An Updated Threat Model for Security Ceremonies

Marcelo Carlomagno Carlos*
Information Security Group
Royal Holloway, University of London
Egham, TW20 0EX, UK
marcelo.carlos.2009@rhul.ac.uk

Jean Everson Martina
Universidade Federal de Santa Catarina, INE
Florianópolis, Brazil
everson@inf.ufsc.br

Geraint Price
Information Security Group
Royal Holloway, University of London
Egham, TW20 0EX, UK
geraint.price@rhul.ac.uk

Ricardo Felipe Custódio
Universidade Federal de Santa Catarina, INE
Florianópolis, Brazil
custodio@inf.ufsc.br

ABSTRACT

Since Needham and Schroeder introduced the idea of an active attacker, a lot of research has been made in the protocol design and analysis area in order to verify the protocols' claims against this type of attacker. Nowadays, the Dolev-Yao threat model is the most widely accepted attacker model in the analysis of security protocols. Consequently, there are several security protocols considered secure against an attacker under Dolev-Yao's assumptions. With the introduction of the concept of ceremonies, which extends protocol design and analysis to include human peers, we can potentially find and solve security flaws that were previously not detectable. In this paper, we discuss that even though Dolev-Yao's threat model can represent the most powerful attacker possible in a ceremony, the attacker in this model is not realistic in certain scenarios, especially those related to the human peers. We propose a dynamic threat model that can be adjusted according to each ceremony, and consequently adapt the model and the ceremony analysis to realistic scenarios without degrading security and improving usability.

Categories and Subject Descriptors

H.1.1 [**Information Systems**]: Systems and Information Theory; H.1.2 [**Information Systems**]: User/Machine Systems; K.6.m [**Miscellaneous**]: Security

General Terms

Security, Human Factors, Verification

*Supported by CNPq/Brazil

SAC'13 March 18-22, 2013, Coimbra, Portugal.

Keywords

Threat models, Security Ceremonies, Security Protocols

1. INTRODUCTION

Security protocols are generally secure against passive attackers who eavesdrop the communication medium. However, since Needham and Schroeder introduced the idea of an active attacker in 1978 [19] a lot of research has been conducted in this area in order to prove protocols' claims. Needham and Schroeder's attacker model assumed that the attacker could alter, copy, replay and create messages (or parts of messages) in all communication paths. Dolev and Yao [14] further developed this attacker model by formalising it and adding new assumptions. In general, we can say that the Dolev Yao attacker has complete control of the network but is not able to perform cryptanalysis.

Nowadays, the Dolev-Yao threat model is the most widely accepted model to analyse security protocols [5]. Consequently, there are several security protocols considered secure against Dolev-Yao's assumptions. In general, we assume that if a protocol is secure against this powerful attacker, it is secure against less powerful variations.

However, recent research [13, 17, 15] shows that even protocols verified under Dolev-Yao threat model assumptions might be susceptible to attacks when implemented. Several security protocols, when implemented in practice, are used by humans. The non-deterministic nature of human behaviour can produce situations where an unexpected (but plausible) operation is performed. In general, these problems happen not due to a design flaw or user's misconduct, but due to the implementation of a protocol assumption. Ellison [15] introduced the concept of a broader view to security protocols, called a "ceremony". In his definition, a ceremony is an extension of the network protocol, its nodes may be humans or computers, and the communication channels are not limited to the network.

By extending protocol analysis to ceremony analysis, we can potentially find and solve security flaws that were previously not detectable. However, the formal verification of ceremonies is not a straightforward process. In ceremonies we have new communication mediums, new nodes, and con-

sequently new attacker variations. For that reason, an appropriate threat model must be designed to fit into this new architecture. We argue that, even though Dolev-Yao's threat model can represent the most powerful attacker possible in a ceremony, the attacker in this model is not realistic in certain scenarios. One of the reasons that certain protocols fail when implemented is because their assumptions are either not well specified or not realistic, forcing implementations to create mechanisms to circumvent these problems. Consequently, these workarounds may introduce security problems, making the implementation of the protocol, in certain contexts, flawed. In this case, despite the fact that the problem was created during the implementation, its cause was an inaccurate assumption forced by an unrealistic threat model. In this paper, we revisit Dolev-Yao's threat model so we can have a tailored threat model for ceremonies, allowing more aligned design and implementation components for ceremonies.

In this paper, we present in Section 2 a comparison between ceremonies and protocols. We describe protocols' threat models in Section 3. Our premises for the proposed threat model for ceremonies are discusses in Section 4. The proposed threat model is presented in Section 5. We describe some example scenarios in Section 6. In Section 7, we discuss the gains of describing ceremonies under a realistic threat model. Finally, we conclude our paper with our final thoughts and future expectations in Section 8.

2. CEREMONY VERSUS PROTOCOL

Security protocols can be defined as a prescribed sequence of interactions among entities designed to achieve a certain end [21]. Their goals include authentication, key distribution, secrecy, anonymity and several others. Security ceremonies [15] are a superset of security protocols (as we can see in Figure 1). Ceremonies, when compared to protocols, include additional node types, communication channels and operations which were previously out-of-bounds. As an example of these out-of-bound operations we have user interaction and pre-key distribution.

