THE APPLIED PI-CALCULUS AND PROVERIF

Tom Chothia

Summary of Yesterday

- Protocol can be complex
- Understanding the right security properties can be difficult.
- Working out if the security properties hold or not is even worse
- pi & spi calculus help us model and understand protocols

Today

- The applied pi-calculus
- Case study: traceability in e-passports
- Proverif: an automated tool for the applied pi calculus
- Case study: a better pacemarker protocol
- Tamarin: another protocol checking tool
- Case study: checking WPA security properites.

The applied pi-calculus

- The applied pi-calculus adds arbitrary functions.
  - Let's us model lots of different crypto.

```
fun senc/2.
reduc sdec(y, senc(y,x)) = x.

fun penc/2. fun pk/1.
reduc pdec(penc(x,pk(y)),y) = x.

hash/1
```

Applied pi calculus Syntax

$M, N ::=$	terms
x, y, z	variables
a, b, c, k, s	names
$f(M_1, \ldots, M_n)$	constructor application
$D ::= g(M_1, \ldots, M_n)$	destructor application
$P, Q ::=$	processes
0	nil
$\overline{M}\langle N\rangle.P$	output
$M(x).P$	input
$P \mid Q$	parallel composition
$!P$	replication
$\nu\, a.P$	restriction
let $x = D$ in P else Q	term evaluation
$t{:}\,P$	phase

$E, \mathcal{P} \cup \{t{:}\,0\} \rightarrow E, \mathcal{P}$ (Red Nil)

$E, \mathcal{P} \cup \{t{:}\,!P\} \rightarrow E, \mathcal{P} \cup \{t{:}\,!P, t{:}\,P\}$ (Red Repl)

$E, \mathcal{P} \cup \{t{:}\,(P \mid Q)\}) \rightarrow E, \mathcal{P} \cup \{t{:}\,P, t{:}\,Q\}$ (Red Par)

$E, \mathcal{P} \cup \{t{:}\,\nu\, a.P\} \rightarrow E \cup \{a'\}, \mathcal{P} \cup \{t{:}\,P\{a'/a\}\}$ (Red Res)
for some name $a' \notin E$

$E, \mathcal{P} \cup \{t{:}\,\overline{N}\langle M\rangle.P, t{:}\,N(x).Q\} \rightarrow E, \mathcal{P} \cup \{t{:}\,P, t{:}\,Q\{M/x\}\}$ (Red I/O)

$E, \mathcal{P} \cup \{t{:}\,\text{let } x = D \text{ in } L\} \rightarrow E, \mathcal{P} \cup \{t{:}\,P\{M/x\}\}$ (Red Destr 1)
if there exists M such that $D \rightarrow M$

$E, \mathcal{P} \cup \{t{:}\,\text{let } x = D \text{ in } P \text{ else } Q\}) \rightarrow E, \mathcal{P} \cup \{t{:}\,Q\}$ (Red Destr 2)
if there is no M such that $D \rightarrow M'$

$E, \mathcal{P} \cup \{t{:}\,t'{:}\,P\} \rightarrow E, \mathcal{P} \cup \{t'{:}\,P\}$ (Red Order)
if $t < t'$

Some Hints on ProVerif

- Might produce false attacks, due to how fresh names are modelled.
- Use the `-graph <dir>` option to get an attempted at an attack diagram.
- You can add types to the ver version
- Lots of options e.g., `set attacker = passive`
- Limited bi-simulation checking.
- Recent improvements, e.g., added state, numbers,…

Case study: E-passport

Analysing Unlinkability and Anonymity Using the Applied Pi Calculus

Myrto Arapinis, Tom Chothia, Eike Ritter and Mark Ryan

School of Computer Science, University of Birmingham, UK
{m.d.arapinis, t.chothia, e.ritter, m.d.ryan}@cs.bham.ac.uk

