

# VALIDATION DE SERVICES WEB

Yannick, Denis et Michael

2007

# OUTLINE

1 INTRODUCTION

2 TREATMENT OF SECURED MESSAGES

3 FINITE-STATE ANALYSIS OF SERVICES

# MESSAGE-LEVEL SECURITY

- Security of data protected off-line by Access Control
  - Web Services : move from off-line to on-line
  - Thus need to protect the data on its trips
- ⇒ Message level security, derived from :
- additional need : authentication of requester, integrity of request, . . .
  - mapping from AC to Message-level security is not the topic of this talk !
  - AC + authentication enforced at the level of messages with cryptographic protocols and cryptographic primitives

# MESSAGE-LEVEL SECURITY

- Security of data protected off-line by Access Control
  - Web Services : move from off-line to on-line
  - Thus need to protect the data on its trips
- ⇒ Message level security, derived from :
- the environment in which the data will be accessed
  - the AC requirements on the data
  - additional need : authentication of requester, integrity of request, ...
  - mapping from AC to Message-level security is not the topic of this talk !
  - AC + authentication enforced at the level of messages with cryptographic protocols and cryptographic primitives

# MESSAGE-LEVEL SECURITY

- Security of data protected off-line by Access Control
  - Web Services : move from off-line to on-line
  - Thus need to protect the data on its trips
- ⇒ Message level security, derived from :
- environment in which the data will be accessed
  - AC restrictions on the data
  - additional need : authentication of requester, integrity of request, . . .
  - mapping from AC to Message-level security is not the topic of this talk !
  - AC + authentication enforced at the level of messages with cryptographic protocols and cryptographic primitives

# MESSAGE-LEVEL SECURITY

- Security of data protected off-line by Access Control
  - Web Services : move from off-line to on-line
  - Thus need to protect the data on its trips
- ⇒ Message level security, derived from :
- environment in which the data will be accessed
  - AC restrictions on the data
- additional need : authentication of requester, integrity of request, . . .
- mapping from AC to Message-level security is not the topic of this talk !
- AC + authentication enforced at the level of messages with cryptographic protocols and cryptographic primitives

# MESSAGE-LEVEL SECURITY

- Security of data protected off-line by Access Control
  - Web Services : move from off-line to on-line
  - Thus need to protect the data on its trips
- ⇒ Message level security, derived from :
- environment in which the data will be accessed
  - AC restrictions on the data
- additional need : authentication of requester, integrity of request, . . .
  - mapping from AC to Message-level security is not the topic of this talk !
  - AC + authentication enforced at the level of messages with cryptographic protocols and cryptographic primitives

# MESSAGE-LEVEL SECURITY

- Security of data protected off-line by Access Control
  - Web Services : move from off-line to on-line
  - Thus need to protect the data on its trips
- ⇒ Message level security, derived from :
- environment in which the data will be accessed
  - AC restrictions on the data
- additional need : authentication of requester, integrity of request, . . .
  - mapping from AC to Message-level security is not the topic of this talk !
  - AC + authentication enforced at the level of messages with cryptographic protocols and cryptographic primitives

# MESSAGE-LEVEL SECURITY

- Security of data protected off-line by Access Control
  - Web Services : move from off-line to on-line
  - Thus need to protect the data on its trips
- ⇒ Message level security, derived from :
- environment in which the data will be accessed
  - AC restrictions on the data
- additional need : authentication of requester, integrity of request, . . .
  - mapping from AC to Message-level security is not the topic of this talk !
  - AC + authentication enforced at the level of messages with cryptographic protocols and cryptographic primitives

# MESSAGE-LEVEL SECURITY

- Security of data protected off-line by Access Control
  - Web Services : move from off-line to on-line
  - Thus need to protect the data on its trips
- ⇒ Message level security, derived from :
- environment in which the data will be accessed
  - AC restrictions on the data
- additional need : authentication of requester, integrity of request, . . .
  - mapping from AC to Message-level security is not the topic of this talk !
  - AC + authentication enforced at the level of messages with cryptographic protocols and cryptographic primitives

# MESSAGE-LEVEL SECURITY

- Security of data protected off-line by Access Control
  - Web Services : move from off-line to on-line
  - Thus need to protect the data on its trips
- ⇒ Message level security, derived from :
- environment in which the data will be accessed
  - AC restrictions on the data
- additional need : authentication of requester, integrity of request, . . .
  - mapping from AC to Message-level security is not the topic of this talk !
  - AC + authentication enforced at the level of messages with cryptographic protocols and cryptographic primitives

# EXPRESSION OF MESSAGE-LEVEL SECURITY

- **WS-Security** : at the level of the SOAP requests and response
- basically defines how to apply XML-Enc and XML-DSig on Soap messages
- **WS-SecurityPolicy** : attached to a service, an operation or a message
- declares :

# EXPRESSION OF MESSAGE-LEVEL SECURITY

- WS-Security : at the level of the SOAP requests and response
- basically defines how to apply XML-Enc and XML-DSig on Soap messages
- WS-SecurityPolicy : attached to a service, an operation or a message
- declares :

# EXPRESSION OF MESSAGE-LEVEL SECURITY

- WS-Security : at the level of the SOAP requests and response
- basically defines how to apply XML-Enc and XML-DSig on Soap messages
- WS-SecurityPolicy : attached to a service, an operation or a message
- declares :
  - ✦ admissible transport protocol (e.g. HTTP or TLS?)
  - ✦ cryptographic primitives to apply on the nodes of the requests and the response

