Random polynomial-time attacks and Dolev-Yao models

Mathieu Baudet *
LSV — CNRS UMR 8643 & INRIA Futurs projet SECSI & ENS Cachan
61, av du président Wilson 94235 Cachan Cedex, France
baudet@lsv.ens-cachan.fr

Abstract

For several decades two different communities have been working on the formal security of cryptographic protocols. Many efforts have been made recently to take benefit of both approaches, in brief: the comprehensiveness of computational models and the automatizability of formal methods. The purpose of this paper is to investigate an original approach to relate the two views, that is: to extend existing Dolev-Yao models to account for random polynomial-time (Las Vegas) computability. This is done first by noticing that Dolev-Yao models can be seen as transition systems, possibly infinite. We then extend these transition systems with computation times and probabilities. The extended models can account for normal Dolev-Yao transitions as well as nonstandard operations such as inverting a one-way function. Our main contribution consists of showing that under sufficient realistic assumptions the extended models are equivalent to standard Dolev-Yao models as far as security is concerned. Thus our work enlarges the scope of existing decision procedures.

Keywords: Cryptographic protocols, random polynomial time, Dolev-Yao model, Markov decision processes.

1 Introduction

Proving the security of cryptographic protocols has been a major concern ever since flaws were first discovered in some established protocols, the most well-known example being [20]. Rigorous approaches now exist and have allowed the analysis of many protocols with respect to various security models, depending on the attacker's capabilities and purposes [13, 22, 34]. However this variety of approaches may tend to puzzle the practitioner. Indeed two families of models with very little in common have been used for twenty years by two different communities.

A first class of models is the *computational* ones. In those models security is defined in a semantic way by requiring the probability of success of any attacker to be negligible [12]. The class of attacks considered here include virtually all logical attacks implementable by a probabilistic polynomial-time Turing machine. In this approach a proof of security consists in a reduction proof: from any hypothetical attack it shows how to build a random polynomial

*Partially supported by the the RNTL projects EVA and Prouvé, the ACI Sécurité Informatique Rossignol, the ACI Cryptologie Psi-Robuste, and the ACI jeunes chercheurs "Sécurité informatique, protocoles cryptographiques et détection d'intrusions".

algorithm that solves a reputedly intractable problem. Computational security proofs, when they can be achieved, are thus considered strong evidence of security.

A second class of models is used by the community of *formal methods*, and includes typically the Dolev-Yao model [11] and cryptographic process calculi such as the Spi-calculus [1]. By focusing on the protocol layer, these models aim to account for a variety of attacks resulting from complex interactions between an active attacker and a possibly unbounded number of parallel sessions. This is indeed a very hard task in the computational models, where already a passive attacker may lead to highly complex reduction proofs. In the approach based on formal methods, principals exchange structured messages rather than bit-strings. Cryptographic primitives such as hashing, symmetric and asymmetric encryption are assumed perfect in the sense that no attacker is able to retrieve any information from a ciphertext (*resp.* from a hash-code) without the appropriate key. Formal models are now the basis for many automatic tools used to verify protocols ([21, 26, 16, 25, 7] and many others).

Motivated by these observations, efforts have been made to relate the two views of cryptography [3, 19, 34, 15, 6]. Better understanding the links between the two approaches would indeed benefit both communities. For the formal-method approaches, this would help providing more precise justifications and give directions for extending the expressivity of the models and the automatic analyzers. For the computational models, it would give elements toward partially automatizing the security proofs. One could imagine *e.g.* proofs in two steps: the first would establish sufficient computational-security hypotheses on the cryptographic primitives, the second would deal with the protocol aspects by an automatic procedure.

The purpose of this paper is to investigate an original approach in this direction, which is to bring the *existing* Dolev-Yao models closer to the computational models. We show that it is possible to introduce the notion of random polynomial-time calculability in a large class of Dolev-Yao models at no cost. More precisely we prove that the augmented models are equivalent to the standard ones as far as security is concerned. Hence our work enlarges the scope of existing decision procedures.

Related work. Previous work on this theme includes the pioneering paper of Abadi and Rogaway [3], further refined by [2, 23, 18]. There a logic of indistinguishability is introduced and is shown computationally secure against pure eavesdropping attacks. Another approach is investigated by Lincoln et al. [19, 27, 31] by means of a probabilistic process calculus.

Recently Warinschi [34] gave for the first time a computational proof for the Needham-Schroeder-Lowe protocol. A substantial progress was achieved by Backes et al. [6] by providing a generic cryptographic library and its idealized Dolev-Yao-like version. The key result is a *simulatability* theorem which states that every random polynomial-time attack on the real cryptosystem must have a counterpart in the ideal one unless cryptographic primitives are broken. This framework is applied in [5] to give another computational proof of the Needham-Schroeder-Lowe protocol. These works constitute important advances in so far as they allow computational security to be proved on full protocols, by relying on (strong) standard cryptographic assumptions on primitives. Yet in each case studying a different protocol still requires a new proof by hand. Indeed the idealized library of [6] seems to remain far from the Dolev-Yao models handled by typical automatic analyzers [21, 16, 25, 26, 7]. It also rests on a sophisticated computation and communication model. Although we do not consider arbitrary Turing machines as attackers, we can argue that our model is simpler and more transparent. It is based on (infinite) transition systems with (probabilistic) durations.

