in Scenario 1, *i.e.* where the user is invited to enter the one-time password via images displayed on the terminal.

Moreover, given that the password has only a short lifetime, the possibility that it could be learnt by a human monitoring a recorded transaction later, is no longer a significant threat. That is, the security situation is improved because of the short-term nature of the secrets transferred across the channel.

6 Scenario 4: Using an Untrusted PC for E-commerce

In an electronic commerce transaction made using an untrusted terminal, the user is typically required to enter account details to complete the transaction. These details would typically constitute not only the user PAN (Personal Account Number), but also the three-digit security code designed to reduce the risk of re-use of stolen account details. (To reduce the risk that these three-digit codes are themselves stolen, merchant servers are prohibited from storing them).

This type of transaction therefore poses a significant security risk to the secrecy of the cardholder account information. As in Scenarios 1 and 3, HIPs might be used to protect this end user information against automated capture. The analysis would appear to be very similar to that in Scenario 1.

7 Scenario 5: Context Establishment

Our final scenario is a little different. It may be possible to use HIPs to increase the security level of a communications channel used for initial security context establishment. One scenario in which this idea might be useful is 'near-zero-configuration' of personal devices, *e.g.* in the home. This relates, of course, to Stajano and Anderson's notion of imprinting of devices (see, for example, [7]).

Currently, protocols such as Bluetooth involve a pairing process, which, as currently specified, is rather insecure. (More secure alternatives exist, but have yet to be implemented in Bluetooth — see, for example, ISO/IEC 9798-6 [8], or section 10.7 of [9]). It would be interesting if HIPs could be used to enhance the security of such initial exchanges. Of course, the use of a HIP would not help against an active human interceptor, but might help against the more likely threat of an eavesdropper capable of automated interference with a protocol, but not capable of breaking the HIP.

The scenario of use envisaged is where a naive user has to initialise a device, but does not have the expertise (or the patience) to engage in a sophisticated security initialisation procedure. Moreover, if the devices being 'paired' lack a user interface, then the more secure processes described in ISO/IEC 9798-6 are difficult if not impossible to use. We therefore consider the case where the devices being paired make use of untrusted third party devices which do possess the necessary sophisticated user interface.

More specifically suppose that a PIN-based procedure is to be used to secure device pairing, where the PINs are entered into the devices being paired via third party (untrusted) devices. In such a case, if no additional security measures are taken, the third party devices could subvert the pairing process, almost regardless of how the pairing protocol actually works. The use of a HIP, *e.g.* where PIN entry involves selection of distorted images of characters, might enable this pairing process to be made secure, at least against machine-only adversaries.

If the pairing process involves a Diffie-Hellman key exchange authenticated using PINs, then it may be possible to design the scheme so that a human adversary involved only after the exchange cannot break the established key. That is, the security context established between the two devices will remain sound. However, the details of such a system remain to be worked out in detail, and will, of course depend on the details of the operational scenario.

8 Conclusions

We have conducted an initial investigation of a variety of scenarios to see how HIPs might be used to improve the security of human-machine interactions via untrusted intermediaries. The preliminary conclusion would appear to be that, whilst HIPs can offer some benefits in all scenarios, the greatest benefits occur in scenarios where the interaction involves the completion of a transaction (*e.g.* a purchase of goods, or the establishment of a security context), rather than merely the transfer of a password or PIN, unless the password or PIN is only of short term interest.

This is because the use of a HIP does not prevent a human learning any secrets transferred in a monitored exchange either at the time of the exchange or later. However, if a process is completed during the interaction, the only way in which this can be attacked is if the malicious human monitor is actively interfering at the time of the exchange.

Note that it is sometimes much easier for a fraudulent individual to monitor interactions after they have occurred, rather than at the time of the interaction. Apart from the obvious convenience issues for the attacker, in some cases it may only be possible to examine interactions after the event, *e.g.* when key-loggers are used.

We would also point out that the scenarios introduced here have not been analysed in very great detail, and that there may be other more effective approaches for the use of HIPs. Given the significance of the underlying problem, this therefore appears to be an area meriting further research.

Finally, the nature of the secure channel which can be provided using a HIP would also appear to merit further investigation. In particular, one might reasonably ask which types of HIP offer the possibility of the highest bandwidth channels between human and computer.

Acknowledgements

The author would like to gratefully acknowledge a number of helpful comments made by participants in the 2006 Security Protocols Workshop — these have significantly improved the paper.

References

1. Berta, I.Z., Buttyan, L., Vajda, I.: Mitigating the untrusted terminal problem using conditional signatures. In: Proceedings of ITCC 2004, The International Conference on Information Technology: Coding and Computing, Las Vegas, NV, USA, April 2004, pp. 12–16. IEEE, Los Alamitos (2004)
2. Gobioff, H., Smith, S., Tygar, J.D., Yee, B.: Smart cards in hostile environments. In: Proceedings of the Second USENIX Workshop on Electronic Commerce, Oakland, California, November 1996. USENIX Association, Berkeley (1996)
3. Stabell-Kulo, T., Arild, R., Myrvang, P.H.: Providing authentication to messages signed with a smart card in hostile environments. In: Proceedings of the USENIX Workshop on Smartcard Technology, Chicago, Illinois, USA, May 10–11. USENIX Association, Berkeley (1999)
4. Dhamija, R., Tygar, J.D.: Phish and HIPs: Human Interactive Proofs to detect phishing attacks. In: Baird, H.S., Lopresti, D.P. (eds.) HIP 2005. LNCS, vol. 3517, pp. 127–141. Springer, Heidelberg (2005)
5. Lopresti, D.: Leveraging the CAPTCHA problem. In: Baird, H.S., Lopresti, D.P. (eds.) HIP 2005. LNCS, vol. 3517, pp. 97–110. Springer, Heidelberg (2005)
6. Dhamija, R., Tygar, J.D.: The battle against phishing: Dynamic security skins. In: Symposium On Usable Privacy and Security (SOUPS) 2005, Pittsburgh, PA, USA, July 6-8, pp. 77–88. ACM Press, New York (2005)
7. Stajano, F.: Security for Ubiquitous Computing. John Wiley and Sons Ltd., Chichester (2002)
8. International Organization for Standardization Genève, Switzerland: ISO/IEC 9798–6: 2005, Information Technology — Security techniques — Entity authentication — Part 6: Mechanisms using manual data transfer (2005)
9. Dent, A.W., Mitchell, C.J.: User's Guide to Cryptography and Standards. Artech House, Boston (2005)