# EXPRESSION OF MESSAGE-LEVEL SECURITY

- WS-Security : at the level of the SOAP requests and response
- basically defines how to apply XML-Enc and XML-DSig on Soap messages
- WS-SecurityPolicy : attached to a service, an operation or a message
- declares :
  - admissible transport protocol (e.g. HTTP or TLS?)
  - cryptographic primitives to apply on the nodes of the requests and the response
  - asymmetry : requirement on the request, declaration of what will be ensured on the response
  - Out of scope : there are different possible alternatives

# EXPRESSION OF MESSAGE-LEVEL SECURITY

- WS-Security : at the level of the SOAP requests and response
- basically defines how to apply XML-Enc and XML-DSig on Soap messages
- WS-SecurityPolicy : attached to a service, an operation or a message
- declares :
  - admissible transport protocol (e.g. HTTP or TLS?)
  - cryptographic primitives to apply on the nodes of the requests and the response
  - asymmetry : requirement on the request, declaration of what will be ensured on the response
  - Out of scope : there are different possible alternatives

# EXPRESSION OF MESSAGE-LEVEL SECURITY

- WS-Security : at the level of the SOAP requests and response
- basically defines how to apply XML-Enc and XML-DSig on Soap messages
- WS-SecurityPolicy : attached to a service, an operation or a message
- declares :
  - admissible transport protocol (e.g. HTTP or TLS?)
  - cryptographic primitives to apply on the nodes of the requests and the response
  - asymmetry : requirement on the request, declaration of what will be ensured on the response
  - Out of scope : there are different possible alternatives

# EXPRESSION OF MESSAGE-LEVEL SECURITY

- WS-Security : at the level of the SOAP requests and response
- basically defines how to apply XML-Enc and XML-DSig on Soap messages
- WS-SecurityPolicy : attached to a service, an operation or a message
- declares :
  - admissible transport protocol (e.g. HTTP or TLS?)
  - cryptographic primitives to apply on the nodes of the requests and the response
  - asymmetry : requirement on the request, declaration of what will be ensured on the response
  - Out of scope : there are different possible alternatives

# EXPRESSION OF MESSAGE-LEVEL SECURITY

- WS-Security : at the level of the SOAP requests and response
- basically defines how to apply XML-Enc and XML-DSig on Soap messages
- WS-SecurityPolicy : attached to a service, an operation or a message
- declares :
  - admissible transport protocol (e.g. HTTP or TLS?)
  - cryptographic primitives to apply on the nodes of the requests and the response
  - asymmetry : requirement on the request, declaration of what will be ensured on the response
  - Out of scope : there are different possible alternatives

# POLICY ON MESSAGES

- Possibility to encrypt/sign the whole message, the body or the header
- Possibility to encrypt/sign nodes based on XPath expressions
- Permits to specify encryption/signature on nodes
  - with a grant tag
  - with a grant place
- In particular it permits by encapsulation to map uniformly an access control policy on all services published by an organization (?)
- Validation of a message : process of checking that a message is a model for the Security Policy

# POLICY ON MESSAGES

- Possibility to encrypt/sign the whole message, the body or the header
- Possibility to encrypt/sign nodes based on XPath expressions
- Permits to specify encryption/signature on nodes
  - ✦ with a given tag
  - ✦ at a given place
- In particular it permits by encapsulation to map uniformly an access control policy on all services published by an organization (?)
- Validation of a message : process of checking that a message is a model for the Security Policy

# POLICY ON MESSAGES

- Possibility to encrypt/sign the whole message, the body or the header
- Possibility to encrypt/sign nodes based on XPath expressions
- Permits to specify encryption/signature on nodes
  - with a given tag
  - at a given place
- In particular it permits by encapsulation to map uniformly an access control policy on all services published by an organization (?)
- Validation of a message : process of checking that a message is a model for the Security Policy

# POLICY ON MESSAGES

- Possibility to encrypt/sign the whole message, the body or the header
- Possibility to encrypt/sign nodes based on XPath expressions
- Permits to specify encryption/signature on nodes
  - with a given tag
  - at a given place
- In particular it permits by encapsulation to map uniformly an access control policy on all services published by an organization (?)
- Validation of a message : process of checking that a message is a model for the Security Policy

# POLICY ON MESSAGES

- Possibility to encrypt/sign the whole message, the body or the header
- Possibility to encrypt/sign nodes based on XPath expressions
- Permits to specify encryption/signature on nodes
  - with a given tag
  - at a given place
- In particular it permits by encapsulation to map uniformly an access control policy on all services published by an organization (?)
- Validation of a message : process of checking that a message is a model for the Security Policy

# POLICY ON MESSAGES

- Possibility to encrypt/sign the whole message, the body or the header
- Possibility to encrypt/sign nodes based on XPath expressions
- Permits to specify encryption/signature on nodes
  - with a given tag
  - at a given place
- In particular it permits by encapsulation to map uniformly an access control policy on all services published by an organization (?)
- Validation of a message : process of checking that a message is a model for the Security Policy