A similar model to ours was developed independently by Zunino and Degano [35], simultaneously to the first versions of this paper. There, a cryptographic process calculus is introduced, where the capability of the attacker is extended with probabilistic nonstandard operations such as randomly guessing a secret key. Their main result, as ours, states that security in the augmented model is equivalent to security in the standard model. This can be seen as a consequence of our theorem, applied to the semantics of the considered language and where the times of computation are restricted to be either 0 or ∞.

Outline of the paper. In Section 2, we describe an abstract view of a general class of formal models including Dolev-Yao variants. The abstraction consists in modeling Dolev-Yao security as the unreachability of certain unsafe states —where for instance a secret has been illegitimately obtained— in possibly infinite transition systems. We then extend this model to include two features absent from the simplistic Dolev-Yao-like models: computation times (Section 3), and probabilistic computations (Section 4).

In Section 3, we only consider the extension of the simplistic models of Section 2 to include deterministic computation times for each action. This can be seen as the first ring of our approach. This is done by labeling transitions by functions of a complexity parameter n. Usual operations of the Dolev-Yao intruder are modeled by polynomially-bounded (polynomial) times. The benefit of this model is its simplicity: it accounts for so-called *extraordinary operations* —such as guessing a key or breaking a cryptographic primitive— by means of new transitions labeled with non-polynomial times. Security is defined as the fact that no unsafe state can be reached in polynomial time. Our contribution here lies in proving that under sufficient realistic assumptions the security of our extended model is equivalent to the security of the underlying Dolev-Yao model.

We deal with the expected generalization to a probabilistic framework in Section 4. There tractable operations are those for which a random polynomial-time (Las Vegas) algorithm exists. Reachability is defined as the results of $1\frac{1}{2}$-player games between the attacker and a probabilistic opponent (*a.k.a.* Markov decision processes). We show again that the security of the underlying Dolev-Yao model is equivalent to the security of the extended model, defined as the fact that no strategy can reach the set of unsafe states with a non-negligible probability. We conclude in Section 5.

2 Dolev-Yao models and transition systems

Dolev-Yao models distinguish themselves from other models by several particularities:

- First cryptographic primitives are assumed to be perfect: there is no way to retrieve any information about a message from its hash-code or from its encryption unless we have the adequate key.
- More generally Dolev-Yao models suppose that the attacker is not interested in —and does not exploit— partial or probabilistic information. In other words, messages are considered secret unless they are entirely and definitely compromised.
- The network is modeled in the most pessimistic way. Namely the principals' messages can not only be read but also deleted whereas new messages can be forged by the attacker. In this context, it is natural to assimilate the network to the intruder itself.

These choices lead to a modeling where the principals and the attacker are represented by a set of inference rules. Each rule intuitively states that under certain conditions about previously seen messages, the network might learn some other messages, either by receiving them from a principal or by inferring them from its knowledge.

As an illustration let us consider the Diffie-Hellman Key-Exchange protocol [10]. Suppose a prime number p, a generator g of $\mathbb{Z}_p^*$ and some acknowledgment message Ack have been chosen in advance. We write $\{X\}_K$ for the encryption of message X with key K by some symmetrical encryption algorithm. A session of the protocol between two principals A and B can be described as follows:

1. $A \to B : \; g^{N_a}$
2. $B \to A : \; g^{N_b}$
3. $A \to B : \; \{Ack\}_{g^{N_a N_b}}$

where N_a and N_b design private randomly-chosen fresh numbers. The claim of this protocol is that at the end $g^{N_a N_b}$ is a secret shared between the two principals A and B. They both know it because $g^{N_a N_b} = (g^{N_a})^{N_b} = (g^{N_b})^{N_a}$ so the only thing to do is to exponentiate the other's message with one's number. For one session the claim of secrecy is known as the intractability of the (computational/decisional) Diffie-Hellman problem.

In order to study several sessions in parallel, one can resort to an abstracted Dolev-Yao model. A possible model could be sketched as follows. Messages are terms defined by:

$$\begin{array}{rcll} M & ::= & N & \text{nonces } N \in \mathcal{N} \\ & | & Ack & \text{acknowledgment message} \\ & | & \{M_1\}_{M_2} & \text{symmetric encryption} \\ & | & e(M) & \text{modular exponentiation of } g \text{ by } M \\ & | & M_1 \oplus M_2 & \text{product inside exponents} \end{array}$$

where $\oplus$ is an associative-commutative (AC) symbol and the other symbols are free. Formal models with non-free symbols, that is, satisfying algebraic theories such AC or XOR, have been studied among others in [24, 9, 33, 14]; practical results on the GDH protocol were achieved in [29, 9, 32].