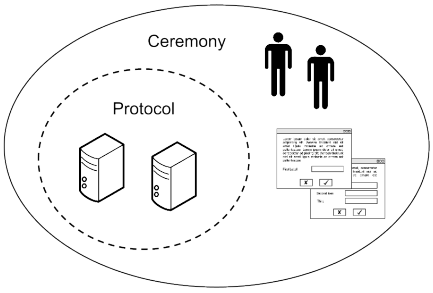

Figure 1: Protocol and Ceremony association

The main difference and probably the most challenging in terms of design and analysis, is the addition of human nodes into the specification. In a protocol specification we usually have devices (e.g. computers) communicating with other devices via a network medium. By including a human node, we have to define and use extra mediums such as user-interfaces, for human-device interaction, and a human medium, to represent speech, gestures, etc, for human-human interaction.

Security ceremonies are designed to help humans achieve their security goals instead of focussing on computer goals. In a protocol specification, we define assumptions to represent out-of-bound operations. In ceremonies we break down these assumptions into smaller and well described assumptions. The inclusion of human interaction, behaviour and cognitive processes adds a considerable amount of complexity to the design and verification tasks. However, their inclusion also enriches the details and coverage of the analysis.

In the same way that a ceremony allows a more detailed analysis of a protocol, the threats, or the capabilities of an attacker under a ceremony scope requires finer granularity in its description. The assumptions made by Needham and Schroeder [19] and extended by Dolev and Yao [14] are the current standard for protocol analysis. However, for ceremonies they are not always consistent with real world threats. For example, an attacker capable of modifying (or replaying) a "speech" packet in a human-human medium is unrealistic if this communication happens in person.

We can anticipate that by specifying and verifying security ceremonies we will be able to encompass a more human-centric security view. The extensions to Dolev-Yao's threat model will help us design more realistic ceremonies which will assist the human peer to assess the threat level he is subject to. By not overstating assumptions we inherently make them plausible and achievable. An example how a protocol can assist a human peer in assessing the threat model he is subject to can be seen in Section 6

3. ABSTRACT THREAT-MODELS FOR PROTOCOLS

A threat model is the setting where our security claims must hold in an implementation environment. A threat model should capture the potential attacker to our protocols and his capabilities. The *Spy*, also called the attacker or intruder, was first mentioned in the classical Needham-Schroeder paper [19]. They made assumptions about the behaviour of principals executing the protocols and the potential attacker they were protecting against. Later, this definition became a central point in discussing whether the protocols were correct or not, and definitely demonstrated how the security claims and their verification are tied to the threat model to which they are subject to.

Needham and Schroeder's assumptions are commonly accepted by the security community, and are up to date and usable even with the new shape the Internet has given to computer networks. Their assumptions are:

- Cryptography in security protocols is perfect and not breakable by the use of brute force or cryptanalysis.
- The attacker can manipulate all communication paths. He can oversee all traffic as well as he can delay, prevent delivery, and fake messages of his own using all resources he has available, with exception to cryptanalysis powers that would be contrary to the first point.
- Principals of the network (except the attacker) follow the protocols. Security protocols do not force all the communication to be carried out in a secure fashion.

With the evolution of computer networks, we see new additions that bring the threat model closer to the real threat that protocols would be subject to. Dolev and Yao [14] formalised their attacker with the previous capabilities but added some further important assumptions:

- The attacker can break messages down to their atomic components and decrypt all encrypted messages to which he possesses the key.
- The attacker can forward encrypted messages he cannot read.
- The attacker can be an internal agent to the protocol that engages in a run to learn information and use it later to leverage gain.

The threat model known as Dolev-Yao is today the standard in terms of security protocols research. From the Dolev-Yao threat model we see the evolution of two different research lines. The first research line agrees with the threat model in terms of architecture and tries to extend it with probabilistic and cryptanalysis powers. Their main motivation is that, by doing such extensions, the threat model would depict an even more realistic threat scenario. This has been focus of interest since Bellare and Rogaway [7] worked on computational complexity and threat models.

The second line of interest in threat modelling description believes that the subtleties of protocols attacks can still be discovered by rearranging the power distribution between the omnipresent and powerful Dolev-Yao attacker and other potentially interested subjects. This line of research brings us more practical scenarios from protocols to adhere in the sense that, threats today are different from the cold war ones historically embedded in the Dolev-Yao model.

We tend to focus more on this second line of research since it is easier to implement into symbolic analysis, which is our aim. An example of the evolution of threat models in the line of rearranging powers is the *BUG* threat model [6]. In BUG, agents participating in the protocol are divided in three groups: Bad, Ugly and Good. They are agents that follow, subvert or change behaviour respectively. The *BUG* threat model is important because of its novelty in having attackers not sharing their knowledge and changing their behaviour during the run. This threat model explores ideas that the attacker will change his behaviour to adapt to the possibilities of gain or loss that may happen.

As a direct derivation from *BUG* we can cite the *Rational Attacker* [3]. The Rational Attackers make cost/benefit decisions on when to behave according to the protocol or not. This new rule based on the a cost benefit analysis fundamentally changes the last rule of Dolev-Yao's proposition. Now the attacker must decide if the gain outweighs the risks of being caught. The Rational Attacker once acting maliciously, shares his information with the superset of attackers.