# POLICY ON MESSAGES

- Possibility to encrypt/sign the whole message, the body or the header
- Possibility to encrypt/sign nodes based on XPath expressions
- Permits to specify encryption/signature on nodes
  - with a given tag
  - at a given place
- In particular it permits by encapsulation to map uniformly an access control policy on all services published by an organization (?)
- Validation of a message : process of checking that a message is a model for the Security Policy

# OUTLINE

1 INTRODUCTION

2 TREATMENT OF SECURED MESSAGES

3 FINITE-STATE ANALYSIS OF SERVICES

# XML FIREWALLS

- For modularity reasons, services should not handle the validation of Soap requests themselves
- This task can be given to the Web Server before the invocation of the service
- Messages can also be validated before the arrival on the server
- XML firewalls check the validity of incoming requests, remove the encryption/signature parts, and forward message to the Web server
- Possibly several layers, and forwarding with new encryption/signature

# XML FIREWALLS

- For modularity reasons, services should not handle the validation of Soap requests themselves
- This task can be given to the Web Server before the invocation of the service
- Messages can also be validated before the arrival on the server
- XML firewalls check the validity of incoming requests, remove the encryption/signature parts, and forward message to the Web server
- Possibly several layers, and forwarding with new encryption/signature

# XML FIREWALLS

- For modularity reasons, services should not handle the validation of Soap requests themselves
- This task can be given to the Web Server before the invocation of the service
- Messages can also be validated before the arrival on the server
- XML firewalls check the validity of incoming requests, remove the encryption/signature parts, and forward message to the Web server
- Possibly several layers, and forwarding with new encryption/signature

# XML FIREWALLS

- For modularity reasons, services should not handle the validation of Soap requests themselves
- This task can be given to the Web Server before the invocation of the service
- Messages can also be validated before the arrival on the server
- XML firewalls check the validity of incoming requests, remove the encryption/signature parts, and forward message to the Web server
- Possibly several layers, and forwarding with new encryption/signature

# XML FIREWALLS

- For modularity reasons, services should not handle the validation of Soap requests themselves
- This task can be given to the Web Server before the invocation of the service
- Messages can also be validated before the arrival on the server
- XML firewalls check the validity of incoming requests, remove the encryption/signature parts, and forward message to the Web server
- Possibly several layers, and forwarding with new encryption/signature

# VALIDATION OF A MESSAGE

- w.r.t. the schema of the request in the WSDL description of the operation
- w.r.t. the security annotations/transformations that must be present
- since it is possible to encrypt XML nodes, one needs to perform both at the same time, or to first validate the security policy, and then the schema
- validation of the schema can be strict or lax

# VALIDATION OF A MESSAGE

- w.r.t. the schema of the request in the WSDL description of the operation
- w.r.t. the security annotations/transformations that must be present
- since it is possible to encrypt XML nodes, one needs to perform both at the same time, or to first validate the security policy, and then the schema
- validation of the schema can be strict or lax

# VALIDATION OF A MESSAGE

- w.r.t. the schema of the request in the WSDL description of the operation
- w.r.t. the security annotations/transformations that must be present
- since it is possible to encrypt XML nodes, one needs to perform both at the same time, or to first validate the security policy, and then the schema
- validation of the schema can be strict or lax

# VALIDATION OF A MESSAGE

- w.r.t. the schema of the request in the WSDL description of the operation
- w.r.t. the security annotations/transformations that must be present
- since it is possible to encrypt XML nodes, one needs to perform both at the same time, or to first validate the security policy, and then the schema
- validation of the schema can be strict or lax

# STRICT/LAX VALIDATION OF XML MESSAGES

- **Strict** : the message has to conform exactly to the schema given in the WSDL description
- Re-ordering of nodes in a same hedge is considered a minor flaw, and may be accepted
- **Lax** : the schema defines the nodes that must be present, but nodes out of scope are admissible
- **Advantage** : permits to give a single schema for several operations, based on an OO view
- **note** : `SecurityPolicy :Layout :Strict` is independent, it only describes the ordering of security tokens (timestamps, then hash, then signature, for example)

# STRICT/LAX VALIDATION OF XML MESSAGES

- Strict : the message has to conform exactly to the schema given in the WSDL description
- Re-ordering of nodes in a same hedge is considered a minor flaw, and may be accepted
- Lax : the schema defines the nodes that must be present, but nodes out of scope are admissible
- Advantage : permits to give a single schema for several operations, based on an OO view
- note : SecurityPolicy :Layout :Strict is independent, it only describes the ordering of security tokens (timestamps, then hash, then signature, for example)

# STRICT/LAX VALIDATION OF XML MESSAGES

- Strict : the message has to conform exactly to the schema given in the WSDL description
- Re-ordering of nodes in a same hedge is considered a minor flaw, and may be accepted
- Lax : the schema defines the nodes that must be present, but nodes out of scope are admissible
- Advantage : permits to give a single schema for several operations, based on an OO view
- note : SecurityPolicy :Layout :Strict is independent, it only describes the ordering of security tokens (timestamps, then hash, then signature, for example)

# STRICT/LAX VALIDATION OF XML MESSAGES

- Strict : the message has to conform exactly to the schema given in the WSDL description
- Re-ordering of nodes in a same hedge is considered a minor flaw, and may be accepted
- Lax : the schema defines the nodes that must be present, but nodes out of scope are admissible
- Advantage : permits to give a single schema for several operations, based on an OO view
- note : SecurityPolicy :Layout :Strict is independent, it only describes the ordering of security tokens (timestamps, then hash, then signature, for example)