Each principal $i \in \mathcal{I}$ has an internal state q_i that contains for each opened session $s = (s_{role}, s_{name}, s_{step}, s_{nonce}, s_{data}) \in q_i$ where it is involved:

- its role in the protocol $s_{role} \in \{A, B\}$
- the name of the correspondent s_{name},
- the number of the expected next message in the protocol $s_{step} \in \{1, 2, 3, done\}$,
- the private nonce s_{nonce},
- the received data s_{data} (here this is the $e(N)$ sent by the correspondent).

The intruder's state is the set of all messages E that he knows (from the network or from its deductions). The global state of the system is the product of the states of all the principals and the intruder $q = ((q_i)_{i \in \mathcal{I}}, E)$. The global state can evolve according to two kinds of rules: communication rules and deduction rules. Communication rules specifically describe the rules of the protocol. Here for instance the rules 2 and 3 seen by the role A would be:

if a principal i has initiated a session as A with some j, if its private session number is N, and if the network knows a message $e(N')$, he may send the message $\{Ack\}_{e(N \oplus N')}$.

Formally this is written:

if $s = (A, j, 2, N, s_{data}) \in q_i$ and $e(N') \in E$
then $(q, E) \longrightarrow (q', E')$
where $q'_l = q_l$ for $l \neq i$, $q'_i = q_i - \{s\} \cup \{s'\}$, $s' = (A, s_{name}, done, s_{nonce}, e(N'))$,
$E' = E \cup \{ \{Ack\}_{e(N \oplus N')} \}$.

The deduction rules are protocol independent. They describe the possible deductions for the attacker. Typical deduction rules are:

- encryption: if $M \in E$ and $K \in E$ then $(q, E) \longrightarrow (q, E \cup \{\{M\}_K\})$,
- decryption: if $\{M\}_K \in E$ and $K \in E$ then $(q, E) \longrightarrow (q, E \cup \{M\})$,
- exponentiation: if $e(M) \in E$ and $M' \in E$ then $(q, E) \longrightarrow (q, E \cup \{e(M \oplus M')\})$.

Our point is that these rules can always be seen as an (infinite) transition system. In many cases security can be defined as a *safety property*, *i.e.* the fact that certain *unsafe* states are unreachable. We will concentrate on this notion in the following. This includes for instance secrecy and various forms of authentification [4].

3 Transition systems with computation times

In the previous section we have outlined the fact that Dolev-Yao models can be seen as transition systems. In this section, we consider a slightly more complex model where the transitions are labeled by computation times. These times are functions of a security parameter n, meant to represent the overall strength of the cryptographic schemes, such as the size of the keys.

Formally, a *transition system with computation times* is a triple $T = (Q, q^0, \delta)$ where Q is the set of states, q^0 is the initial set, and $\delta : \mathbb{N} \times Q \times Q \rightarrow [0, \infty]$ is a weight function that maps every $n \in \mathbb{N}$ and every transition to a non-negative real number or to infinity (modeling an impossible transition). Besides we write $q_1 \xrightarrow{f(n)} q_2$ when for all n, $\delta(n, q_1, q_2) = f(n)$. Notice that we do not assume that Q is finite. Practical implementations of Dolev-Yao models are usually based on some finite representations of such infinite graphs.

Doing so, we gain the ability to include nonstandard transitions in the model, such as inverting a one-way function or guessing a key. In the previous example, typical nonstandard transitions would be:

- illegitimate decryption: if $\{M\}_K \in E$ then $(q, E) \xrightarrow{f_1^0(n)} (q, E \cup \{M\})$,
- key guessing: if $\{M\}_K \in E$ (and $M \in E$) then $(q, E) \xrightarrow{f_2^0(n)} (q, E \cup \{K\})$,
- discrete logarithm: if $e(M) \in E$ then $(q, E) \xrightarrow{f_3^0(n)} (q, E \cup \{M\})$.

For these *extraordinary* operations it is fair to assume time-complexities $f_i^0(n)$ to be intractable. Following a standard asymptotic approach we define tractable transitions as those labeled by a polynomially-bounded (in short, polynomial) function of n.

Definition 1. *$q_1 \xrightarrow{f(n)} q_2$ is called* tractable *if $f(n)$ is bounded by a polynomial, or equivalently if $\frac{\log f(n)}{\log n}$ is bounded from above ($n \geq 2$).*

Clearly enough, defining intractability as the negation of tractability is not sufficient for security purposes. Such a definition would not eliminate *e.g.* primitives that are breakable for even values of n but secure for odd values. For this reason, intractable transitions has to be defined in a stricter way.

Definition 2. *$q_1 \xrightarrow{f(n)} q_2$ is called* intractable *if $\lim_{n\to\infty} \frac{\log f(n)}{\log n} = \infty$.*

This is the same as requiring that $f(n)$ eventually reaches —and remains higher than— any polynomial, or that $\frac{1}{f(n)}$ is *negligible* in the usual cryptographic sense [12, 13].

