Deciding Secrecy of Security Protocols with Well Structured Transition Systems

Emanuele D'Osualdo
(joint work with Luke Ong and Alwen Tiu)

1 Motivation

Security protocols are distributed programs that are designed to achieve secure communications using cryptography. They are extensively deployed today but their design is notoriously error-prone. In contrast to other safety critical systems, a distinctive feature of the security properties of protocols is that they must hold in the presence of an adversary or intruder, and this makes them challenging to verify. An important example of such a security property is *secrecy*: to verify that a protocol satisfies secrecy amounts to checking whether it can leak a given (secret) message to the environment as a result of interference by the intruder.

In essence, to verify secrecy, we need a way of analysing the set of messages that the intruder knows; if a message does not belong to this set, then that message is not leaked. Here we assume a model of intruders as defined by Dolev and Yao [4]. The difficulty is that this set of messages is in general infinite, because the Dolev-Yao intruder is in control of three sources of infinity: a) messages of unbounded size, b) an unbounded set of nonces (and other freshly generated data such as session keys), and c) an unbounded number of sessions. Indeed the secrecy problem was proved to be undecidable by Durgin et al. [7]. Amadio et al. [2] and Heintze and Tygar [11] showed that the problem is undecidable even if the set of atomic terms is fixed and finite, assuming that terms of arbitrary size can be substituted for nonces.

In [6], we show that the problem of secrecy is decidable for a class of security protocols with an unbounded number of sessions and unlimited fresh data. Roughly speaking, we show that the inability of the protocol to generate what we call *encryption chains* of unbounded length is a sufficient condition to guarantee decidability of secrecy. An encryption chain of length n is a set of messages of the form $\{N_1\}_{N_2}, \{N_2\}_{N_3}, \ldots, \{N_{n-1}\}_{N_n}$, for secret nonces $N_1, \ldots, N_n$.

Besides the obvious applications, this result's relevance is chiefly conceptual: we show, for the first time, that limiting the length of the encryption chains is enough, together with the necessary message size limitation common to other approaches, to obtain decidability of secrecy. As importantly, we show how to technically implement the restriction so that only the relevant encryption chains are considered.

In this abstract we will outline the main techniques used to prove this result.

2 Modelling Protocols

We use a process calculus with symmetric encryption, based on the π-calculus [13]. As is standard, we use an inference system to define what the intruder knows. The message algebra is deliberately chosen as the simplest that illustrates our approach. The techniques used in our decidability result are robust enough to handle asymmetric key cryptographic protocols as well.

Fix an enumerable set of names, ranged over by lowercase letters $a, b, \ldots$; messages have the syntax:

$$M, N, K ::= a \mid (M, N) \mid \{M\}_K$$

where (-,-) denotes the pairing operator and $\{M\}_K$ denotes the symmetric encryption of the message M with encryption key K. For a finite set of messages Γ, we write $\Gamma \vdash M$ if the message M can be derived by only using knowledge available in Γ, that is by pairing, projecting, encrypting and decrypting with derivable keys. The *size* of a message is defined as the height of its syntax tree, i.e. $\mathrm{size}(a) := 1$ and $\mathrm{size}((M, N)) := \mathrm{size}(\{M\}_N) := 1 + \max(\mathrm{size}(M), \mathrm{size}(N))$.

In our calculus a protocol is modelled by a finite set of (possibily recursive) definitions of processes of the form $\mathsf{Q}[\vec{x}] := A$ where

$$A ::= \mathbf{in}(\vec{x} : M).P \mid A + A \qquad \text{action}$$
$$P ::= \mathbf{0} \mid \nu x.P \mid P \parallel P \mid \langle M \rangle \mid \mathsf{Q}[\vec{x}] \qquad \text{process}$$

The initial state of a protocol is then specified by an intial process P_0. The terms of this grammar can mostly be interpreted as in the Applied π-calculus. A process $\langle M \rangle$ represents the transmission of the message M over an insecure channel: M can be fetched by the intended recipient, but can also be intercepted and manipulated by the intruder. An input $\mathbf{in}(\vec{x} : M).P$ indicates the ability of consuming from the insecure communication channel a message that matches M with free variables $\vec{x}$. A message N matches the pattern $\vec{x} : M$ if there is a substitution θ of the names $\vec{x}$ such that $N = M\theta$. When a matching message is consumed, the input process continues by executing $P\theta$. In this model, pattern matching implements decryption $(x : \{x\}_k)$ and projection $(x, y : (x, y))$.