# STRICT/LAX VALIDATION OF XML MESSAGES

- Strict : the message has to conform exactly to the schema given in the WSDL description
- Re-ordering of nodes in a same hedge is considered a minor flaw, and may be accepted
- Lax : the schema defines the nodes that must be present, but nodes out of scope are admissible
- Advantage : permits to give a single schema for several operations, based on an OO view
- note : SecurityPolicy :Layout :Strict is independent, it only describes the ordering of security tokens (timestamps, then hash, then signature, for example)

# CONSEQUENCE FOR THE ANALYSIS OF WEB SERVICES AT THE MESSAGE LEVEL

- Before applying other techniques, one needs to know
  - what messages are acceptable
  - what treatment is applied on these messages
- Fournet-Gordon : the implementation is given
- In general, one needs to validate services wrt the implementation (the specification of the security policy is not enough)

# CONSEQUENCE FOR THE ANALYSIS OF WEB SERVICES AT THE MESSAGE LEVEL

- Before applying other techniques, one needs to know
  - what messages are acceptable
  - what treatment is applied on these messages
- Fournet-Gordon : the implementation is given
- In general, one needs to validate services wrt the implementation (the specification of the security policy is not enough)

# CONSEQUENCE FOR THE ANALYSIS OF WEB SERVICES AT THE MESSAGE LEVEL

- Before applying other techniques, one needs to know
  - what messages are acceptable
  - what treatment is applied on these messages
- Fournet-Gordon : the implementation is given
- In general, one needs to validate services wrt the implementation (the specification of the security policy is not enough)

# CONSEQUENCE FOR THE ANALYSIS OF WEB SERVICES AT THE MESSAGE LEVEL

- Before applying other techniques, one needs to know
  - what messages are acceptable
  - what treatment is applied on these messages
- Fournet-Gordon : the implementation is given
- In general, one needs to validate services wrt the implementation (the specification of the security policy is not enough)

# CONSEQUENCE FOR THE ANALYSIS OF WEB SERVICES AT THE MESSAGE LEVEL

- Before applying other techniques, one needs to know
  - what messages are acceptable
  - what treatment is applied on these messages
- Fournet-Gordon : the implementation is given
- In general, one needs to validate services wrt the implementation (the specification of the security policy is not enough)

# SUMMARY ON SERVICE VALIDATION

- **Implementation independent :**
  - assumed XML list/multiset properties
  - fetching a node and replacing it
- **Implementation dependent :**
- **Preliminary work :** addition of a multi-set operator to the standard Dolev-Yao model (the remainder of this talk)

# SUMMARY ON SERVICE VALIDATION

- Implementation independent :
  - assumed XML list/multiset properties
  - fetching a node and replacing it
- Implementation dependent :
  - not just one or all nodes in the result of an XPath expression are treated
  - if some bogus nodes are allowed
- Preliminary work : addition of a multi-set operator to the standard Dolev-Yao model (the remainder of this talk)

# SUMMARY ON SERVICE VALIDATION

- Implementation independent :
  - assumed XML list/multiset properties
  - fetching a node and replacing it
- Implementation dependent :
  - ✦ If just one or all nodes in the result of an XPath expression are treated
  - ✦ If some bogus nodes are allowed
- Preliminary work : addition of a multi-set operator to the standard Dolev-Yao model (the remainder of this talk)

# SUMMARY ON SERVICE VALIDATION

- Implementation independent :
  - assumed XML list/multiset properties
  - fetching a node and replacing it
- Implementation dependent :
  - If just one or all nodes in the result of an XPath expression are treated
  - If some bogus nodes are allowed
- Preliminary work : addition of a multi-set operator to the standard Dolev-Yao model (the remainder of this talk)

# SUMMARY ON SERVICE VALIDATION

- Implementation independent :
  - assumed XML list/multiset properties
  - fetching a node and replacing it
- Implementation dependent :
  - If just one or all nodes in the result of an XPath expression are treated
  - If some bogus nodes are allowed
- Preliminary work : addition of a multi-set operator to the standard Dolev-Yao model (the remainder of this talk)

# SUMMARY ON SERVICE VALIDATION

- Implementation independent :
  - assumed XML list/multiset properties
  - fetching a node and replacing it
- Implementation dependent :
  - If just one or all nodes in the result of an XPath expression are treated
  - If some bogus nodes are allowed
- Preliminary work : addition of a multi-set operator to the standard Dolev-Yao model (the remainder of this talk)

# SUMMARY ON SERVICE VALIDATION

- Implementation independent :
  - assumed XML list/multiset properties
  - fetching a node and replacing it
- Implementation dependent :
  - If just one or all nodes in the result of an XPath expression are treated
  - If some bogus nodes are allowed
- Preliminary work : addition of a multi-set operator to the standard Dolev-Yao model (the remainder of this talk)

# OUTLINE

1 INTRODUCTION

2 TREATMENT OF SECURED MESSAGES

3 FINITE-STATE ANALYSIS OF SERVICES

# THE EASY WAY

- Intruder constraint systems formalism has been developed for the bounded number of messages analysis of cryptographic protocols
- Constraint systems parameterized by a signature and the possible operations of the intruder
- It suffices to solve Constraint Systems for the intruder acting on free unary symbols (for tagged XML nodes) and on an AC symbol (for descendants of an element)
- Both problems are quite easy to solve
- Then one combine these results with already-known decidability results on the analysis of protocols
- And one obtain the decidability of Web Services message level analysis
- almost right!

