An Interlock Protocol Critique

Physics and Computer Science, Wilfrid Laurier University

Josh Barber

CP614: Applied Cryptography

Dr. Angèle Foley

February 28th, 2023

During the introduction of computing telecommunications in the 1970's, security was at best "lagging well behind other areas of communications technology" according to Whitfield and Martin. With the computers then, using network protocols to communicate with other business partners, has since then revolutionized the practice of cryptography. Primitive cryptography relied solely on all party members having a private key to decipher cipher text on arrival. This private key was handed out to all parties in privacy, so the key wouldn't be easily leaked to someone unauthorized to use it. This was actually the case for centuries, and this practice was sufficient. However, with the implementation of computer telecommunications over the network, this methodology of exchanging the private key in person is too burdensome. As we see now, the growth of telecommunications over the internet has exponentially grown since the 1970's, and cryptography has come a long way since as we evidently don't exchange our private keys to individuals we want to talk to on social media.

In 1976, a paper cited as "New Directions in Cryptography" by Whitefield Diffie and Martin E. Hellman, brought forth upcoming propositions by other cryptographers to help mitigate the security risks for what we know today as the internet. At this time, there was two directions being proposed; sticking to old conventions and try to keep the key as private as possible, or introduce a new tactic of having not only a private key, but also a public key for each individual party, with the public key to be accessed over insecure communications. Choosing the latter proposition, would go against the old practice of cryptography; hence, in retrospect, the revolution of cryptography. This however was met with skepticism at the time, which of course is understandable, because it is exploring a new methodology that needs to make sure messages are secure. Just having the word public in itself, personally doesn't seem secure. Whitefield and Martin point out that eavesdropping on the communication exchange has problems of its own, which at the time only had partial solutions.

Another major issue with the public key exchange, was that there is no way of proving authenticity of communication. There is no validity of a contract between two public parties making an exchange of information.

Rivest, Shamir and Adleman in 1977 introduce the RSA algorithm which uses a one-way trapdoor from the single user's public key to their private key. In effect, this algorithm helps prove authenticity of the user. The algorithm draws techniques from elementary number theory and isn't infeasible to do, as the cost of performing such algorithm is relatively inexpensive compared to other propositions. Hope for a public key standard came into fruition with the introduction of the RSA algorithm, as at the time it had proven secure. Although, the authenticity of a public key may have been rectified, there was still the unfortunate lingering issue of eavesdropping. Riverst and Shamir propose a solution to mitigate it. They introduce the interlock protocol in 1984.

The interlock protocol as described by Rivest and Shamir is a protocol to frustrate eavesdropper attacks against two parties that use an anonymous key exchange protocol to secure their conversation. Say there exists party *A*, party *B* and the eavesdropper party *C*. Party *C* wants to just monitor the messages, *MA* and *MB*, and observe the public keys *KA* and *KB* of the respective parties *A* and *B*. Consider that communication between *A* and *B* to be unauthenticated and that the public key exchange protocols between the two parties can be transparently monitored by *C*. Once *C* has full access between the transmissions of communications between *A* and *B*, it is then theoretically impossible to foil the transmissions, as *C* can easily mimic *B* or *A* in the communications. The interlock protocol suggests a five step process that will manipulate how the data is transferred over through communications to help prevent *C* from taking over the communications as previously described.

After *A* and *B* have exchanged their public keys, they exchange a pair of data blocks, *MA* and *MB* as follows:

- *A* encrypts *MA* under *KB* but sends *B* only the first half of the bits of the resulting ciphertext $E_{KB}(MA)$.
- *B* encrypts *MB* under *KA* and sends *A* the first half of $E_{KA}(MB)$.
- *A* sends *B* the second half $E_{KB}(MA)$.
- *B* sends *A* the second half of $E_{KA}(MB)$.
- *A* and *B* concatenate the two halves of $E_{KA}(MB)$ and $E_{KB}(MA)$, respectively, and use their secret decryption keys to read the messages.

This protocol in theory would prevent *C* from eavesdropping, as when *C* intercepts an instance of the transmission between *A* to *B*, *C* will only have half of $E_{KC}(MA)$, and this won't be enough to discover anything from *MA*. *C* is now in a predicament to send *B* something to keep the communication alive. If *C* sends *B* the same half ciphertext he received from *A, B* will later try to decrypt it with the wrong key. Thus *C* is forced to create *MA'* (which probably has nothing to do with *MA*) to send to *B* making the transaction non-transparent and *C* will receive garbled text. *C* would likewise have to repeat the process for the *MB* that is sent to *A.* By the time *C* discovers the true values of *MA* and *MB*, it would be two late to change *MA'* and *MB'* because of step 3 and step 4.