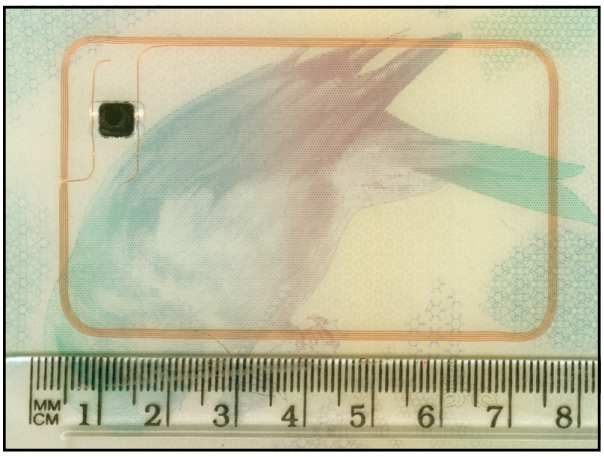
***Abstract*—An attacker that can identify messages as coming from the same source, can use this information to build up a picture of targets' behaviour, and so, threaten their privacy. In response to this danger, unlinkable protocols aim to make it impossible for a third party to identify two runs of a protocol as coming from the same device. We present a framework for analysing unlinkability and anonymity in the applied pi calculus. We show that unlinkability and anonymity are complementary properties; one does not imply the other. Using our framework we show that the French RFID e-passport preserves anonymity but it is linkable therefore anyone carrying a French e-passport can be physically traced.**

I. Introduction

The proliferation of portable computing devices has lead to a range of new computer security problems. Mobile phones, Bluetooth devices and RFID tags have all been shown to leak private information, and such security concerns are regularly

the case where an attacker has observed a system and decides that two particular messages might be from different sessions being performed by the same agent. The system is weakly unlinkable if, for all such cases, there exists another trace of the system, which looks the same to the attacker, and in this other trace the two messages came from different agents. A failure of weak unlinkability directly implies an attack.

We show that our definition of strong unlinkability implies the weaker version. As strong unlinkability can sometimes be checked automatically using the ProVerif tool, this means that when checking a protocol it is useful to check the strong definition first. If it fails, one may go on to use the weaker version to look for a practical attack. We also show that unlinkability does not imply anonymity, contradictory to what has been suggested by other authors [13]. An example of a device that would be unlinkable but not anonymous is an



e-Passport

e-Passports contain an RFID tag.

It stores picture, personal details, and in some cases fingerprints.

Believed to be readable from up to 50cm, can be eavesdropped on from greater distance.

Read access is protected with a key based on the DoB, DoI and passport number.

Basic Access Control

Reader — Passport

--- GET CHALLENGE →

Pick random N_P

← N_P ---

Pick random N_R,K_R

--- $\{N_R,N_P,K_R\}_{Ke}$,$MAC_{Km}(\{N_R,N_P,K_R\}_{Ke})$ →

Check MAC, Decrypt,
Check N_P Pick random K_P

← $\{N_P,N_R,K_P\}_{Ke}$,$MAC_{Km}(\{N_P,N_R,K_P\}_{Ke})$ --

Check MAC, Decrypt,
Check N_R

Error Messages

For a complete model we must also include the error messages from the passport.

Each country has a different implementation of the passport.

We use the French passport as an example.

Basic Access Control

Reader — Passport

--- GET CHALLENGE →

Pick N_P

←------ N_P ---------

Pick N_R,K_R

--- $\{N_R,N_P,K_R\}_{Ke}$,MAC(X) →

Check MAC,
Check Fails

← 6300 --

Basic Access Control

Reader — Passport

--- GET CHALLENGE →

Pick N_P

←------ N_P ---------

Pick N_R,K_R

--- $\{N_R,X,K_R\}_{Ke}$,$MAC_{Km}(\{N_R,X,K_R\}_{Ke})$ →

Check MAC,
Nonce Check Failed

← 6A80 --

Traceability

- The passport is designed to be untraceable.
- I.e., you should not be able to tell if two runs of the protocol are from the same passport.
- An excellent privacy property for protocol, but with no formal definition.

Well formed protocols

Definition 5 (Well-formed Protocol): A p-party protocol is said to be *well-formed* if it is a closed plain process P of the form:

$$\begin{array}{rcl} P & = & \nu\tilde{n}.\ (!R_1 \mid \ldots \mid !R_p) \\ \forall i \in \{1,\ldots,p\}\ R_i & = & \nu id.\ \nu\tilde{m}.\ init_i.\ !(\nu s.\ main_i) \end{array}$$

Defining Unlinkability

Definition 12 (Strong unlinkability): Let Σ be a signature and E an equational theory for this signature, and let P be a well-formed p-party protocol over Σ of the form described at Definition 5. For all $i \in \{1,\ldots,p\}$, we build the protocol P^{R_i} over Σ as follows:

$$\begin{array}{rcl} P^{R_i} & \triangleq & \nu\tilde{n}.\ (!R_1 \mid \ldots \mid !R_{i-1} \mid !R_i'' \mid !R_{i+1} \mid \ldots, \mid !R_p) \\ R_i'' & \triangleq & \nu id.\ \nu\tilde{m}.\ init_i.\ main_i \end{array}$$