# THE EASY WAY

- Intruder constraint systems formalism has been developed for the bounded number of messages analysis of cryptographic protocols
- Constraint systems parameterized by a signature and the possible operations of the intruder
- It suffices to solve Constraint Systems for the intruder acting on free unary symbols (for tagged XML nodes) and on an AC symbol (for descendants of an element)
- Both problems are quite easy to solve
- Then one combine these results with already-known decidability results on the analysis of protocols
- And one obtain the decidability of Web Services message level analysis
- almost right !

# THE EASY WAY

- Intruder constraint systems formalism has been developed for the bounded number of messages analysis of cryptographic protocols
- Constraint systems parameterized by a signature and the possible operations of the intruder
- It suffices to solve Constraint Systems for the intruder acting on free unary symbols (for tagged XML nodes) and on an AC symbol (for descendants of an element)
- Both problems are quite easy to solve
- Then one combine these results with already-known decidability results on the analysis of protocols
- And one obtain the decidability of Web Services message level analysis
- almost right!

# THE EASY WAY

- Intruder constraint systems formalism has been developed for the bounded number of messages analysis of cryptographic protocols
- Constraint systems parameterized by a signature and the possible operations of the intruder
- It suffices to solve Constraint Systems for the intruder acting on free unary symbols (for tagged XML nodes) and on an AC symbol (for descendants of an element)
- Both problems are quite easy to solve
- Then one combine these results with already-known decidability results on the analysis of protocols
- And one obtain the decidability of Web Services message level analysis
- almost right !

# THE EASY WAY

- Intruder constraint systems formalism has been developed for the bounded number of messages analysis of cryptographic protocols
- Constraint systems parameterized by a signature and the possible operations of the intruder
- It suffices to solve Constraint Systems for the intruder acting on free unary symbols (for tagged XML nodes) and on an AC symbol (for descendants of an element)
- Both problems are quite easy to solve
- Then one combine these results with already-known decidability results on the analysis of protocols
- And one obtain the decidability of Web Services message level analysis
- almost right !

# THE EASY WAY

- Intruder constraint systems formalism has been developed for the bounded number of messages analysis of cryptographic protocols
- Constraint systems parameterized by a signature and the possible operations of the intruder
- It suffices to solve Constraint Systems for the intruder acting on free unary symbols (for tagged XML nodes) and on an AC symbol (for descendants of an element)
- Both problems are quite easy to solve
- Then one combine these results with already-known decidability results on the analysis of protocols
- And one obtain the decidability of Web Services message level analysis
- almost right !

# THE EASY WAY

- Intruder constraint systems formalism has been developed for the bounded number of messages analysis of cryptographic protocols
- Constraint systems parameterized by a signature and the possible operations of the intruder
- It suffices to solve Constraint Systems for the intruder acting on free unary symbols (for tagged XML nodes) and on an AC symbol (for descendants of an element)
- Both problems are quite easy to solve
- Then one combine these results with already-known decidability results on the analysis of protocols
- And one obtain the decidability of Web Services message level analysis
- almost right !

# COMBINATION RESULT

## THEOREM

*If Intruder Constraint systems with lcr are satisfiable w.r.t. two intruder deduction systems  $\langle \mathcal{F}_1, S_1, \mathcal{E}_1 \rangle$  and  $\langle \mathcal{F}_2, S_2, \mathcal{E}_2 \rangle$  on disjoint signatures then they are satisfiable w.r.t. the deduction system  $\langle \mathcal{F}_1 \cup \mathcal{F}_2, S_1 \cup S_2, \mathcal{E}_1 \cup \mathcal{E}_2 \rangle$*

- $\mathcal{F}$  is a signature and  $\mathcal{E}$  is an equational theory
- $S_1$  and  $S_2$  are sets of terms
- deductions are the ground instances of rules  $\text{Var}(t) \rightarrow t$  with  $t \in S$

# COMBINATION RESULT

## THEOREM

*If Intruder Constraint systems with lcr are satisfiable w.r.t. two intruder deduction systems  $\langle \mathcal{F}_1, S_1, \mathcal{E}_1 \rangle$  and  $\langle \mathcal{F}_2, S_2, \mathcal{E}_2 \rangle$  on disjoint signatures then they are satisfiable w.r.t. the deduction system  $\langle \mathcal{F}_1 \cup \mathcal{F}_2, S_1 \cup S_2, \mathcal{E}_1 \cup \mathcal{E}_2 \rangle$*

- $\mathcal{F}$  is a signature and  $\mathcal{E}$  is an equational theory
- $S_1$  and  $S_2$  are sets of terms
- deductions are the ground instances of rules  $\text{Var}(t) \rightarrow t$  with  $t \in S$

# COMBINATION RESULT

## THEOREM

*If Intruder Constraint systems with lcr are satisfiable w.r.t. two intruder deduction systems  $\langle \mathcal{F}_1, S_1, \mathcal{E}_1 \rangle$  and  $\langle \mathcal{F}_2, S_2, \mathcal{E}_2 \rangle$  on disjoint signatures then they are satisfiable w.r.t. the deduction system  $\langle \mathcal{F}_1 \cup \mathcal{F}_2, S_1 \cup S_2, \mathcal{E}_1 \cup \mathcal{E}_2 \rangle$*