Let us define now the set of states that can be reached in polynomial time from the initial state. To do so, we define the n-duration of a path $\gamma : q_0 \xrightarrow{f_1(n)} q_1 \xrightarrow{f_2(n)} \dots \xrightarrow{f_p(n)} q_p$ as the sum of its internal durations:

$$|\gamma|_n = \sum_{i=1}^{p} f_i(n)$$

The n-time cost of a state q is the greatest lower bound of the durations of the paths γ going from the initial state q^0 to q:

$$|q|_n = \inf\{|\gamma|_n, \gamma : q^0 \to \dots \to q\}$$

Finally we will say that a state q can be reached in polynomial time if $|q|_n$ is polynomial *i.e.* if $\frac{\log |q|_n}{\log n}$ is bounded from above ($n \geq 2$).

Security is defined as the fact that all polynomially-reachable states satisfy a given security property. The question at this point is whether or not the security of our extended model reduces to the security of the underlying standard Dolev-Yao model, obtained by removing intractable transitions, then ignoring computation time altogether.

3.1 Reduction theorem for finite graphs

We start proving a reduction theorem in the case of finite transition systems.

Proposition 1. *Let $T = (Q, q^0, \delta)$ be a transition system with computation times, assume Q finite. Suppose that every transition is either tractable or intractable. Then a state q is reachable in polynomial time if and only if there exists a path $\gamma : q^0 = q_0 \xrightarrow{f_1(n)} q_1 \xrightarrow{f_2(n)} \dots \xrightarrow{f_p(n)} q_p = q$ such that every $f_i(n)$ ($1 \leq i \leq p$) is polynomial in n.*

The interpretation of this theorem is that extending a (finite) Dolev-Yao model with extraordinary but intractable transitions does not change the set of tractably reachable states. Thus both systems are equivalent as far as security is concerned.

Proof. The right-to-left implication is clear. Let us consider a state q such that $|q|_n$ is polynomially bounded: let $M > 0$ be such that $\forall n \geq 2$, $|q|_n \leq n^M$.

For each intractable transition $q_1 \xrightarrow{f(n)} q_2$, by definition there exists a n_0 such that: $\forall n \geq n_0, f(n) \geq n^{M+1}$. Recall that Q is finite, so the number of intractable transitions is finite. Therefore for $n_0 \geq 2$ large enough, the previous inequality holds for *every* intractable transition at the same time.

Now suppose that every path γ from q^0 to q contains at least one intractable transition. Since weights are positive, this would imply for all $n \geq n_0$, for all such γ, $|\gamma|_n \geq n^{M+1}$. Thus $|q|_n \geq n^{M+1} > n^M$. Contradiction. $\square$

3.2 Reduction theorem for infinite graphs

We now try to generalize the previous result to the infinite case. Some care must be taken because the existence of a uniform value n_0 of n in the previous proof is not guaranteed: intuitively we may have an infinite sequence γ_0, $\gamma_1, \ldots$, γ_k, $\ldots$ of paths from q_0 to q such that each $|\gamma_k|_n$ is null for $n \leq k$, which implies $|q|_n = 0$ for all n, and yet for all k, $\lim_{n\to\infty} \frac{\log |\gamma_k|_n}{\log n} = \infty$.

Fortunately this case is unlikely to happen for our purpose. Recall that intractable transitions model some new extraordinary rules in the Dolev-Yao approach. Although rules may have infinitely many instances (*e.g.* sending M over the network would be implemented by as many "send" transitions as there are possible messages, and messages are terms in Dolev-Yao models, of which there are infinitely many), most likely a finite number of rules is applied to a finite number of cryptographic primitives. For that reason, intractable transitions in practice are labeled by (copies of) a finite number of time functions $f_i^0(n)$. Therefore it is fair to assume that the intractable transitions of the system are *uniformly intractable* in the following sense.

Definition 3. *The intractable transitions are called* uniformly intractable *if for each $M > 0$, there exists a n_0 such that for every intractable transition $q_1 \xrightarrow{f(n)} q_2$, we have $\forall n \geq n_0$, $\frac{\log f(n)}{\log n} \geq M$.*

Under this assumption, the same proof as before now provides the expected generalization of the reduction theorem.

Proposition 2. *Let $T = (Q, q^0, \delta)$ be a transition system with computation times. Suppose that every transition is either tractable or intractable, and that the intractable transitions are uniformly intractable. Then a state q is reachable in polynomial time if and only if there exists a path $\gamma : q^0 = q_0 \xrightarrow{f_1(n)} q_1 \xrightarrow{f_2(n)} \ldots \xrightarrow{f_p(n)} q_p = q$ such that every $f_i(n)$ $(1 \leq i \leq p)$ is polynomial in n.*

4 Transition systems with probabilistic computation times

In the previous section, we have shown how to account for intractable operations in a Dolev-Yao model with deterministic computation times. In practice yet, algorithms may be probabilistic, and it is more relevant to consider the class of tractable problems to be random polynomial-time rather than polynomial-time. By random polynomial-time algorithms we mean here polynomial-time algorithms using a random oracle, which succeed (give a correct result) with a probability at least $\frac{1}{2}$ and fail (give no result) otherwise. This definition corresponds to the so-called Las Vegas algorithms (see *e.g.* [28]).