P preserves *strong unlinkability* of R_i if

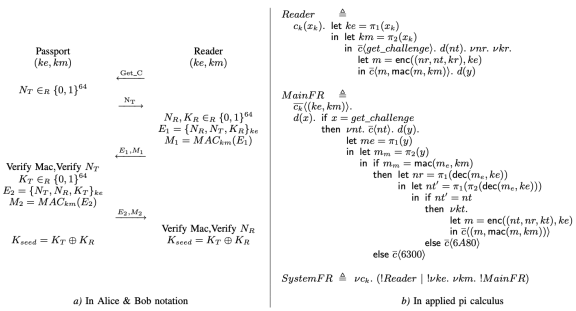
$$P \approx_l P^{R_i}$$

new c.(!Reader | !new tagID.!Tag) =
new c.(!Reader | !new tagID.Tag)

Why strong?

$$\begin{array}{rcl} P & \triangleq & \nu c_{pv}.\ (!R \mid !T) \\ R & \triangleq & !(\nu s.\ c_{pv}(x).\ c_{pv}(y).\ \text{if } x = y \text{ then } \overline{c_{pb}}\langle beep\rangle) \\ T & \triangleq & \nu id.\ !(\nu s.\ \overline{c_{pv}}\langle id\rangle) \end{array}$$

This process fails strong unlikability by the attacker can't really trace any tags.



Attack Part 1

Attacker eavesdrops on Alice using her passport

Reader Alice

--- GET CHALLENGE →

Pick random N_P

←------ N_P ---------

Pick random N_R,K_R

--- M = $\{N_R,N_P,K_R\}_{KAe}$$MAC_{KAm}(\{N_R,N_P,K_R\}_{KAe})$ →

Attack records message M.

Attack Part 2

Attacker ????

--- GET CHALLENGE →

Pick random N_{P2}

←------ N_{P2} ---------

--- $\{N_R,N_P,K_R\}_{KAe}$,$MAC_{KAm}(\{N_R,N_P,K_R\}_{KAe})$ →

MAC check fails

← 6300 --

It's a different passport

Attack Part 2

Attacker ????

--- GET CHALLENGE →

Pick random N_{P2}

←------ N_{P2} ---------

--- $\{N_R,N_P,K_R\}_{KAe}$,$MAC_{KAm}(\{N_R,N_P,K_R\}_{KAe})$ →

MAC passess, nonce fails

← 6A80 --

It's the same passport as before.

Did the Tool Find the Attack?

No, it occurred to us when we saw the different error messages.

But we would not have been looking for these without the need to make a formal model for testing.

The formal modelling process did find the attack.

Real value of the work was a formal testing method for traceability.

DISCOVERING EPASSPORT VULNERABILITIES USING BISIMILARITY

ROSS HORNE AND SJOUKE MAUW

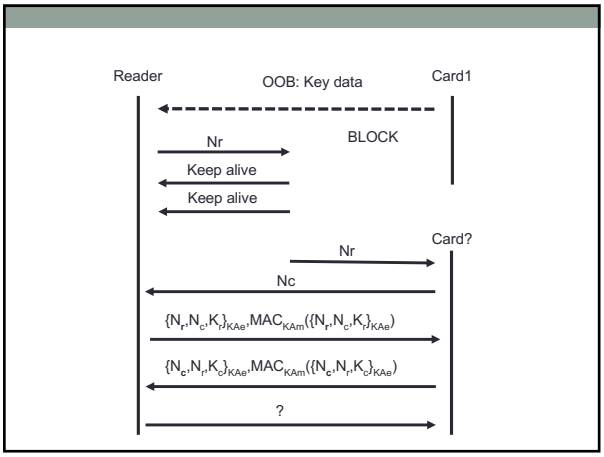
Department of Computer Science, University of Luxembourg, Esch-sur-Alzette, Luxembourg
e-mail address: ross.horne@uni.lu

Department of Computer Science and SnT, University of Luxembourg, Esch-sur-Alzette, Luxembourg
e-mail address: sjouke.mauw@uni.lu