A follow up along the same lines is the *General Attacker* [3]. In this threat model the cost/benefit function is dropped, but principals still collude as in the original Dolev-Yao model. In fact each peer is a potential attacker that can use the information lawfully acquired to subvert the protocol. This threat model is more realistic than the BUG and the Rational Attacker in an Internet scenario and has the benefit of not having to deal with the gain/loss function.

Finally, we have the introduction of the *Multi-Attacker* [3], which is a variant of the BUG family where each principal may behave as a Dolev-Yao attacker but will never share his long term secrets with other agents. The Multi-Attacker brings a population of attackers that act in a selfish way by not sharing information.

The advantages of evolving the original Dolev-Yao attacker into forms like the *BUG* family of threat models is that these properties more accurately represent their actual execution environments. The possibility of retaliation or anticipation attacks as suggested by Arsac et al. [3] justifies the consideration of such new threat models.

On the area of human centric security we see works from Roscoe and Creese [12] as a good starting point. The main characteristic of their threat model is a division of channels where the information flows. They propose a high bandwidth channel where a Dolev-Yao attacker is present. They also propose a lower bandwidth channel where a weakened version of such attacker is used to represent the threat model involving human peers. Creese and Roscoe highlight the necessity of constraining the human channels to plausible actions. They introduce the idea of empirical channels, where the attacker possesses limited capabilities.

Understanding protocol goals and threat models is key to understanding attacks. Although initially simple, a threat model hides subtleties that can validate or invalidate most claims made regarding the achievement of security goals. Attacks can be seen as the marriage between weak goal achievements and the misunderstanding of the correct threat model.

4. PREMISES FOR CEREMONIES AND CEREMONIES THREAT MODELLING

It seems obvious that developing a ceremony that is secure against a Dolev-Yao attacker will imply that the same ceremony will be secure against any weaker real-world attacker. However, it is often the case that, to guarantee that a certain ceremony is secure against a such powerful attacker, we have to include very complex mechanisms, or degrade usability. By doing that, a new threat is introduced, which is the fact that the user will try to circumvent the security mechanisms in order to accomplish his/her tasks. If we consider a more realistic threat model, which might not be safe against a Dolev-Yao attacker, but it is safe against a real-world attacker, we can prevent the user from being overloaded, and consequently make the ceremony more usable and secure.

One important premise for a reasonable threat model for security ceremonies involving human peers is that **no being is omnipotent in human-human channels.** This premise is easily verifiable by humans. The detection of powers beyond usual human capability is straightforward in the setting of security ceremonies. The impact of such a premise is that, depending on the situation, the presence of an active attacker is not realistic. We already mentioned the example that replaying or blocking "speech" in a human-to-human channel will involve the use of powers that are not feasible for a human peer.

Following a similar aspect, we have a premise that **omnipotency in the human-device channel is not always realistic**. Although we have scenarios where we expect that an attacker has full control over the human-device channel (therefore a Dolev-Yao attacker), in some specific situations such a powerful attacker does not represent reality. An example for such situation is a ceremony that makes use of single-purpose devices (e.g. one-time password generators). When these devices are used, the capabilities of the attacker over the human-device are limited. Thus, the threat model used on such channel is ceremony and context-specific.

Next, we have the premise that **a threat model including human peers should be constrained by the laws of physics.** It is unrealistic to assume an omnipresent at-

tacker in human-human channels. The implications of such a premise will allow human peers to properly choose a location to execute their ceremonies taking into account the verifiable presence of a potential attacker. This can be exemplified by a ceremony in a physical context where humans peers have strict physical access control. A real world example of such premise is execution of security ceremonies for PKIs in safe rooms with strict physical and electromagnetic controls.

Another important premise for security ceremonies threat model is that **humans are capable of performing basic information recall or mathematical operations.** Some security protocols and their related security ceremonies are designed to encompass unrealistic human capabilities regarding the recall of information or the execution of mathematical operations. In a realistic threat model, human peers are required to recall just fresh information and to execute basic mathematical operations. This premise impacts how the personification of the attacker in the human-human channel behaves. Without support from a device, a human peer has limited memory and limited mathematical capability. The presence of external aids is detectable and can be used to verify an expected behaviour. An example of such a premise is the verification of possession of a device in an authentication scenario to generate one-time passwords.

Finally, we have the premise that **one should never use more crypto than needed**. Using more crypto than needed often impacts on usability problems or inaccurate assumptions for a human-device interaction perspective. Although this is not a ceremony specific problem, as noted by Anderson and Needham [2], it could lead to problems in related security ceremonies. The addition of extra layers of crypto, that do not address the threat model, may induce the human who is taking part in the ceremony to misunderstand the threat level he is subject to. An example of such extra layer not addressing the threat model is the usage of one-time password devices by banks. The extra layer of crypto does not address the active man-in-the-middle attacks, but establishes a very strong device possession premise.

With the premises above in mind we propose a threat model that encompass the characteristics of each specific channel. For every channel, we start from Dolev-Yao premises, but we weaken the attacker to fit real-world conditions.