As we have been interested in durations previously, it is more natural to state this class in terms of computation time, using the following characterization.

Proposition 3. *A computational problem $\mathcal{P}(x)$ admits a Las Vegas algorithm if and only if there exists an algorithm A which always succeeds in giving an answer to $\mathcal{P}(x)$ within probabilistic time $F(n)$ —where n is the size of the entry x— and such that:*

$$\exists M > 0, \exists N > 0, \exists n_0, \forall n \geq n_0, \mathbb{P}\left(F(n) \leq n^M\right) \geq n^{-N}$$

Proof. The left-to-right implication is clear. Suppose that $\mathcal{P}(x)$ admits an algorithm A satisfying the given property for certain $M > 0$, $N > 0$ and n_0:

$$\forall n \geq n_0, \mathbb{P}\left(F(n) \leq n^M\right) \geq n^{-N}$$

We build a Las Vegas algorithm A' parametrized by $n_1 \geq n_0$ and by a polynomial function $f(n)$ as the following:

- if $n < n_1$, return the correct pre-computed answer (only finitely many entries have a size less than n_1).
- if $n \geq n_1$, execute A on the entry during at most n^M steps, repeat the execution at most $f(n)$ times, or until success.

By construction A' is polynomial-time and succeeds at least with probability:

$$\begin{aligned}\rho_n &= 1 - \left(1 - \mathbb{P}\left(F(n) \leq n^M\right)\right)^{f(n)+1} \\ &\geq 1 - \left(1 - n^{-N}\right)^{f(n)+1}\end{aligned}$$

Using the log function, we see that $\rho_n \to 1$ if $\frac{f(n)+1}{n^N} \to \infty$. We conclude by choosing $f(n) = n^{N+1}$ and n_1 big enough such that $\forall n \geq n_1, \rho_n \geq \frac{1}{2}$. □

4.1 The probabilistic model

We now extend our previous model with probabilities. A *transition system with probabilistic computation times* is a triple $T = (Q, q^0, \delta)$ as before but where the values of the weight function $\delta(n, q_1, q_2)$ are independent random variables over some probabilistic space $(\Omega, \mathcal{A}, \mathbb{P})$ [1]. For simplicity we shall assume that the set of states Q is countable (this is the case for our extended Dolev-Yao models). We write $q_1 \xrightarrow{F(n)} q_2$ to say that $F(n)$ is the random variable such that, for all drawing of lots, $\delta(n, q_1, q_2) = F(n)$.

There remains to define a suitable notion of security. Intuitively a system is secure if for every attacker the probability to reach an unsafe state within a polynomial time is negligible. More precisely let P be a security property, that is the choice of a subset $Q_P \subseteq Q$ of *safe states*. To define a suitable notion of reachability we consider for every fixed $n \in \mathbb{N}$ and $t_0 \geq 0$ a $1\frac{1}{2}$-player game between the attacker and a probabilistic opponent. Such probabilistic nondeterministic systems are also known as Markov decision processes (see *e.g.* [30]). The game $G(Q_P, n, t_0)$ is set up as follows:

[1] This means $\delta(n, q_1, q_2)$ is implicitly a (measurable) function of the drawing of lots $\omega \in \Omega$.

- The attacker begins in the state q^0 with a time zero.
- Let q be the attacker's state and t the time at the beginning of a turn:
  - if $t \leq t_0$ and $q \notin Q_P$ the attacker wins,
  - otherwise the attacker (possibly randomly) chooses a transition $q \xrightarrow{F(n)} q'$ from its current state. The actual value d of $F(n)$ is drawn. The attacker then moves to state q and at time $t + d$.

The goal of the attacker is to reach the set of unsafe states within a fixed amount of time with the highest probability. Since durations are positive numbers, paths that contains cycles are useless for this purpose. So it is fair to assume that every reasonable strategy of the attacker can be described by a function $\sigma : Q \times [0, \infty] \times Q \to [0, 1]$ that gives for a current state q and a clock t the probability $\sigma(q, t, q')$ of choosing q' as the next state. Let us note $\mathbb{P}(G_\sigma(Q_P, n, t_0))$ the probability for the strategy σ to win in the game $G(Q_P, n, t_0)$.

Definition 4. *We will say that P is* verified against every random polynomial attacker *if the probability to reach the unsafe states $Q - Q_P$ within a polynomial time is negligible for every strategy:*

$$\forall M > 0, \forall N > 0, \exists n_0, \forall n \geq n_0, \forall \sigma, \mathbb{P}(G_\sigma(Q_P, n, n^M)) \leq n^{-N}$$

To state our reduction theorem, there remains to define tractable and intractable transitions. To capture the notion of Las Vegas computability, we define tractability as suggested by Proposition 3.

