Negotiation Transparency in Configurable Protocols

A Case Study on the TLS Protocol and the Forward Secrecy Property

Eman Salem Alashwali, CDT15

Negotiation, always perceived as a critical phase in politics and business protocols, is just as important in communication security protocols. In the latter, the term "negotiation" usually refers to the process of exchanging security-related parameters between the communicating parties (e.g. client and server) in order to reach a mutual agreement on an optimal set of parameters that are supported by both communicating parties. These sensitive parameters include the protocol version, and the set of algorithms (ciphersuite) that will be used for the key-exchange, encryption, and hash, to secure subsequent messages of the protocol.

Such a negotiation phase is commonly used in protocols that support multiple versions and multiple algorithms, and are widely deployed on various types of platforms that vary in their capabilities such as personal computers and embedded (IoT) devices. The Transport Layer Security (TLS) and The Secure SHell (SSH) protocols are two notable examples of such widely used protocols.

Experience shows that the negotiation of security parameters is an attractive phase for downgrade attacks, where an active man-in-the-middle attacker interferes with the exchanged messages by the communicating parties, leading them to agree on a mode weaker than they support and prefer. This allows the attacker to perform subsequent attacks that would not have been possible in the strong mode.

It has become clear that ensuring the integrity (i.e. the messages have not been tampered with) and authenticity (i.e. the messages are coming from the intended party) of the exchanged parameters is of paramount importance in the negotiation phase, in order to prevent downgrade attacks.

While the literature has looked at negotiation integrity and authenticity in the active man-in-the-middle attacker model, we look at the problem from a new perspective: we consider transparency, as a result of a novel attacker model that we propose, which we call the "discriminatory" adversarial model. To the best of our knowledge, transparency and discrimination in security protocols negotiation have not been discussed in the existing literature. We are the first to observe and write about them [1][2][3].

In our research, we made an observation pertaining to parameters negotiation in security protocols. That is, certain client-server negotiation models, which we call "server-dominant", result in an imbalanced power between the communicating parties, the client and server. To illustrate, as shown in Figure 1, in the TLS protocol case, the protocol performs the parameters negotiation as follows: the client proposes a set of parameters such as the protocol versions and ciphersuites, ordered by preference, to the server. The server selects one of them and imposes its choice

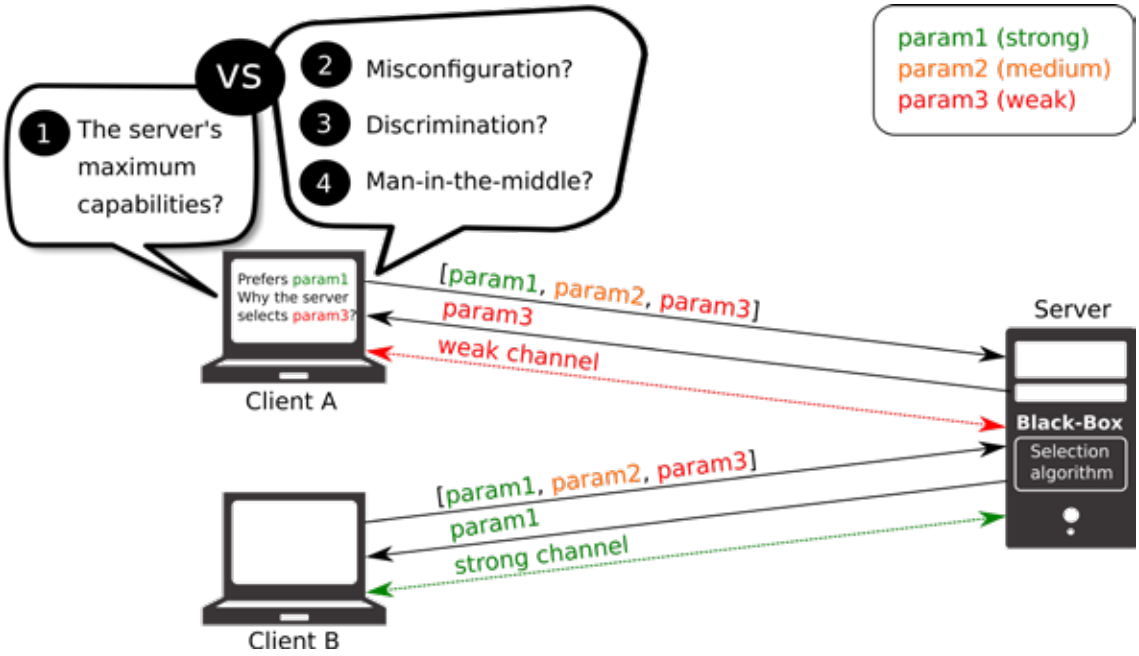

Figure 1: Illustration of our newly introduced discriminatory adversarial model in parameters negotiation in security protocols with server-dominant negotiation model such as the TLS protocol. The term "Param" denotes parameter.

