Securing statically-verified communications protocols against timing attacks

Stephen Gilmore
LFCS, Edinburgh

Joint work with Mikael Buchholtz,
Jane Hillston and Flemming Nielson

Outline

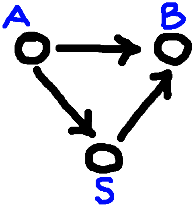
Outline

1. Secure communications protocols
   - Timing attacks
   - Remote timing attacks
2. Static analysis for security properties using LySa
   - LySa process calculus
   - LySa model of the Wide-Mouthed Frog protocol
   - Static analysis with the LySatool
3. Dynamic analysis for performance properties using PEPA
   - The PEPA stochastic process algebra
   - PEPA model of the Wide-Mouthed Frog protocol
   - Dynamic analysis with IPC/DNAmaca
4. Summary

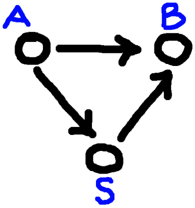
Purpose of a secure communications protocol

- Allows sender and receiver to exchange confidential messages.
- Authenticates the principals to confirm their identity.

An example: the Wide-Mouthed Frog protocol

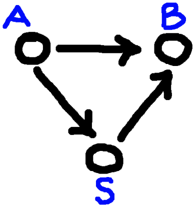


An example: the Wide-Mouthed Frog protocol



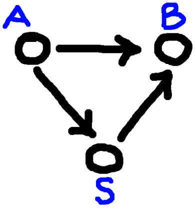
1. A sends a message to S including the name of B and the new session key K_{AB}, encrypted under K_{AS}.

An example: the Wide-Mouthed Frog protocol



1. A sends a message to S including the name of B and the new session key K_{AB}, encrypted under K_{AS}.
2. S decrypts this and sends the name of A and the key K_{AB} to B, encrypted under K_{BS}.

An example: the Wide-Mouthed Frog protocol



1. A sends a message to S including the name of B and the new session key K_{AB}, encrypted under K_{AS}.
2. S decrypts this and sends the name of A and the key K_{AB} to B, encrypted under K_{BS}.
3. A sends a message to B encrypted under K_{AB}.

Timing attacks

- Attacks where information is leaked through an inference obtained from timing a secure interaction are known as timing attacks.

Timing attacks

- Attacks where information is leaked through an inference obtained from timing a secure interaction are known as timing attacks.
- Secure communications protocols depend on encryption algorithms which take a measureable time to execute.

Timing attacks

- Attacks where information is leaked through an inference obtained from timing a secure interaction are known as timing attacks.
- Secure communications protocols depend on encryption algorithms which take a measureable time to execute.
- If security-sensitive operations can be repeatedly timed then information about the secret keys used for decryption can be gained bit by bit until they are entirely known.

Timing attacks

- Attacks where information is leaked through an inference obtained from timing a secure interaction are known as timing attacks.
- Secure communications protocols depend on encryption algorithms which take a measureable time to execute.
- If security-sensitive operations can be repeatedly timed then information about the secret keys used for decryption can be gained bit by bit until they are entirely known.
- When secret keys become known then the confidentiality and authenticity offered by secure protocols are entirely lost.

Remote timing attacks

- It has been known for some years that timing attacks can be used to extract keys from weak computing devices such as smartcards.

Remote timing attacks

- It has been known for some years that timing attacks can be used to extract keys from weak computing devices such as smartcards.
- It was shown last year that remote timing attacks can be used to uncover secure keys stored on servers. [*Remote timing attacks are practical, Brumley and Boneh, 12th USENIX Security Symposium, 2003*].

Remote timing attacks

- It has been known for some years that timing attacks can be used to extract keys from weak computing devices such as smartcards.
- It was shown last year that remote timing attacks can be used to uncover secure keys stored on servers. [*Remote timing attacks are practical, Brumley and Boneh, 12th USENIX Security Symposium, 2003*].
    - Mounted an attack across multiple routers and switches on an OpenSSL-based web server.

Remote timing attacks

- It has been known for some years that timing attacks can be used to extract keys from weak computing devices such as smartcards.
- It was shown last year that remote timing attacks can be used to uncover secure keys stored on servers. [*Remote timing attacks are practical, Brumley and Boneh, 12th USENIX Security Symposium, 2003*].
  - Mounted an attack across multiple routers and switches on an OpenSSL-based web server.
  - Attack applies in networked, inter-process and virtual machine environments.