5. PROPOSED THREAT MODEL FOR CEREMONIES

A proposition for a new threat model for security ceremonies is justified because no protocol is executed without context. We know that even if a protocol is proven secure against a powerful attacker (Dolev-Yao), it might still fail due to some reasons, which may include:

- Clear usability problems – the user must have unrealistic capabilities to perform his activities.
- The assumptions are too big/strong or too generic – it is often necessary to assume that previous steps were successfully performed, or that the user is capable of performing some kind of operation.

While those reasons are not directly related to the network channel, and therefore the protocol is secure, we cannot make the same statements when the protocol is implemented. When put in practice, assumptions that involves human-device and human-human interaction have to be implemented somehow. They are replaced by dynamic user-interactions [10]. By doing that, we introduce two new possible communication channels, as Figure 2 illustrates. In this case, we cannot assure that the expected security properties assumed in the protocol design will hold in this ceremony.

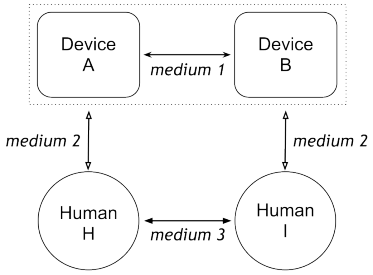

Figure 2: Ceremony communication mediums

Another important issue regarding threat models for security ceremonies is the fact that, humans make different decisions regarding their security based on a dynamic evaluation of the threat level they are subject to in the environment [20]. An example of such embodied decision making regarding threats we are subject to is the evolutionary pressure humans suffered deciding whether to engage in attacks to become hunters or to keep a way of life of gatherers, and thus being exposed to less risk [1]. This inherent faculty of human nature is usually not taken into account when we always assume the worst case scenario as in a Dolev-Yao setting. Some attacks may be thwarted by using an overly pessimistic threat model, but inherently this action will provoke human nature into acting and finding an easier and plausible solution.

With the above in mind we stress that for a security ceremony, the threat model must be adaptive. Even the same protocol (e.g., bluetooth pairing) might need to run under different threat models and achieve goals in different but still reasonable ways. Considering worst case is not always the best option since it degrades usability. The adaptive model we are proposing applies only on the human-device and human-human channels. For network communication (device-device channel) we will always assume a Dolev-Yao attacker since it is the de facto standard. This is important since it is very well studied and developed, and we do not intend to modify protocol verification at the network level.

In addition, having different threat models for different environments can potentially "teach" users to be more aware of threats and to better (intuitively) understand the threat model for each circumstance. For example, a user pairing two bluetooth devices at home is under a different threat than when in an airport, or at an ATM operation ceremony in America and Europe, or even when the ceremony takes place in the same location but at different times. All of these scenarios have different subtleties.

Based on that, the most challenging step in designing and analysing ceremonies is to define the threats and the conditions where this ceremony will be used. A threat model for ceremonies must be ceremony and context-dependent. However, we can define a limited set of threat models that encompasses the great majority of those cases. The existence of a standardised threat model scenario is paramount to the establishment of security goals of ceremonies and for

the comparison between the efficacy of different ceremonies.

Therefore, our proposal for a threat model for ceremonies starts from Dolev-Yao, and then we remove one or more capabilities of the attacker, such as replaying, blocking and creating messages to enable the fulfilment of our premises stated in Section 4. Our final goal is to measure the security of ceremonies against a Dolev-Yao attacker with a smaller set of capabilities. This approach will also help us to reuse some of the abstract verification techniques and tools already in use for security protocols.

To describe our threat model approach, we break down the Dolev-Yao attacker's set of capabilities. Based on this set, we will be able to design and verify ceremonies that are secure against a realistic attacker with different capabilities under different channels (e.g. Dolev-Yao on the network channel, while Dolev-Yao without initiate on the human-device channel, and finally Dolev-Yao without block on the human-human channel). To simplify the notation, we will consider that "DY" is a Dolev-Yao attacker, and everything followed by a "–" symbol will represent the capabilities removed from this attacker. For example, a "DY-BR" means a Dolev-Yao attacker without the blocking and replaying capabilities. We start with the definition for the Eavesdrop capability as shown below:

Definition 1 (Eavesdrop – **E**).

$$\forall X \in M.\ A \rightarrow B : X \Rightarrow X \in knows(I)$$

Our definition for the eavesdrop capability reads as follows: for all messages X in the set of messages M, if the agent A sends to B a message X, this implies that the intruder I will learn X. All the logical connectives have their usual meaning and the set knows(Y), represents the set of knowledge of an agent Y in the protocol.

Another important capability of a Dolev-Yao attacker is the capacity of initiating a new communication with another peer using the knowledge the attacker possesses. The definition 2 describes such capability:

Definition 2 (Initiate – **I**).

$$\forall X \in knows(I).\ I \rightarrow B : X$$

Definition 2 reads as follows: for all messages X in the knowledge set of the intruder I, the intruder I can initiate a communication with a peer B and send message X.