- $\mathcal{F}$  is a signature and  $\mathcal{E}$  is an equational theory
- $S_1$  and  $S_2$  are sets of terms
- deductions are the ground instances of rules  $\text{Var}(t) \rightarrow t$  with  $t \in S$

# LIMIT OF THE MODEL

- Can express any protocol
- used in many instances
- but cannot express non-determinism : by transitivity of equality, two reductions of a same term are equal
- In particular, it is not possible to express the extraction of one element from a set (one has to specify the implementation of sets and a deterministic algorithm employed to extract the element)
- Does not fit with the inherent non-determinism of XML

# LIMIT OF THE MODEL

- Can express any protocol
- used in many instances
- but cannot express non-determinism : by transitivity of equality, two reductions of a same term are equal
- In particular, it is not possible to express the extraction of one element from a set (one has to specify the implementation of sets and a deterministic algorithm employed to extract the element)
- Does not fit with the inherent non-determinism of XML

# LIMIT OF THE MODEL

- Can express any protocol
- used in many instances
- but cannot express non-determinism : by transitivity of equality, two reductions of a same term are equal
- In particular, it is not possible to express the extraction of one element from a set (one has to specify the implementation of sets and a deterministic algorithm employed to extract the element)
- Does not fit with the inherent non-determinism of XML

# LIMIT OF THE MODEL

- Can express any protocol
- used in many instances
- but cannot express non-determinism : by transitivity of equality, two reductions of a same term are equal
- In particular, it is not possible to express the extraction of one element from a set (one has to specify the implementation of sets and a deterministic algorithm employed to extract the element)
- Does not fit with the inherent non-determinism of XML

# LIMIT OF THE MODEL

- Can express any protocol
- used in many instances
- but cannot express non-determinism : by transitivity of equality, two reductions of a same term are equal
- In particular, it is not possible to express the extraction of one element from a set (one has to specify the implementation of sets and a deterministic algorithm employed to extract the element)
- Does not fit with the inherent non-determinism of XML

# A NEW HOPE ?

- Take the previous combination result, and replace along the proof the rules  $\text{Var}(t) \rightarrow t$  with rules  $t_1, \dots, t_n \rightarrow t$
- This results in a “pattern-matching” model, where received messages are patterns
- But :

# A NEW HOPE ?

- Take the previous combination result, and replace along the proof the rules  $\text{Var}(t) \rightarrow t$  with rules  $t_1, \dots, t_n \rightarrow t$
- This results in a “pattern-matching” model, where received messages are patterns

- But :

- Few results with patterns and equational theories
- In fact, always with specific (to the equational theory) restrictions on admissible patterns.

There are some interesting results in the theory of *rewriting* (see [10]).

There are also some interesting results in the theory of *quantification* (see [11]).

There are also some interesting results in the theory of *quantification* (see [11]).

There are also some interesting results in the theory of *quantification* (see [11]).

There are also some interesting results in the theory of *quantification* (see [11]).

There are also some interesting results in the theory of *quantification* (see [11]).

There are also some interesting results in the theory of *quantification* (see [11]).

There are also some interesting results in the theory of *quantification* (see [11]).

# A NEW HOPE ?

- Take the previous combination result, and replace along the proof the rules  $\text{Var}(t) \rightarrow t$  with rules  $t_1, \dots, t_n \rightarrow t$
- This results in a “pattern-matching” model, where received messages are patterns
- **But :**
  - Few results with patterns and equational theories
  - In fact, always with specific (to the equational theory) restrictions on admissible patterns
  - These restrictions are specific to the theory handled (AG+DY), and are checked on the fly (quantification on all possible instantiations)
  - Even the proof with these restriction is not completely convincing
  - Conclusion : a generic framework seems hard to give

# A NEW HOPE ?

- Take the previous combination result, and replace along the proof the rules  $\text{Var}(t) \rightarrow t$  with rules  $t_1, \dots, t_n \rightarrow t$
- This results in a “pattern-matching” model, where received messages are patterns
- **But :**
  - Few results with patterns and equational theories
  - In fact, always with specific (to the equational theory) restrictions on admissible patterns
  - These restrictions are specific to the theory handled (AG+DY), and are checked on the fly (quantification on all possible instantiations)
  - Even the proof with these restriction is not completely convincing
  - Conclusion : a generic framework seems hard to give

# A NEW HOPE ?

- Take the previous combination result, and replace along the proof the rules  $\text{Var}(t) \rightarrow t$  with rules  $t_1, \dots, t_n \rightarrow t$
- This results in a “pattern-matching” model, where received messages are patterns
- **But :**
  - Few results with patterns and equational theories
  - In fact, always with specific (to the equational theory) restrictions on admissible patterns
  - These restrictions are specific to the theory handled (AG+DY), and are checked on the fly (quantification on all possible instantiations)
  - Even the proof with these restriction is not completely convincing
  - Conclusion : a generic framework seems hard to give

# A NEW HOPE ?

- Take the previous combination result, and replace along the proof the rules  $\text{Var}(t) \rightarrow t$  with rules  $t_1, \dots, t_n \rightarrow t$
- This results in a “pattern-matching” model, where received messages are patterns
- **But :**
  - Few results with patterns and equational theories
  - In fact, always with specific (to the equational theory) restrictions on admissible patterns
  - These restrictions are specific to the theory handled (AG+DY), and are checked on the fly (quantification on all possible instantiations)
  - Even the proof with these restriction is not completely convincing
  - Conclusion : a generic framework seems hard to give