Definition 5. $q_1 \xrightarrow{F(n)} q_2$ *is called* tractable *if*

$$\exists M > 0, \exists N > 0, \exists n_0, \forall n \geq n_0, \mathbb{P}\left(F(n) \leq n^M\right) \geq n^{-N}$$

For the same reason as before, intractability has to be stated in a stronger way than just by negating tractability:

Definition 6. $q_1 \xrightarrow{F(n)} q_2$ *is called* intractable *if*

$$\forall M > 0, \forall N > 0, \exists n_0, \forall n \geq n_0, \mathbb{P}\left(F(n) \leq n^M\right) \leq n^{-N}$$

This definition is satisfactory as it matches the classical definitions of cryptographic security that require the probability of *e.g.* successfully inverting a one-way function in probabilistic non-polynomial time to be *negligible.* For infinite systems, as in the deterministic case, we have to introduce the notion of *uniform intractability* and require the n_0 above to be chosen uniformly over the intractable transitions.

Definition 7. *The intractable transitions of the system are called* uniformly intractable *if*

$$\forall M > 0, \forall N > 0, \exists n_0, \forall n \geq n_0, \forall q_1 \xrightarrow{F(n)} q_2 \textit{ intractable}, \mathbb{P}\left(F(n) \leq n^M\right) \leq n^{-N}$$

Again this hypothesis is realistic because a finite number of extraordinary rules and primitives is used.

4.2 Reduction theorem for infinite probabilistic graphs

We can now state the corresponding reduction theorem:

Theorem 4. *Let $T = (Q, q^0, \delta)$ be a transition system with probabilistic computation times and P be a security property. Assume Q countable. Suppose that every transition is either tractable or intractable, and that the intractable transitions are uniformly intractable. Then P is verified against every random polynomial attacker if and only if there exists no path $\gamma : q^0 = q_0 \xrightarrow{F_1(n)} q_1 \xrightarrow{F_2(n)} \dots \xrightarrow{F_p(n)} q_p = q$ such that every $F_i(n)$ $(1 \leq i \leq p)$ is tractable and $q \notin Q_P$.*

Proof. The left-to-right implication is obvious (if there exists such a polynomial path, one can define a strategy that follows it). Suppose that P is not verified: there exists $M > 0$ and $N > 0$ such that: $\forall n_0, \exists n \geq n_0, \exists \sigma,\ \mathbb{P}(G_\sigma(Q_P, n, n^M)) > n^{-N}$. Since intractable transitions are uniformly intractable, we deduce there exist n and σ such that: $\mathbb{P}(G_\sigma(Q_P, n, n^M)) > n^{-N}$ and for every intractable transition $q_1 \xrightarrow{F(n)} q_2$, $\mathbb{P}\left(F(n) \leq n^M\right) \leq n^{-N}$.

Before we proceed we have to express more precisely the probability of gain $\mathbb{P}(G_\sigma(Q_P, n, n^M))$. To do so let $p(q, t, k)$ be the probability for σ to win in the current game from the state q and the time t in at most k steps. Using the fact that the drawings are independent from each other, the definitions of the game and the strategy σ imply that:

- $p(q, t, k) = 1$ if $t \leq n^M$ and $q \notin Q_P$,
- otherwise, $p(q, t, 0) = 0$ and the probability to win in $(k + 1)$ steps can be written as a sum of the probability to win in k steps conditioned by the choice of the attacker and the drawing of the next duration:

$$p(q, t, k+1) = \sum_{q \xrightarrow{F(n)} q'} \sigma(q, t, q') \int_\Omega p(q', t + F(n)(\omega), k)\ d\mathbb{P}(\omega) \tag{4.1}$$

Now we can rewrite the probability of gain as:

$$\mathbb{P}(G_\sigma(Q_P, n, n^M)) = \sup_{k \in \mathbb{N}} p(q^0, 0, k)$$

Since $p(q^0, 0, k)$ is monotone in k, the hypothesis implies that there exists a k_0 such that $p(q^0, 0, k_0) > n^{-N}$. We prove the auxiliary lemma:

Lemma 5. *For all q, t, k such that $p(q, t, k+1) > n^{-N}$ we have $t \leq n^M$ and*

$$\begin{array}{ll} \textit{either} & q \notin Q_P \\ \textit{or} & \exists q \xrightarrow{F(n)} q', \exists t' \geq t, \mathbb{P}(F(n) \leq n^M) > n^{-N} \text{ and } p(q', t', k) > n^{-N} \end{array}$$

Using this lemma it is straightforward to prove by recurrence on k_0 that there exists a path from q^0 that leads to a state $q \notin Q_P$ and of which every transition $q \xrightarrow{F(n)} q'$ satisfies $\mathbb{P}(F(n) \leq n^M) > n^{-N}$ and thus is tractable.