to the client. The client does not necessarily receive its most preferred choice. This can be due to several reasons, such as: server's lack of support for the client's most preferred parameter, server's misconfiguration, server's bad implementation, or a man-in-the-middle attacker that tampered with the messages. However, it can also be due to the server's discrimination against its clients for a powerful third party's advantage (e.g. government intelligence) with minimum liabilities related to the server's involvement. This is a realistic assumption. In fact, it is inspired by past real-life events such as "export-grade" cryptography, a US (deprecated) law, which used to enforce weak cryptography to products (including software) exported outside the US, in addition to Edward Snowden's allegations about the "PRISM" program, for mass surveillance in collaboration with giant cloud service providers.

In the server-dominant negotiation model, the client does not have the means of verifying the server's choice, i.e. justifying the server's decision if it is not optimal, or against the client's preference order. While the server's parameters selection algorithm is known in the protocol specifications, the selection algorithm implementation along with the server's actual supported parameters, are a black-box from the client's perspective.

There is currently no way for the client to verify that the server has behaved correctly, and its selected parameters are optimal. *Figure 1* illustrates our observation which applies to the TLS protocol.

To prove the realism of our model, and most importantly, that our proposed adversarial model can go unnoticed in most mainstream clients today such as web browsers, we consider the case of the TLS protocol and the Forward Secrecy (FS) property. We conduct an empirical analysis on over 10M TLS server addresses, including top domains, random domains, and random IPv4 addresses.

FS is a highly desirable property nowadays, which guarantees that a compromise in the secrecy of the server's long-term key does not compromised the secrecy of past session keys. Therefore, if a passive adversary has been collecting traffic today, the adversary can not decrypt past traffic if the server's private-key is compromised at some point in the future. Some key-exchange algorithms such as ECDHE provide this property, while other key-exchange algorithms such as RSA do not provide it. Experience has shown that it is possible for servers' long-term private-keys to become compromised. For example, RSA long-term private-keys have been compromised through prime factorisation, due to advancement in computing power, or due to low entropy during keys generation. Furthermore, long-term private-keys can be compromised through implementation bugs such as in the Heartbleed bug, through social engineering, or other attacks. While the latest version of TLS, TLS 1.3, mandates FS by design, FS is not mandated in pre-TLS 1.3 versions, which are still widely used by most mainstream TLS clients and servers today, and still support non-FS algorithms. Pre-TLS 1.3 versions (mainly TLS 1.2) may continue to be used for decades to come.

Our empirical study aims to answer the following question: *Do servers that select non-FS key-exchange support a FS one?* That is, are there servers that choose a weaker key-exchange algorithm while they are capable of choosing a stronger one?

To this end, we developed a TLS client that mimics a Chrome browser's proposed versions and ciphersuites, but we implement a heuristic procedure. That is, when the server selects a non-FS key-exchange as a result of our client's default proposal, the client immediately repeats the client's offer to the same server, but with a new set of parameters that proposes FS-only algorithms. This allows us to test if the server is indeed incapable of FS key-exchange algorithms or not.

Our results show that 5.37% of top domains, 7.51% of random domains, and 26.16% of random IPs do not select FS key-exchange algorithms. Surprisingly, 39.20% of the top domains, 24.40% of the random domains, and 14.46% of the random IPs that do not select FS, nevertheless do support FS.

We have studied the case of TLS and FS. However, the discriminatory adversarial model and transparency as a requirement in protocol negotiation models can be generalized to any negotiation of any parameters in security protocols with a semi-trusted party who can gain advantage from discrimination.

This article provided a summary of some of our novel insights and contributions in the area of communication security protocols. Our study, which also provides an extensive discussion regarding possible paths towards forward secure internet, has been accepted for publication in the 15th International Conference on Security and Privacy in Communication Networks (SecureComm 2019), Orlando, US. *For more details about our research, please check the on-line pre-print[3] which is available at Google Scholar.*

Acknowledgment

Thanks to Andrew Martin for feedback on this article.

References

[1] Alashwali, E.S. (2016). "On Downgrade Attacks in the TLS Protocol". CDT Mini-Project Report, University of Oxford

[2] Alashwali, E.S. (2017). ""Negotiation-Transparency" as a Property in Configurable Protocols".
DPhil. Transfer of Status Report, University of Oxford

[3] Alashwali, E.S., Szalachowski, P., Martin, A. (2019). "Towards Forward Secure Internet Traffic". In: Security and Privacy in Communication Networks (SecureComm), Orlando, US