Remote timing attacks

- It has been known for some years that timing attacks can be used to extract keys from weak computing devices such as smartcards.
- It was shown last year that remote timing attacks can be used to uncover secure keys stored on servers. [*Remote timing attacks are practical, Brumley and Boneh, 12th USENIX Security Symposium, 2003*].
    - Mounted an attack across multiple routers and switches on an OpenSSL-based web server.
    - Attack applies in networked, inter-process and virtual machine environments.
    - Found that many crypto libraries completely ignore the timing attack.

Outline

LySa process calculus

LySa [*Buchholtz, Nielson and Nielson, 2004*] is a variant of Abadi and Gordon's Spi-calculus which includes pattern matching on values at input and decryption.

LySa process calculus

LySa [*Buchholtz, Nielson and Nielson, 2004*] is a variant of Abadi and Gordon's Spi-calculus which includes pattern matching on values at input and decryption.

$P_1 \mid P_2$	Parallel
$!P$	Replication
0	Nil
$(\nu\, n)\, P$	New
$\langle t_1, \ldots, t_k \rangle.P$	Output
$(t_1, \ldots, t_j;\ x_{j+1}, \ldots x_k).P$	Input
decrypt t as $\{t_1, \ldots, t_j;\ x_{j+1}, \ldots, x_k\}_{t_0}$ in P	Decrypt

Structure of a LySa packet

LySa packet

$$\langle \; \underbrace{A, S,}_{\text{header}} \quad \underbrace{A, \{ \, B, K_{AB} \, \}_{K_{AS}}}_{\text{payload}} \quad \underbrace{[\mathrm{at}\ a1\ \mathrm{dest}\ s1]}_{\text{metadata}} \; \rangle$$

- LySa packets are tuples of information sent across a global network.

Structure of a LySa packet

LySa packet

$$\langle \underbrace{A, S,}_{\text{header}} \quad \underbrace{A, \{\, B, K_{AB} \,\}_{K_{AS}}}_{\text{payload}} \quad \underbrace{[\text{at } a1 \text{ dest } s1]}_{\text{metadata}} \rangle$$

- LySa packets are tuples of information sent across a global network.
- Header: The sender is A and the receiver is S.

Structure of a LySa packet

LySa packet

$$\langle \underbrace{A, S,}_{\text{header}} \quad \underbrace{A, \{ B, K_{AB} \}_{K_{AS}}}_{\text{payload}} \quad \underbrace{[\mathsf{at}\ a1\ \mathsf{dest}\ s1]}_{\text{metadata}} \rangle$$

- LySa packets are tuples of information sent across a global network.
- Header: The sender is A and the receiver is S.
- Payload: The name A is sent in the clear and the name B and the key K_{AB} are sent encrypted under K_{AS}.

Structure of a LySa packet

LySa packet

$$\langle \underbrace{A, S,}_{\text{header}} \quad \underbrace{A, \{\, B, K_{AB} \,\}_{K_{AS}}}_{\text{payload}} \quad \underbrace{[\text{at } a1 \text{ dest } s1]}_{\text{metadata}} \rangle$$

- LySa packets are tuples of information sent across a global network.
- Header: The sender is A and the receiver is S.
- Payload: The name A is sent in the clear and the name B and the key K_{AB} are sent encrypted under K_{AS}.
- Metadata: This is encrypted at $a1$ to be decrypted at $s1$.

LySa model of the Wide-Mouthed Frog protocol

LySa model of the Wide-Mouthed Frog protocol

Principal A

$$!(\nu\, K_{AB})\, \langle A, S, A, \{B, K_{AB}\}_{K_{AS}} [\text{at } a1 \text{ dest } s1\,] \rangle.$$
$$(\nu\, message)\, \langle A, B, \{message\}_{K_{AB}} [\text{at } a2 \text{ dest } b2\,] \rangle.0$$

LySa model of the Wide-Mouthed Frog protocol

Principal A

$$!(\nu\, K_{AB})\, \langle A, S, A, \{B, K_{AB}\}_{K_{AS}}[\text{at } a1 \text{ dest } s1\,]\rangle.$$
$$(\nu\, message)\, \langle A, B, \{message\}_{K_{AB}}[\text{at } a2 \text{ dest } b2\,]\rangle.0$$