Let us now proceed and prove the auxiliary lemma. First we note that $t > n^M$ always implies $p(q, t, k) = 0$ so $t \leq n^M$. Second, in the case where $q \in Q_P$ we use Formula 4.1. This quantity is greater than n^{-M} by assumption. By definition of σ, we have $\sum_{q'} \sigma(q,t,q') = 1$. Thus for the inequality to hold, we must have for some transition $q \xrightarrow{F(n)} q'$:

$$\int_\Omega p(q', t + F(n)(\omega), k) \; d\mathbb{P}(\omega) > n^{-N}$$

But since $p(q,t,k) = 0$ whenever $t > n^M$, this integral is also bounded from above by:

$$\left(\sup_{t' \geq t} \; p(q', t', k) \right) \; \mathbb{P}\left(F(n) \leq n^M \right)$$

As the two factors are not greater than 1, they must be both greater than n^{-N}. □

5 Conclusion

A recent and important trend in security protocol verification is to try and relate the computational models of security, based on networks of probabilistic polynomial-time Turing machines, and the formal ones, based on ideas originating from Dolev and Yao [11]. While the former are precise, the latter offer potential for automated verification. Although these two families of models are very far apart, Abadi and Rogaway [3] were the first to find a connection, in specific cases. This was then extended by several authors. This trend currently culminates with Backes et al.'s work [6], whose simulatability result states that every random polynomial-time attack on the real cryptosystem must have a counterpart in a corresponding idealized protocol, unless the cryptographic primitives are broken. The proofs are technical, and the result rests on some specific assumptions.

In this work, we took the opposite route, and we answered the question: how much do Dolev-Yao style models really prove? We first noticed that, from a sufficiently abstract perspective, Dolev-Yao style models were just transition systems, possibly infinite. Extending these transition systems with computation times and probabilities is natural, and makes it possible to account for probabilistic polynomial-time computations, in the Las Vegas sense. Informally, our main contribution is to show that, if there is any attack in the latter, extended model, then some unsafe state was already reachable in the initial Dolev-Yao model, where only tractable transitions are kept. This applies to synchronous as well as asynchronous models, to static or adaptive models. Also, compared to previous work, our model is conceptually simpler and the proofs are short. On the other hand, we do assume that the intruder capabilities can be accounted by a probabilistic nondeterministic transition system (Markov decision process) where transitions are either tractable or uniformly intractable as a function of the security parameter. We believe that this is reasonably acceptable assumption.

Our approach applies to any security property that can be expressed as random polynomial-time (Las Vegas) unreachability. This includes secrecy, various forms of authentication [4], but also more sophisticated requirements, such as those found in e-commerce protocols [8]. An interesting avenue is whether this can be extended to more complex properties, not expressible by reachability, such as those used in fair contract signing [17], which cannot even be expressed in say, linear time temporal logic, but profit from game semantics.

Acknowledgements. We are grateful to Jean Goubault-Larrecq and Florent Jacquemard for comments on earlier versions of the paper, to Thierry Cachat and Stéphane Messika for interesting talks about probabilistic games.

References

[1] M. Abadi and A. D. Gordon. A calculus for cryptographic protocols: The Spi calculus. In *Proc. 4th ACM Conference on Computer and Communications Security (CCS)*, pages 36–47, 1997.

[2] M. Abadi and J. Jürjens. Formal eavesdropping and its computational interpretation. In *Proc. 4th International Symposium on Theoritical Aspects of Computer Software (TACS)*, volume 2215 of *Lecture Notes in Computer Science*, pages 82–94, 2001.

[3] M. Abadi and P. Rogaway. Reconciling two views of cryptography (the computational soundness of formal encryption). In *Proc. 1st IFIP International Conference on Theoretical Computer Science (IFIP–TCS)*, volume 1872 of *Lecture Notes in Computer Science*, pages 3–22, 2000.

[4] R. M. Amadio and D. Lugiez. On the reachability problem in cryptographic protocols. In *Proc. 11th International Conference on Concurrency Theory (CONCUR)*, volume 1877 of *Lecture Notes in Computer Science*, pages 380–394, 2000.

[5] M. Backes and B. Pfitzmann. A cryptographically sound security proof of the Needham-Schroeder-Lowe public-key protocol. In *Proc. 23rd Conference on Foundations of Software Technology and Theoretical Computer Science (FST–TCS)*, volume 2914 of *Lecture Notes in Computer Science*, pages 1–12, 2003.

[6] M. Backes, B. Pfitzmann, and M. Waidner. A composable cryptographic library with nested operations. In *Proc. 10th ACM Conference on Computer and Communications Security (CCS)*, 2003.

[7] B. Blanchet. An efficient cryptographic protocol verifier based on Prolog rules. In *Proc. 14th IEEE Computer Security Foundations Workshop (CSFW)*, pages 82–96, 2001.

[8] D. Bolignano. Towards the formal verification of electronic commerce protocols. In *Proc. 10th IEEE Computer Security Foundations Workshop (CSFW)*, pages 113–147, 1997.