Next, we define the capability that enables the attacker to break down messages to its atomics parts. This capability is relevant so that the Intruder can use atomic components of previously learned messages to produce new ones. The definition for the Break Down capability is shown in 3:

Definition 3 (Atomic Break Down – **A**).

$$\forall \{X, Y\} \in knows(I). \Rightarrow$$
$$\{X\} \in knows(I)\ \wedge\ \{Y\} \in knows(I)$$

The atomic break down definition is: for all pairs composed by some X and Y elements in the knowledge of the intruder I, the element X and Y are also in the knowledge of the intruder I individually.

The cryptographic capabilities of the attacker are presented in the definition 4.

Definition 4 (Crypto – **C**).

$$\forall \{X\}_k \in M \wedge\ k \in knows(I).\ A \rightarrow B : \{X\}k \Rightarrow X \in knows(I)$$

This definition is described as: for all X ciphered using a key k in the set of messages M, and the key k is in the knowledge of the Intruder I, if an agent A sends a message X encrypted with the key k to an agent B, the unencrypted version of X will also be in the knowledge of the intruder I.

The attacker capability of blocking messages, that is, preventing the receiver from learning the contents of a message sent to them, is presented in the definition 5:

Definition 5 (Block – **B**).

$$\forall X \in M.\ A \rightarrow B : X \Rightarrow X \notin knows(B)$$

This definition is read as: For all messages X in the set of possible messages M, if an agent A sends to an agent B a message X, the agent B will not have, in its set of knowledge $knows(B)$, the message X.

An important capability of the attacker is the usage of public known function to fabricate new messages. Examples of such functions can be cryptographic hashes, public-key encryption, or any other function publicly available to the execution of the ceremony. Fabricate may be an n-ary function, differently than presented bellow. The capability of fabricating messages is presented in the definition 6:

Definition 6 (Fabricate – **F**).

$$\forall X \in knows(I) \Rightarrow F(X) \in knows(I)$$

This definition is read as: For all messages X in the set of knowledge of the intruder I, the result of the application of a publicly available function $F()$ is also in the knowledge set of the intruder I.

Spoofing messages is an attacker's capability where he is able to send a message to an agent pretending to be some other agent. The definition of spoof is presented below:

Definition 7 (Spoof – **S**).

$$\forall X \in knows(I).\ Spoof(I, A) \rightarrow B : X$$

This definition is read as: for all messages X in the knowledge of the intruder I, an intruder I can spoof the identity of an agent A to the agent B and send a message X. Spoof differentiates from Initiate in deliberately not allowing the attacker to be an internal agent in the execution of the ceremony.

The final capability that we can selectively detach from the original Dolev-Yao attacker model is the capability of re-ordering messages, as in Definition 8.

Definition 8 (re-Order – **O**).

$$\forall X, Y \in M.\ A \rightarrow B : X\ \wedge\ C \rightarrow B : Y \Rightarrow$$
$$Y \in knows(B)\ \wedge\ ...\ \wedge\ X \in knows(B)$$

It reads as: for all messages X and Y in the set of possible messages M, so that firstly A sends to B the message X, and later C sends the message Y to B, it will imply that the attacker can make those messages to be added to B's set of knowledge in a different order than they were sent,

and possibly with some other messages in between. An important aspect of this capability is that it is described from the receiver's point of view, since there are many different ways of the Intruder achieving it. We deliberately show this on Definition 8 by using two different senders. Nevertheless, this can also be achieved with $A = C$.

Some of the characteristics of the Dolev-Yao attacker are not directly shown here, since they can be achieved by the combination of our definitions. For example, the capability of **Modifying (M)** messages on the communication channels can be defined as the use of **Block** + **Initiate**, while **Replaying (R)** messages can be represented as **Eavesdrop** + **Initiate**.

Another way of approaching the weakened version of the attacker model would be instead of using decomposition we would start with no threat model, or simply, a model where the attacker has "no capabilities" (N). From that point on we would add to the (N) attacker the desired capabilities, such as N + E for eavesdrop only, or N + EB for eavesdrop and block only. We would like to stress that this is more a denotational problem and using both strategies we will always achieve an equivalence (such as: DY-IDBRSM = N+E). Nevertheless, for the sake of simplification this second approach may prove useful.

Our proposal can be clearly differentiated from similar work by its systematic approach and flexibility on weakening the attacker to fit the desired threat model scenario. We stress that, although the usage of different threat models in different communications channels has been already covered in other works like Belfanz et. al. [4], Creese et. al. [12] and Wong and Stajano [23], the novelty of our work is based on its specific target for security ceremonies and its systematic weakening of the attacker to reflect realistic threats.

6. EXAMPLE SCENARIO: BLUETOOTH PAIRING PROTOCOL

The bluetooth pairing protocol is designed to allow one device to recognise and connect to another. The pairing protocol, from version 1.0 to version 2.0 (known as **legacy pairing**) is performed in a way where both devices are required to enter a common PIN to establish the connection. For devices with limited input capabilities, a fixed PIN number is used (e.g. 1234), whereas for more advanced devices, the PIN might be numeric (e.g. old mobile phones) or alphanumeric (e.g. modern mobile phones and computers).