Abstract. We uncover privacy vulnerabilities in the ICAO 9303 standard implemented by ePassports worldwide. These vulnerabilities, confirmed by ICAO, enable an ePassport holder who recently passed through a checkpoint to be reidentified without opening their ePassport. This paper explains how bisimilarity was used to discover these vulnerabilities, which exploit the BAC protocol – the original ICAO 9303 standard ePassport authentication protocol – and remains valid for the PACE protocol, which improves on the security of BAC in the latest ICAO 9303 standards. In order to tackle such bisimilarity problems, we develop here a chain of methods for the applied π-calculus including a symbolic under-approximation of bisimilarity, called open bisimilarity, and a modal logic, called classical $\mathcal{FM}$, for describing and certifying attacks. Evidence is provided to argue for a new scheme for specifying such unlinkability problems that more accurately reflects the capabilities of an attacker.

1. Introduction

Most of us have the option to pass through automatic passport clearance at an airport. Some of us also have electronic national cards that may be used for government services. All of these machine readable documents employ a protocol to authenticate with a reader, establishing that you really hold a valid machine readable document. In order for ePassports to be read internationally, your passport



Case Study 2: Pacemakers

On the (in)security of the Latest Generation Implantable Cardiac Defibrillators and How to Secure Them

Eduard Marin
ESAT-COSIC and iMinds
KU Leuven, Belgium
eduard.marin@esat.kuleuven.be

Dave Singelée
ESAT-COSIC and iMinds
KU Leuven, Belgium
dave.singelee@esat.kuleuven.be

Flavio D. Garcia
School of Computer Science
University of Birmingham, UK
f.garcia@bham.ac.uk

Tom Chothia
School of Computer Science
University of Birmingham, UK
t.p.chothia@cs.bham.ac.uk

Rik Willems
Cardiology, University Hospital Gasthuisberg
Leuven, Belgium
rik.willems@uzleuven.be

Bart Preneel
ESAT-COSIC and iMinds
KU Leuven, Belgium
bart.preneel@esat.kuleuven.be

ABSTRACT

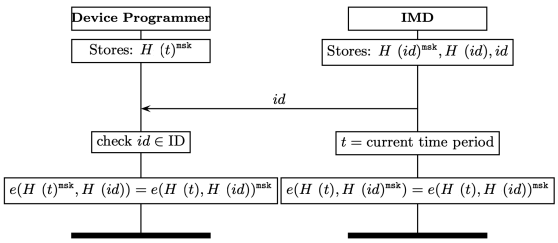
Implantable Medical Devices (IMDs) typically use proprietary protocols with no or limited security to wirelessly communicate with a device programmer. These protocols enable doctors to carry out critical functions, such as changing the IMD's therapy or collecting telemetry data, without having to perform surgery on the patient. In this paper, we fully reverse-engineer the proprietary communication protocol between a device programmer and the latest generato monitor and help control abnormal heart rhythms. ICDs are battery-powered devices that deliver electric shocks to the patient's heart if the heartbeat is too fast. Some ICDs can also act as a pacemaker and give tiny electrical shocks if the heartbeat is too slow. ICDs have evolved over three generations. The first generation (or the oldest) do not have any wireless interface and hence do not allow reprogramming once the ICD is implanted. The second and third generation enable wireless communication with external devices includ-

Proposed Fix

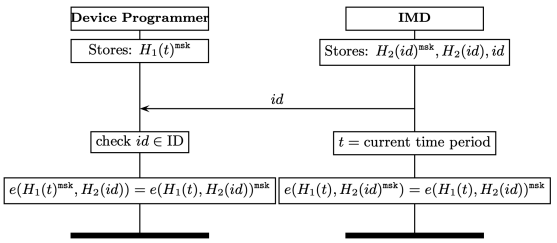
- Readers turn up on e-bay so should not have long term keys.
- Readers may lose Internet access at when urgent care is needed.
- Idea: pacemakers will generate a new key every 3 months. Readers must be updated once every 3 months to get this key.
- Based on bilinear pairings, a functions e such that:

$$e(g^a, h^b) = e(g, h)^{ab}$$

Semi-Offline Key Agreement



Semi-Offline Key Agreement Fixed!



Ways to use protocol checking in your own work:

- If you propose a new protocol in a paper include a formal model include a formal model.
- This gives a precise definition of the protocol and security properties.
- Help remove any referees/readers concerns about your work protocol.
- It might also find a weakness that you can fix!