[9] Y. Chevalier, R. Küsters, M. Rusinowitch, and M. Turuani. Deciding the security of protocols with Diffie-Hellman exponentiation and products in exponents. In *Proc. 23rd Conference on Foundations of Software Technology and Theoretical Computer Science (FST–TCS)*, volume 2914 of *Lecture Notes in Computer Science*, pages 124–135, 2003.

[10] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Society*, 22(6):644–654, 1976.

[11] D. Dolev and A. C. Yao. On the security of public key protocols. *IEEE Transactions on Information Theory*, IT-29(12):198–208, 1983.

[12] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28:270–299, 1984.

[13] S. Goldwasser, S. Micali, and R. L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, 1988.

[14] J. Goubault-Larrecq and K. N. Verma. Alternating two-way AC-tree automata. Technical report, LSV, CNRS UMR 8643, ENS Cachan, 2002.

[15] J. Herzog. The Diffie-Hellman key-agreement scheme in the strand-space model. In *Proc. 16th IEEE Computer Science Foundations Workshop (CSFW)*, pages 234–247, 2003.

[16] A. Huima. Efficient infinite-state analysis of security protocols. In *Proc. FLOC Workshop on Formal Methods and Security Protocols*, 1999.

[17] S. Kremer and J.-F. Raskin. Game analysis of abuse-free contract signing. In *Proc. 15th IEEE Computer Security Foundations Workshop (CSFW)*, 2002.

[18] P. Laud. Sound computational interpretation of formal encryption with composed keys. In *Proc. 6th International Conference on Information Security and Cryptology (ICISC)*. KIISC, 2003.

[19] P. Lincoln, J. C. Mitchell, M. Mitchell, and A. Scedrov. A probabilistic poly-time framework for protocol analysis. In *Proc. 5th ACM Conference on Computer and Communications Security (CCS)*, pages 112–121, 1998.

[20] G. Lowe. An attack on the Needham-Schroeder public-key authentication protocol. *Information Processing Letters*, 56(3):131–133, 1995.

[21] C. Meadows. The NRL protocol analyzer: An overview. *Journal of Logic Programming*, 26(2):113–131, 1996.

[22] C. Meadows. Analysis of the Internet Key Exchange protocol using the NRL protocol analyzer. In *Proc. IEEE Symposium on Security and Privacy*, pages 216–231, 1999.

[23] D. Micciancio and B. Warinschi. Completeness theorems for the Abadi-Rogaway logic of encrypted expressions. *Journal of Computer Security*, 2003.

[24] J. Millen and V. Shmatikov. Symbolic protocol analysis with products and Diffie-Hellman exponentiation. In *Proc. 16th IEEE Computer Security Foundations Workshop (CSFW)*, pages 47–61, 2003.

[25] J. K. Millen and V. Shmatikov. Constraint solving for bounded-process cryptographic protocol analysis. In *Proc. 8th ACM Conference on Computer and Communications Security (CCS)*, pages 166–175, 2001.

[26] J. C. Mitchell, M. Mitchell, and U. Stern. Automated analysis of cryptographic protocols using Murϕ. In *Proc. IEEE Symposium on Security and Privacy*, pages 141–153, 1997.

[27] J. C. Mitchell, A. Ramanathan, A. Scedrov, and V. Teague. A probabilistic polynomial-time calculus for analysis of cryptographic protocols. In *Proc. 17th Annual Conference on the Mathematical Foundations of Programming Semantics*, Aarhus, Denmark, 2001.

[28] R. Motwani and P. Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.

[29] O. Pereira and J.-J. Quisquater. Security analysis of the Cliques protocols suites. In *Proc 14th IEEE Computer Security Foundations Workshop (CSFW)*, pages 73–81, 2001.

[30] M. L. Puterman. *Markov Decison Processes: Discrete Stochastic Dynamic Programming.* John Wiley & Sons, 1994.

[31] A. Ramanathan, J. C. Mitchell, A. Scedrov, and V. Teague. Probabilistic bisimulation and equivalence for security analysis of network protocols. In *Proc. 7th International Conference on Foundations of Software Science and Computation Structures (FOSSACS)*, volume 2987 of *Lecture Notes in Computer Science*, pages 468–483, 2004.

[32] M. Roger. *Raffinements de la résolution et vérification de protocoles cryptographiques (in French)*. PhD thesis, ENS Cachan, 2003.

[33] K. N. Verma. Two-way equational tree automata for AC-like theories: decidability and closure properties. In *Proc. 14th International Conference on Rewriting Techniques and Applications (RTA)*, volume 2706 of *Lecture Notes in Computer Science*, 2003.

[34] B. Warinschi. A computational analysis of the Needham-Schroeder protocol. In *Proc. 16th IEEE Computer Science Foundations Workshop (CSFW)*, pages 248–262, 2003.

[35] R. Zunino and P. Degano. A note on the perfect encryption assumption. In *Proc. 7th International Conference on Foundations of Software Science and Computation Structures (FOSSACS)*, volume 2987 of *Lecture Notes in Computer Science*, pages 514–528, 2004.