In more recent versions of bluetooth (2.1 + EDR and above) a new protocol, called "secure simple pairing" (SSP), is described. This protocol aims to simplify the pairing process from the user's point of view in addition to increasing protection against passive (eavesdrop) and active (man-in-the-middle) attacks. The protocol defined in version 2.1 solves several flaws that allowed attackers to deploy man-in-the-middle (MITM) attacks on earlier versions of the specification and presents four different association modes that aim to encompass most of the devices types [9, 8].

The **numeric comparison** association model is designed for devices capable of displaying digits (a six digit number) and accepting user inputs ("yes" or "no"). In this mode, the device displays six digit numbers on both devices and the users are asked whether the numbers are the equal on both devices. If they are equal, the pairing is successful. The **just works** model focus on devices without displays and possibly without user's input. In this model, both devices associate in the same way as the numeric comparison protocol, but the numbers are never shown to the user[1]. The **out of band (OOB)** model is designed for situations where an OOB mechanism is used to discover the devices and exchange/transfer cryptographic numbers used in the pairing process. The implementation may differ, but an example would be a solution where a user initially touches the two devices together and then confirms via user interface that he wishes to pair with the other device. Finally, the **passkey entry** model is designed for scenarios where one device has only input capabilities (e.g. keyboard) and the other has only output capabilities (e.g. display). In this model, the user is shown a six digit number on the device with display and he inputs the number on the other device. If the values match, the pairing succeeds [9].

Although version 2.1 of the protocol is cryptographically secure, we can still find attacks based on a forced change from a strong association mode to a weaker one [16], or a user's misinterpretation of concurrent pairing sessions [11]. Both attacks focus on possible real-world uses (and on specific association modes) of the protocol rather than its specification, that is, they focus on the bluetooth SSP ceremony rather than the SSP protocol.

When we thoroughly analyse the protocol specifications, we find that the association modes are designed under assumptions that imply in a weaker threat model for the pairing protocol. In the legacy mode, the device-device medium (DD) is designed considering a DY attacker, while the human-device (HD) and human-human mediums (HH) are assumed to have no attackers. Although there are other flaws in this protocol [18, 16], a relevant, simple and effective passive attack can be found if we add the capability of eavesdropping to the attacker on either HD or HH mediums. In this case, the attacker would learn the PIN by just eavesdropping those mediums (hearing the PIN value) and with that, he could decode all the messages. This works similarly to the attack described in [22], but without the need of deploying a brute force attack on the PIN. This attack would easily be captured when verifying this protocol as a ceremony.

In the case of the SSP protocol, each association mode also needs to be analysed under a different threat model, and more importantly, each implementation should respect the specified threat model. In our examples, we will use the **numeric comparison** against a Dolev-Yao attacker and also under other variations of the attacker capabilities. The following specification describes phase 2 of the SSP protocol using the numeric comparison mode specified as a ceremony. The text below the arrows specify the channel used according to the nomenclature defined earlier, and we define A and B the devices used by the users U_A and U_B respectively.

$$
\begin{array}{llcll}
M_1. & B & \xrightarrow[DD]{} & A & : \quad C_b = f1(pk_B, pk_A, Nb, 0) \\
M_2. & A & \xrightarrow[DD]{} & B & : \quad N_a \\
M_3. & B & \xrightarrow[DD]{} & A & : \quad N_b \\
M_4. & A & \xrightarrow[HD]{} & U_A & : \quad V_a = g(pk_A, pk_B, N_a, N_b) \\
M_5. & B & \xrightarrow[HD]{} & U_B & : \quad V_b = g(pk_A, pk_B, N_a, N_b) \\
M_6. & U_A & \xrightarrow[HH]{} & U_B & : \quad V_a \\
M_7. & U_B & \xrightarrow[HH]{} & U_A & : \quad V_b
\end{array}
$$

[1]The user may be asked whether to accept the connection, but it depends on the manufacturer's implementation.

In our analysis, we considered the ceremony using the **numeric comparison** (NumComp) mode under different variations of the threat model. The theorems bellow presents the results of our analysis.

THEOREM 1 (NUMCOMP+DY). *If the protocol messages M_1 to M_7 are run against a DY attacker, the attacker can prevent U_A from learning V_a or V_b and U_B from learning V_b or V_a, forcing them to learn V_i instead.*

$$\frac{M_{1...7} \cup DY}{\begin{array}{r} V_a \wedge V_b \wedge V_i \in knows(I) \wedge \\ V_a \notin knows(A) \wedge V_b \notin knows(A) \wedge \\ V_b \notin knows(B) \wedge V_a \notin knows(B) \wedge \\ V_i \in knows(U_A) \wedge V_i \in knows(U_B) \end{array}}$$

PROOF. Due to space limitations we will skip the initial steps of the proof and assume that the intruder I, acting as a man-in-the-middle, initiated two parallel pairing sessions with A and B during Messages M_1 to M_3. The authentication from A to B starts on M_4 where the value V_a is sent to U_A. The equivalent message from B to U_B occurs in M_5. A DY intruder I, by using his block (B) and initiate (I) capabilities, can prevent the message M_4 and M_5 from being delivered to U_A and U_B respectively, and instead, send them any chosen value V_i. In M_6 and M_7, A and B would complete the protocol by sending V_i to each other, successfully concluding the pairing and allowing the man-in-the-middle attack to be deployed. □

Using an alternative threat model, which we term the Adaptive Threat Model V1, we assume the attacker can only eavesdrop the HD channel. In the specific case of the bluetooth pairing, the assumption is that the device is free from malware and the display is presenting the correct information.