# A NEW HOPE ?

- Take the previous combination result, and replace along the proof the rules  $\text{Var}(t) \rightarrow t$  with rules  $t_1, \dots, t_n \rightarrow t$
- This results in a “pattern-matching” model, where received messages are patterns
- **But :**
  - Few results with patterns and equational theories
  - In fact, always with specific (to the equational theory) restrictions on admissible patterns
  - These restrictions are specific to the theory handled (AG+DY), and are checked on the fly (quantification on all possible instantiations)
  - Even the proof with these restriction is not completely convincing
  - Conclusion : a generic framework seems hard to give

# A NEW HOPE ?

- Take the previous combination result, and replace along the proof the rules  $\text{Var}(t) \rightarrow t$  with rules  $t_1, \dots, t_n \rightarrow t$
- This results in a “pattern-matching” model, where received messages are patterns
- **But :**
  - Few results with patterns and equational theories
  - In fact, always with specific (to the equational theory) restrictions on admissible patterns
  - These restrictions are specific to the theory handled (AG+DY), and are checked on the fly (quantification on all possible instantiations)
  - Even the proof with these restriction is not completely convincing
  - Conclusion : a generic framework seems hard to give

# BACK TO BASICS

- Start from “old”, unoptimized models
- Basics on honest agents :
  - They have an initial knowledge, which is a set of ground terms
  - In the course of the protocol, they receive messages and send responses
  - Responses are constructed from the initial knowledge and the received messages
  - We think of the construction as being the response of a construction, which are deductive steps
- Ditto for the intruder, though we search the sequence of constructions

# BACK TO BASICS

- Start from “old”, unoptimized models
- Basics on honest agents :
  - They have an initial knowledge, which is a set of ground terms
  - In the course of the protocol, they receive messages and send responses
  - Responses are constructed from the initial knowledge and the received messages
  - emphasis on the construction : we keep the sequence of constructions, which are deduction steps
- Ditto for the intruder, though we search the sequence of constructions

# BACK TO BASICS

- Start from “old”, unoptimized models
- Basics on honest agents :
  - They have an initial knowledge, which is a set of ground terms
  - In the course of the protocol, they receive messages and send responses
  - Responses are constructed from the initial knowledge and the received messages
  - emphasis on the construction : we keep the sequence of constructions, which are deduction steps
- Ditto for the intruder, though we search the sequence of constructions

# BACK TO BASICS

- Start from “old”, unoptimized models
- Basics on honest agents :
  - They have an initial knowledge, which is a set of ground terms
  - In the course of the protocol, they receive messages and send responses
  - Responses are constructed from the initial knowledge and the received messages
  - emphasis on the construction : we keep the sequence of constructions, which are deduction steps
- Ditto for the intruder, though we search the sequence of constructions

# BACK TO BASICS

- Start from “old”, unoptimized models
- Basics on honest agents :
  - They have an initial knowledge, which is a set of ground terms
  - In the course of the protocol, they receive messages and send responses
  - Responses are constructed from the initial knowledge and the received messages
  - emphasis on the construction : we keep the sequence of constructions, which are deduction steps
- Ditto for the intruder, though we search the sequence of constructions

# BACK TO BASICS

- Start from “old”, unoptimized models
- Basics on honest agents :
  - They have an initial knowledge, which is a set of ground terms
  - In the course of the protocol, they receive messages and send responses
  - Responses are constructed from the initial knowledge and the received messages
  - emphasis on the construction : we keep the sequence of constructions, which are deduction steps
- Ditto for the intruder, though we search the sequence of constructions

# BACK TO BASICS

- Start from “old”, unoptimized models
- Basics on honest agents :
  - They have an initial knowledge, which is a set of ground terms
  - In the course of the protocol, they receive messages and send responses
  - Responses are constructed from the initial knowledge and the received messages
  - emphasis on the construction : we keep the sequence of constructions, which are deduction steps
- Ditto for the intruder, though we search the sequence of constructions

# SYMBOLIC DERIVATIONS

- A deduction system  $\langle \mathcal{F}, S, \mathcal{E} \rangle$ , where  $S$  is a set of *deduction rules*  $t_1, \dots, t_n \rightarrow t$
- A sequence of variables  $\mathcal{V}$  each denoting a basic step in the execution of the agent
- A set of equations  $\mathcal{S}$  representing the constructions and the extra checkings performed by the agent
- A finite set of ground terms  $\mathcal{K}$
- Two subsequences of  $\mathcal{V}$ , the *input* and the *output* variables

# SYMBOLIC DERIVATIONS

- A deduction system  $\langle \mathcal{F}, S, \mathcal{E} \rangle$ , where  $S$  is a set of *deduction rules*  $t_1, \dots, t_n \rightarrow t$
- A sequence of variables  $\mathcal{V}$  each denoting a basic step in the execution of the agent
- A set of equations  $\mathcal{S}$  representing the constructions and the extra checkings performed by the agent
- A finite set of ground terms  $\mathcal{K}$
- Two subsequences of  $\mathcal{V}$ , the *input* and the *output* variables