The interference of the intruder is embedded in the semantic rule. Intuitively, transitions have the form

$$\nu\vec{a}.\big(\langle\Gamma\rangle \parallel \mathbf{in}(\vec{x} : M).P \parallel \dots\big) \to \nu\vec{a}.\nu\vec{c}.\big(\langle\Gamma\rangle \parallel P\theta \parallel \dots\big)$$

if there is a message N that can be derived from the public knowledge—plus some names $\vec{c}$ that may be introduced by the intruder—i.e. $\Gamma, \vec{c} \vdash N$, that matches the input pattern $N = M\theta$. The matching message N may be a legitimate message from some principal, as well as one fabricated by the intruder.

2.1 Knowledge Congruence and Depth

A technical innovation crucial to our decidability proof is the internalisation of the inference system as a congruence relation of the process terms, called *knowledge congruence* ($\equiv_{\mathsf{kn}}$). By construction $\equiv_{\mathsf{kn}}$ contains the standard structural congruence; further it is sound and complete with respect to derivability, i.e., $\Gamma \vdash M$ if, and only if, $\langle\Gamma\rangle \equiv_{\mathsf{kn}} \langle\Gamma\rangle \parallel \langle M\rangle$. Every finite set of messages is knowledge-equivalent to a unique irreducible set of messages. Further, every process is knowledge congruent to a process in irreducible standard form.

Our decidability result relies on two process measures, and their corresponding notions of boundedness. The first is a the maximum size of messages occurring in a process. The second measure is *depth*, an adaptation of a concept introduced by Meyer [12] for the π-calculus. A subterm of a process has nesting of restriction $k \in \mathbb{N}$ just if it is in the scope of k restrictions; the nesting of restriction of a process is just the maximum nesting of restriction of its subterms. The *depth* of a process is then defined as the minimal nesting of restrictions in its *knowledge* congruence class.

For example, consider the process $P = \nu a, b, c.(\langle a\rangle \parallel \langle\{b\}_a\rangle \parallel \langle\{c\}_b\rangle \parallel \langle c\rangle)$ which has nesting of restrictions 3; by using standard structural congruence, we can obtain the equivalent term $Q = \nu b.\big(\nu a.\big(\langle a\rangle \parallel \langle\{b\}_a\rangle\big) \parallel \nu c.\big(\langle\{c\}_b\rangle \parallel \langle c\rangle\big)\big)$ with nesting of restriction 2. Using structural congruence alone does not improve this lower bound, but if one admits knowledge congruence, the term P can be transformed to the knowledge equivalent irreducible process $P' = (\nu a.\langle a\rangle \parallel \nu b.\langle b\rangle \parallel \nu c.\langle c\rangle)$ with nesting of restrictions 1, from which it follows that $\mathrm{depth}(P) = 1$.

3 Main result

Given $s, k \in \mathbb{N}$, we say that a process is (s, k)-*bounded* if all processes reachable from it have depth at most k, when only messages of size up to s are allowed. To understand what bounding depth means intuitively, assume y is a secret name and consider processes of the form $EC_n = \nu x_1, \dots, x_n.\langle\{x_1\}_{x_2}, \{x_2\}_{x_3}, \cdots, \{x_n\}_y\rangle$ which we call an *encryption chain* of length n. The depth of a process containing EC_n and not revealing y will be at least $\lceil \log_2(n) \rceil$. As a result of any interaction—be it honest or malicious—involving only messages of size at most s, a (s, k)-bounded protocol cannot produce encryption chains of unbounded length.

Our main result says that, restricted to messages of up to a given size s, secrecy is decidable for all (s, k)-bounded processes, for all $k \in \mathbb{N}$.

The decidable fragment of security protocols that we have identified captures many real-world symmetric key protocols, including Needham-Schroeder Symmetric Key, Otway-Rees, and Yahalom. Classes of protocols defined in related work [10, 3], are incomparable to ours.

Our proof of decidability is an application of the theory of well-structured transition system (WSTS) [9, 1]. Recall that the coverability problem of an effective WSTS is decidable. The main technical argument lies in the proof that with respect to the process reduction relation, and a notion of *knowledge embedding*, the set of processes reachable from a given (s, k)-bounded process forms an effective WSTS. Secrecy queries are then encoded as appropriate instances of coverability.

Knowledge embedding $\sqsubseteq_{\mathsf{kn}}$ relates two processes $P \sqsubseteq_{\mathsf{kn}} Q$ if $P \equiv \nu\vec{x}.(\langle\Gamma\rangle \parallel P')$ and $Q \equiv \nu\vec{x}.\nu\vec{y}.(\langle\Gamma'\rangle \parallel P' \parallel Q')$ with $\forall M : \Gamma \vdash M \implies \Gamma' \vdash M$. That is, if Q contains all the processes of P and the public (leaked) messages of P can all be derived from the public messages of Q. We prove this order to be a simulation, and a wqo for (s, k)-bounded processes.