After creating the RSA algorithm, Riverst, Shamir and Adleman have been considered geniuses. So having another proposed solution, the interlock protocol, by Riverst and Shamir, it may have had many cryptographers naive in believing that this algorithm is the ultimate solution for eavesdropping. In 1993, in the "An Attack on the Interlock Protocol When Used for Authentication" paper by Steven M. Bellovin and Michael Merrit, it outlined that the interlock protocol actually isn't safe at all. That in fact, an attacker can capture messages with simple means.

Bellovin and Merrit propose that in theory, no matter what, the eavesdropper *C* can imitate *A* or *B* at any given time as *C* can modify messages, delete messages or even generate totally new messages. In this case, what is preventing *C* from imitating *B* in any scenario where *A* wants to send a message to *B*. *C* could imitate *B* and be able to intercept the message, convincing *A*, that *C* is *B*. In this instance, *C* just has to iterate over the interlock protocol steps with *A* to complete the message and then at that point, *C* has the message to interact with *B*. Although, this is assuming *C* can crack *A's* encryption.

$$\begin{array}{lll} \boxed{A} & & \boxed{C} \qquad\qquad \boxed{B} \\ (E_{A,C'}(MA))\langle 1\rangle & \longrightarrow & \\ & \longleftarrow & (E_{A,C'}(MA))\langle 1\rangle \\ (E_{A,C'}(MA))\langle 2\rangle & \longrightarrow & \end{array}$$

At this point, the connection from *A* to *C* can be dropped. Now *C* can combine the two halves of *A*, decrypt it, then send to *B*:

$$\begin{array}{lll} \boxed{A} & \qquad\qquad \boxed{C} & \qquad\qquad \boxed{B} \\ & (E_{C,B}(MA))\langle 1\rangle & \longrightarrow & \\ & & \longleftarrow & (E_{C,B}(MB))\langle 1\rangle \\ & (E_{C,B}(MA))\langle 2\rangle & \longrightarrow & \\ & & \longleftarrow & (E_{C,B}(MB))\langle 2\rangle . \end{array}$$

It is unfortunate to see the interlock protocol so easily foiled. Bellovin and Merrit note, that instead of considering a specific attack in designing protocols, it may be more fruitful to consider classes of attacks. They also note that no protocol should be accepted purely on the basis of defeating a single attack. Although the man-in-the-middle attack (eavesdropping) could still be easily performed against the interlock protocol, the attack is still a lingering issue even in modern days.

Since the interlock protocol, a new standard protocol in 1995 was introduced called the secure sockets layer protocol, or SSL for short. It was poorly designed for security standards with many vulnerabilities such as man-in-the-middle attacks, and several others. The SSL protocol has had many reworks, much of which has been depreciated. The SSL protocol in 2013, was still very much vulnerable for even top tech companies such as Google. A report by CNET, in 2013, suggests that Edward Snowden released a document confirming that the NSA performed a man-in-the-middle attack on Google. Where the NSA mimicked Google with using fake security certificates that were easy foiled by SSL cryptographic protocol, which is designed to verify the authenticity of websites and ensure secure Net communications. It is thought the NSA was collecting data whilst it was mimicking Google. With TLS being the new predecessor, and SSL depreciating, there seems to be an ongoing pattern. Security measures will always need updating as new attacks continue to exploit newer protocols. Whether it be through use of powerful brute force or through tactical means, it will always be an ongoing issue.

References

Bellovin, S. M., & Merritt, M. (1994). An attack on the interlock protocol when used for authentication. *IEEE Transactions on Information Theory*, *40*(1), 273–275. https://doi.org/10.1109/18.272497

Diffie, W., & Hellman, M. (1976). New Directions in cryptography. *IEEE Transactions on Information Theory*, *22*(6), 644–654. https://doi.org/10.1109/tit.1976.1055638

Moyer, E. (2013, September 12). *NSA disguised itself as Google to spy, say reports*. CNET. Retrieved February 28, 2023, from https://www.cnet.com/tech/tech-industry/nsa-disguised-itself-as-google-to-spy-say-reports/

Rivest, R. L., & Shamir, A. (1984). How to expose an eavesdropper. *Communications of the ACM*, *27*(4), 393–394. https://doi.org/10.1145/358027.358053