Server S

$$!(A, S, A;\, z).\text{decrypt } z \text{ as } \{B;\, zk\}_{K_{AS}}[\text{at } s1 \text{ orig } a1\,] \text{ in}$$
$$\langle S, B, \{A, zk\}_{K_{BS}}[\text{at } s2 \text{ dest } b1\,]\rangle.0$$

LySa model of the Wide-Mouthed Frog protocol

Principal A

$$!(\nu\, K_{AB})\, \langle A, S, A, \{B, K_{AB}\}_{K_{AS}} [\text{at } a1 \text{ dest } s1\,] \rangle.$$
$$(\nu\, message)\, \langle A, B, \{message\}_{K_{AB}} [\text{at } a2 \text{ dest } b2\,] \rangle.0$$

Server S

$$!(A, S, A;\, z).\text{decrypt } z \text{ as } \{B;\, zk\}_{K_{AS}} [\text{at } s1 \text{ orig } a1\,] \text{ in}$$
$$\langle S, B, \{A, zk\}_{K_{BS}} [\text{at } s2 \text{ dest } b1\,] \rangle.0$$

Principal B

$$!(S, B;\, x).\text{decrypt } x \text{ as } \{A;\, xk\}_{K_{BS}} [\text{at } b1 \text{ orig } s2\,] \text{ in}$$
$$(A, B;\, y).\text{decrypt } y \text{ as } \{;\, ym\}_{xk} [\text{at } b2 \text{ orig } a2\,] \text{ in } 0$$

Static analysis with the LySatool

- The LySa processes are annotated with authentication properties specifying intended origin and destinations of messages.

Static analysis with the LySatool

- The LySa processes are annotated with authentication properties specifying intended origin and destinations of messages.
- The LySatool works by computing over-approximations to what a LySa process can do in *all executions* of the process executed in parallel with an arbitrary attacker.

Static analysis with the LySatool

- The LySa processes are annotated with authentication properties specifying intended origin and destinations of messages.
- The LySatool works by computing over-approximations to what a LySa process can do in *all executions* of the process executed in parallel with an arbitrary attacker.
    - The analysis may report too many errors in protocols, but *cannot report too few*.

Static analysis with the LySatool

- The LySa processes are annotated with authentication properties specifying intended origin and destinations of messages.
- The LySatool works by computing over-approximations to what a LySa process can do in *all executions* of the process executed in parallel with an arbitrary attacker.
  - The analysis may report too many errors in protocols, but *cannot report too few*.
  - Reporting too many errors does not pose a big problem in practice.

Static analysis with the LySatool

- The LySa processes are annotated with authentication properties specifying intended origin and destinations of messages.
- The LySatool works by computing over-approximations to what a LySa process can do in *all executions* of the process executed in parallel with an arbitrary attacker.
    - The analysis may report too many errors in protocols, but *cannot report too few*.
    - Reporting too many errors does not pose a big problem in practice.
- The LySatool reports no errors in the WMF protocol.

Outline

Performance Evaluation Process Algebra

PEPA [*Hillston, 1994*] is a stochastic process algebra in which the rate at which each activity can be performed is quantified.

Performance Evaluation Process Algebra

PEPA [*Hillston, 1994*] is a stochastic process algebra in which the rate at which each activity can be performed is quantified.

$(\alpha, r).P$	Prefix
$P_1 + P_2$	Choice
$P_1 \bowtie_L P_2$	Co-operation
P/L	Hiding
X	Variable

PEPA model of the Wide-Mouthed Frog protocol

PEPA model of the Wide-Mouthed Frog protocol

Principal A

$$P_A \stackrel{def}{=} (as, r_{as}).(ab, r_{ab}).P_A$$

PEPA model of the Wide-Mouthed Frog protocol

Principal A

$$P_A \stackrel{\mathit{def}}{=} (as, r_{as}).(ab, r_{ab}).P_A$$

Server S

$$P_S \stackrel{\mathit{def}}{=} (as, \top).(sb, r_{sb}).P_S$$

PEPA model of the Wide-Mouthed Frog protocol

Principal A

$$P_A \stackrel{def}{=} (as, r_{as}).(ab, r_{ab}).P_A$$

Server S

$$P_S \stackrel{def}{=} (as, \top).(sb, r_{sb}).P_S$$

Principal B

$$P_B \stackrel{def}{=} (sb, \top).(ab, \top).P_B$$

Dynamic analysis of PEPA models

- A PEPA model is analysed relative to valuations which map the symbolic rates of the model to concrete values determined by estimation or measurement.