THEOREM 2 (NUMCOMP + AD. THREAT MODEL V1). *If the protocol messages M_1 to M_3 are run against a DY attacker; the messages M_4 to M_5 are run against a N+E attacker; and messages M_6 to M_7 are run against a DY attacker, the attacker can prevent U_A from learning V_b and U_B from learning V_a, forcing them to learn the repetition (replay) of V_a and V_b (respectively) instead.*

$$\frac{(M_{1...3} \cup DY) \wedge (M_{4...5} \cup N + E) \wedge)(M_{6...7} \cup DY)}{V_a \wedge V_b \in knows(I) \wedge V_a \notin knows(B) \wedge V_b \notin knows(A)}$$

PROOF. Again, due to space limitations we will skip the initial steps of the proof and assume that the intruder I, acting as a man-in-the-middle initiated two parallel pairing sessions with A and B during Messages M_1 to M_3. The authentication from A to B starts on M_4 where the value V_a is sent to U_A. The equivalent message from B to U_B occurs in M_5. In this case, the N+E intruder can only learn the values V_a and V_b. In M_6 and M_7, A and B complete the protocol by sending V_a and V_b respectively, to each other. A DY attacker in messages M_6 and M_7 can perform a similar attack to the one described in Theorem 1. By preventing V_a from being delivered from U_A to U_B in M_6 and V_b from U_B to U_A in M_7, and then replaying the values V_b and V_a (respectively) instead (by making use of Blocking, Replaying and Spoof capabilities), the protocol run would be successfully concluded and would allow a man-in-the-middle attack to be deployed. □

Finally, when using an alternative threat model, which we call the Adaptive Threat Model V2, the attacker can eavesdrop both the HD and the HH channels, that is, the attacker can eavesdrop on any communications in the pairing process that involve the humans.

THEOREM 3 (NUMCOMP + AD. THREAT MODEL V2). *If the protocol messages M_1 to M_3 are run against a DY attacker and the messages M_4 to M_7 are run against an N+E attacker the attacker cannot produce any relevant attack.*

$$\frac{(M_{1...3} \cup DY) \wedge (M_{4...7} \cup N + E)}{\emptyset}$$

PROOF. Once again, due to space limitations we will skip the initial steps of the proof and assume that the intruder I, acting as a man-in-the-middle, initiated two parallel pairing sessions with A and B during Messages M_1 to M_3. The authentication from A to B starts on M_4 where the value V_a is sent to U_A. The equivalent message from B to U_B occurs in M_5. In this case, the N+E intruder can only learn the values V_a and V_b. In M_6 and M_7, A and B complete the protocol by sending V_a and V_b respectively, to each other. In messages M_6 and M_7, the N+E intruder can only learn the values V_a and V_b, and the in this case, V_a received by U_B in M_6 and V_b received by U_A in M_7 would not match the V_b and V_a in $knows(B)$ and $knows(A)$ respectively, not allowing the attack to succeed. □

Although the attack described in Theorem 1 is plausible in real world scenarios, it is very difficult to be deployed. An attacker would have to corrupt both devices as well as start parallel sessions with both users during a short period of time. This is a good example of a technically feasible attack but highly unlikely to happen in practice. By removing capabilities B and I of the attacker, we can analyse the protocol further, and possibly find other (more) relevant attacks.

The second attack, found in Theorem 2 is completely unrealistic. To be deployed in practice, the attacker would have to block a communication between two humans and then replay some data over a channel where the user would easily notice if some other party wanted to spoof the identity of the sender. In this case, the attack does not exist in practice.

The idea is similar for the other association modes. Each one of them must consider the real-world scenario and define the threat model. In addition to that, it should not be possible to use an association mode under a different threat model than specified.

7. GAINS OF CEREMONY DESCRIPTION UNDER A REALISTIC THREAT MODEL

The misunderstanding of the correct threat model for each association mode in the Bluetooth protocols would lead us to two types of incorrect conclusions:

- The protocol (and related ceremony) is not secure due to the fact they do not cope with an overly-pessimistic threat model. Usually this would lead the protocol/ ceremony designer to add features that would assume non-plausible assumptions and/or degraded usability.
- The protocol is secure, but the user misunderstands the threat he is subject to. This is again related to the

costs of the added security measures to protect against all threats, even the unrealistic ones.

The ceremony for the bluetooth pairing can be described avoiding the above conclusions. The ceremony could enforce the correct threat model choice at implementation level by restricting the modes available to the strongest type possible for that specific device; or at application level, where the application would dynamically allow/block association modes depending on the environment. The ceremony should take into account, for example, whether there are more bluetooth enabled devices around before allowing pairing under the just works mode. If there is more than one, the just works mode should not be available since the weakened threat model requires that, if only one device is found, the just works mode could be securely used, since it respects the threat model specified for its use.