4 Algorithmic Aspects

From a more practical perspective, there is a gap to fill to be able to apply our results. This is mainly due to two problems:

1. The complexity of the algorithm extracted from the decidability proof is very high[1] and the procedure suffers from unnecessary state-explosion.
2. Determining if a protocol is (s, k)-bounded is not easy; although it is decidable, the complexity is as bad as checking secrecy.

Regarding item 1, the decision procedure works by symbolically exploring the configuration space backwards from a state representing a leak. After saturating, if the initial state of the protocol is not a member of the fixpoint, then the protocol is proven secret. The main bottleneck of the algorithm is the fact that by exploring the space backwards one is bound to potentially consider a big number of configurations that are not actually reachable (nor coverable) from the initial configuration.

A way to fix this problem comes from the theory of ideal completions for WSTS [8]: we are currently developing this theory for the case of (s, k)-bounded protocols. This alternative approach offers two main advantages: a) the exploration is done in a forward manner, thus considering only configurations that are actually reachable; b) the theory requires the development of symbolic representations of sets of configurations that are useful to reason about protocols in a more general sense.

5 Future Directions

The result outlined in this abstract opens up a number of new directions for secrecy verification.

First, one can try to refine the boundedness condition to obtain a larger class of decidable processes. The fact that the current class is incomparable with classes from related work is a hint that the condition can be improved.

Second, it would be interesting to adapt the type system for depth-boundedness in the π-calculus introduced in [5] to statically check (s, k)-boundedness.

Finally, an implementation of the forward algorithm based on ideal completions could provide support for general and effective semi-automated proofs of secrecy.

[1] While the precise complexity class is not known, we know the problem is non-primitive recursive

References

[1] P. A. Abdulla, K. Cerans, B. Jonsson, and Y. Tsay. "Algorithmic Analysis of Programs with Well Quasi-ordered Domains". In: *Inf. Comput.* 160.1-2 (2000), pp. 109–127.

[2] R. M. Amadio, D. Lugiez, and V. Vanackère. "On the symbolic reduction of processes with cryptographic functions". In: *Theor. Comput. Sci.* 290.1 (2003), pp. 695–740.

[3] R. Chrétien, V. Cortier, and S. Delaune. "Decidability of Trace Equivalence for Protocols with Nonces". In: *CSF 2015*. 2015, pp. 170–184.

[4] D. Dolev and A. C. Yao. "On the security of public key protocols". In: *IEEE Trans. Information Theory* 29.2 (1983), pp. 198–207.

[5] E. D'Osualdo and C.-H. L. Ong. "On hierarchical communication topologies in the pi-calculus". In: *ESOP*. 2016.

[6] E. D'Osualdo, C.-H. L. Ong, and A. Tiu. "Deciding Secrecy of Security Protocols for an Unbounded Number of Sessions: The Case of Depth-bounded Processes". In: *CSF*. 2017.

[7] N. A. Durgin, P. D. Lincoln, J. C. Mitchell, and A. Scedrov. "Undecidability of bounded security protocols". In: *FMSP*. 1999.

[8] A. Finkel and J. Goubault-Larrecq. "Forward Analysis for WSTS, Part I: Completions". In: *STACS*. Vol. 3. LIPIcs. 2009, pp. 433–444.

[9] A. Finkel and P. Schnoebelen. "Well-structured transition systems everywhere!" In: *Theor. Comput. Sci.* 256.1-2 (2001), pp. 63–92.

[10] S. B. Fröschle. "Leakiness is Decidable for Well-Founded Protocols". In: *POST, held as Part of ETAPS 2015*. 2015, pp. 176–195.

[11] N. Heintze and J. D. Tygar. "A Model for Secure Protocols and Their Compositions". In: *IEEE Trans. Software Eng.* 22.1 (1996), pp. 16–30.

[12] R. Meyer. "On Boundedness in Depth in the π-calculus". In: *IFIP TCS*. Ed. by G. Ausiello, J. Karhumäki, G. Mauri, and C.-H. L. Ong. Vol. 273. IFIP. Springer, 2008, pp. 477–489.

[13] R. Milner, J. Parrow, and D. Walker. "A Calculus of Mobile Processes, I & II". In: *Inf. Comput.* 100.1 (1992), pp. 1–77.