Dynamic analysis of PEPA models

- A PEPA model is analysed relative to valuations which map the symbolic rates of the model to concrete values determined by estimation or measurement.
- Rates can be chosen to represent communication cost, computation cost, or an aggregate of these.

Dynamic analysis of PEPA models

- A PEPA model is analysed relative to valuations which map the symbolic rates of the model to concrete values determined by estimation or measurement.
- Rates can be chosen to represent communication cost, computation cost, or an aggregate of these.
- We can modify the protocol by adding delays to mask the difference between a faster interaction and a slower one.

Dynamic analysis of PEPA models

- A PEPA model is analysed relative to valuations which map the symbolic rates of the model to concrete values determined by estimation or measurement.
- Rates can be chosen to represent communication cost, computation cost, or an aggregate of these.
- We can modify the protocol by adding delays to mask the difference between a faster interaction and a slower one.
- Finally, we wish to determine whether or not two versions of the PEPA model of the protocol are sufficiently close that we would believe that a timing attack is impractical.

The Imperial PEPA Compiler and DNAmaca

- IPC (The Imperial PEPA Compiler) processes PEPA models to compile them into the input format of the DNAmaca Markov chain analyser.

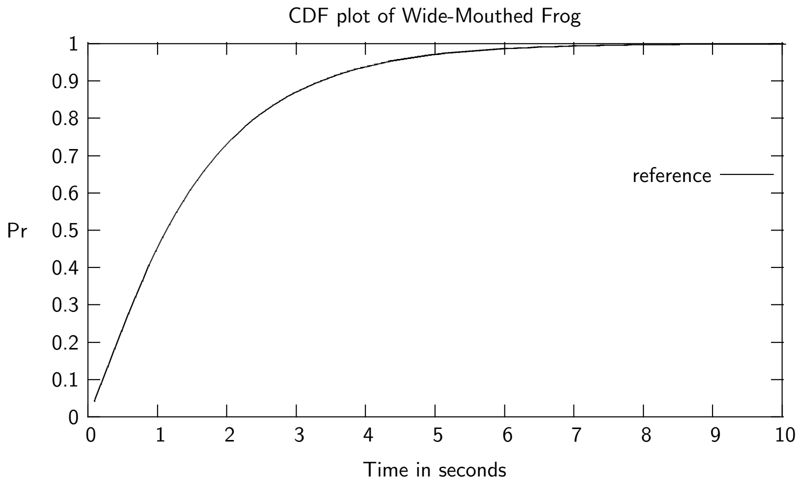
The Imperial PEPA Compiler and DNAmaca

- IPC (The Imperial PEPA Compiler) processes PEPA models to compile them into the input format of the DNAmaca Markov chain analyser.
- IPC allows the modeller to attach stochastic probes to a PEPA model to mark the start and end of passages through the model.

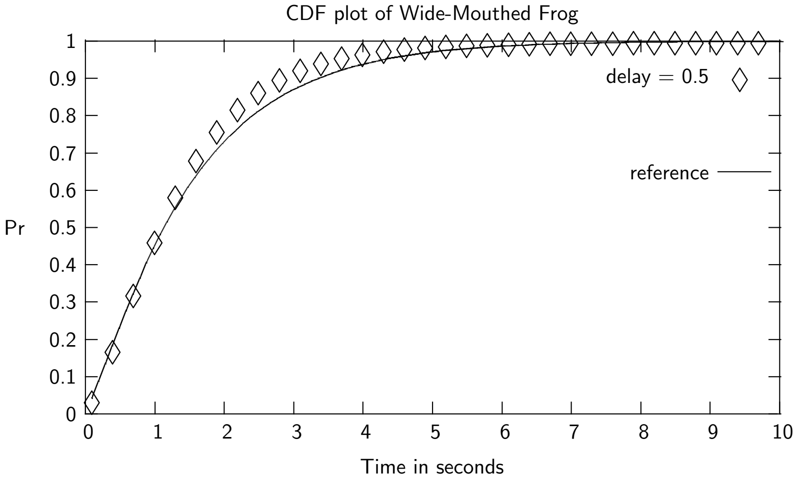
The Imperial PEPA Compiler and DNAmaca

- IPC (The Imperial PEPA Compiler) processes PEPA models to compile them into the input format of the DNAmaca Markov chain analyser.
- IPC allows the modeller to attach stochastic probes to a PEPA model to mark the start and end of passages through the model.
- Via uniformisation, DNAmaca computes passage-time densities for this, allowing them to be presented as a cumulative density function (CDF) for the passage.