This kind of ceremony potentially leads to a positive side-effect. It trains users to detect different threat models for those situations. In the same example, the user is able to learn through experience that to be more protected in public environments he needs to use stronger authentication mechanisms, while in private he is subject to different threats.

Although the examples discussed in this paper focus on the bluetooth pairing protocol, the idea of using variations of the threat model can be applied for several different ceremonies, varying from ATM authentication ceremonies to TLS handshake protocol implementations and its variations.

8. CONCLUSIONS

The existence of a single worst-case scenario threat model is justifiable in security protocol scenarios. However, the same cannot be said for security ceremonies. Human agents executing security ceremonies are constrained by the laws of physics and usual capacities expected from human beings. The existence of such a powerful agent in a setting involving human-to-human communication is not plausible and is likely to demand solutions that are not tailored to reality.

Our approach for describing a threat model for security ceremonies is based on a well established model for security protocols. In our approach we weaken the attacker to conform to the premises governing human-device interaction and human-to-human interaction. This strategy seems plausible because it will help security protocols and ceremony designers to develop ceremonies with reasonable assumptions and tailored to the real capacities of the attacker.

In this paper we presented our proposal and demonstrated by examples that it is already embedded in industry deployed protocols. We described a fraction of a ceremony where a dynamic threat model using the weakened versions of the all powerful attacker is used. Additionally, we showed that we can design a ceremony that helps the user by ensuring the correct assumptions made by protocol designers.

Our future work on this proposal will be on the specification of the threat model using an abstract verification method for security protocols tailored for security ceremonies. By developing this extension we believe the testing and design of security ceremonies can be automated.

9. REFERENCES

[1] R. D. Alexander. The evolution of social behavior. *Annual Review of Ecology and Systematics*, 5, 1974.

[2] R. Anderson and R. Needham. Robustness principles for public key protocols. In *CRYPTO '95*. Springer-Verlag, 1995.

[3] W. Arsac, G. Bella, X. Chantry, and L. Compagna. Multi-attacker protocol validation. *Journal of Automated Reasoning*, 46(3-4), Apr. 2011.

[4] D. Balfanz, D. K. Smetters, P. Stewart, and H. C. Wong. Talking to strangers: Authentication in ad-hoc wireless networks. In *NDSS'02*, San Diego, 2002.

[5] G. Bella. *Formal Correctness of Security Protocols*. Information Security and Cryptography. Springer Berlin Heidelberg, New York, 2007.

[6] G. Bella, S. Bistarelli, and F. Massacci. Retaliation: Can we live with flaws? In *IACS*, volume 6. IOS Press, 2006.

[7] M. Bellare and P. Rogaway. Entity authentication and key distribution. In *CRYPTO' 93*, volume 773 of *LNCS*. Springer, 1994.

[8] Bluetooth Special Interest Group. Bluetooth specifications 1.0 – 2.1+EDR. Technical specifications, http://www.bluetooth.com, 1999–2007.

[9] Bluetooth Special Interest Group. Simple pairing whitepaper v10r00. Technical report, Aug. 2006.

[10] M. C. Carlos, J. E. Martina, G. Price, and R. F. Custodio. A proposed framework for analysing security ceremonies. In *SECRYPT'12*. SciTePress, July 2012.

[11] R. Chang and V. Shmatikov. Formal analysis of authentication in bluetooth device pairing. In *FCS-ARSPA'07*, 2007.

[12] S. Creese, M. Goldsmith, A. W. Roscoe, and I.Zakiuddin. The attacker in ubiquitous computing environments: formalising the threat model. In *FAST'03*, 2003.

[13] R. Dhamija, J. D. Tygar, and M. Hearst. Why phishing works. In *SIGCHI'06*, New York, 2006. ACM.

[14] D. Dolev and A. C. Yao. On the security of public key protocols. *IEEE Trans. on Inform. Theory*, 29, 1983.

[15] C. Ellison. Ceremony Design and Analysis. Cryptology ePrint Archive, Report 2007/399, Oct. 2007.

[16] K. Haataja and P. Toivanen. Practical man-in-the-middle attacks against bluetooth secure simple pairing. In *WiCOM '08*, oct 2008.

[17] M. Jakobsson. The human factor in phishing. In *Priv. & Sec. of Consumer Information '07*, 2007.

[18] M. Jakobsson and S. Wetzel. Security weaknesses in bluetooth. In *CT-RSA 2001*, volume 2020 of *LNCS*. Springer, 2001.

[19] R. M. Needham and M. D. Schroeder. Using encryption for authentication in large networks of computers. *Comm. of ACM*, 21(12), 1978.

[20] G. Parker. Assessment strategy and the evolution of fighting behaviour. *Journal of Theoretical Biology*, 47(1), 1974.

[21] P. Ryan and S. Schneider. *Modelling and analysis of security protocols*. Addison Wesley, 1 edition, 2001.

[22] Y. Shaked and A. Wool. Cracking the bluetooth pin. In *MobiSys '05s*, New York, USA, 2005. ACM.

[23] F. L. Wong and F. Stajano. Multichannel security protocols. *IEEE Pervasive Computing*, 6(4), Oct. 2007.