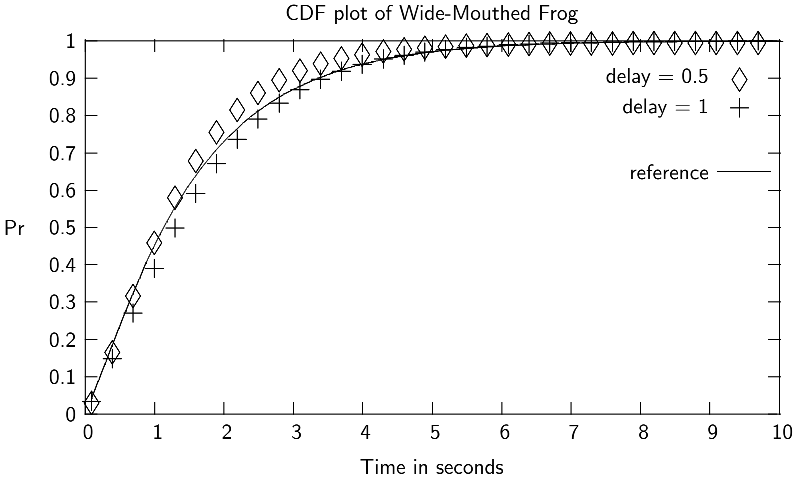
CDF plot of the WMF protocol



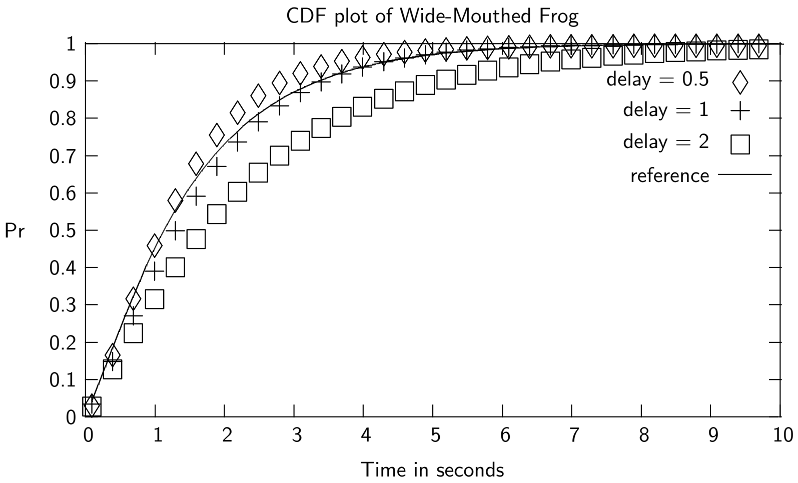
CDF plot of the WMF protocol



CDF plot of the WMF protocol



CDF plot of the WMF protocol



Outline

Summary

- In the design of novel communications protocols it is necessary to consider both security and performance. It is helpful to have a systematic method of analysing protocols with automated support.

Summary

- In the design of novel communications protocols it is necessary to consider both security and performance. It is helpful to have a systematic method of analysing protocols with automated support.
- Security and performance are interrelated issues:
  - Time-dependent behaviour can be used to attack a protocol.
  - Developers who are concerned with achieving peak performance view security measures as an overhead.

Summary

- In the design of novel communications protocols it is necessary to consider both security and performance. It is helpful to have a systematic method of analysing protocols with automated support.
- Security and performance are interrelated issues:
  - Time-dependent behaviour can be used to attack a protocol.
  - Developers who are concerned with achieving peak performance view security measures as an overhead.
- By using the LySatool to check origination and destination of messages and the Imperial PEPA Compiler and DNAmaca for the computation of passage-time quantiles we have been able to guard against certain types of network-based attacks.