# SYMBOLIC DERIVATIONS

- A deduction system  $\langle \mathcal{F}, S, \mathcal{E} \rangle$ , where  $S$  is a set of *deduction rules*  $t_1, \dots, t_n \rightarrow t$
- A sequence of variables  $\mathcal{V}$  each denoting a basic step in the execution of the agent
- A set of equations  $\mathcal{S}$  representing the constructions and the extra checkings performed by the agent
- A finite set of ground terms  $\mathcal{K}$
- Two subsequences of  $\mathcal{V}$ , the *input* and the *output* variables

# SYMBOLIC DERIVATIONS

- A deduction system  $\langle \mathcal{F}, S, \mathcal{E} \rangle$ , where  $S$  is a set of *deduction rules*  $t_1, \dots, t_n \rightarrow t$
- A sequence of variables  $\mathcal{V}$  each denoting a basic step in the execution of the agent
- A set of equations  $\mathcal{S}$  representing the constructions and the extra checkings performed by the agent
- A finite set of ground terms  $\mathcal{K}$
- Two subsequences of  $\mathcal{V}$ , the *input* and the *output* variables

# SYMBOLIC DERIVATIONS

- A deduction system  $\langle \mathcal{F}, S, \mathcal{E} \rangle$ , where  $S$  is a set of *deduction rules*  $t_1, \dots, t_n \rightarrow t$
- A sequence of variables  $\mathcal{V}$  each denoting a basic step in the execution of the agent
- A set of equations  $\mathcal{S}$  representing the constructions and the extra checkings performed by the agent
- A finite set of ground terms  $\mathcal{K}$
- Two subsequences of  $\mathcal{V}$ , the *input* and the *output* variables

# RESTRICTIONS (1/2)

On the deduction systems :

- quantification on all *normalization of pure* ground instances
- The free constants on the right (but for a special one) already appear on the left
- always true for rules  $\text{Var}(t) \rightarrow t$
- easily checkable for regular theories (A, AC, free, ...)

# RESTRICTIONS (1/2)

On the deduction systems :

- quantification on all *normalization of pure* ground instances
- The free constants on the right (but for a special one) already appear on the left
- always true for rules  $\text{Var}(t) \rightarrow t$
- easily checkable for regular theories (A, AC, free, ...)

# RESTRICTIONS (1/2)

On the deduction systems :

- quantification on all *normalization of pure* ground instances
- The free constants on the right (but for a special one) already appear on the left
- always true for rules  $\text{Var}(t) \rightarrow t$
- easily checkable for regular theories (A, AC, free,...)

# RESTRICTIONS (1/2)

On the deduction systems :

- quantification on all *normalization of pure* ground instances
- The free constants on the right (but for a special one) already appear on the left
- always true for rules  $\text{Var}(t) \rightarrow t$
- easily checkable for regular theories (A, AC, free, ...)

# RESTRICTIONS (2/2)

On a symbolic derivations  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  :

- For every variable  $v \in \mathcal{V}$ , either :
  - There exists an equation  $v = k$  in  $\mathcal{S}$  with  $k \in \mathcal{K}$
  - $v$  is an input position
  - There exists equations in  $\mathcal{S}$  ensuring that  $v$  is the rhs of a deduction rule whose lhs consists of variables occurring before  $v$  in  $\mathcal{V}$
  - Note : no restrictions on In/Out
- Note : If rules  $Var(t) \rightarrow t$ , it is possible to give an output as a term whose variables are input variables occurring before the output
- permits to englobe all known decidability results

# RESTRICTIONS (2/2)

On a symbolic derivations  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  :

- For every variable  $v \in \mathcal{V}$ , either :
  - There exists an equation  $v = k$  in  $\mathcal{S}$  with  $k \in \mathcal{K}$
  - $v$  is an input position
  - There exists equations in  $\mathcal{S}$  ensuring that  $v$  is the rhs of a deduction rule whose lhs consists of variables occurring before  $v$  in  $\mathcal{V}$
  - Note : no restrictions on In/Out
- Note : If rules  $Var(t) \rightarrow t$ , it is possible to give an output as a term whose variables are input variables occurring before the output
- permits to englobe all known decidability results

# RESTRICTIONS (2/2)

On a symbolic derivations  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  :

- For every variable  $v \in \mathcal{V}$ , either :
  - There exists an equation  $v = k$  in  $\mathcal{S}$  with  $k \in \mathcal{K}$
  - $v$  is an input position
  - There exists equations in  $\mathcal{S}$  ensuring that  $v$  is the rhs of a deduction rule whose lhs consists of variables occurring before  $v$  in  $\mathcal{V}$
  - Note : no restrictions on In/Out
- Note : If rules  $Var(t) \rightarrow t$ , it is possible to give an output as a term whose variables are input variables occurring before the output
- permits to englobe all known decidability results

# RESTRICTIONS (2/2)

On a symbolic derivations  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  :

- For every variable  $v \in \mathcal{V}$ , either :
  - There exists an equation  $v = k$  in  $\mathcal{S}$  with  $k \in \mathcal{K}$
  - $v$  is an input position
  - There exists equations in  $\mathcal{S}$  ensuring that  $v$  is the rhs of a deduction rule whose lhs consists of variables occurring before  $v$  in  $\mathcal{V}$
  - Note : no restrictions on In/Out
- Note : If rules  $Var(t) \rightarrow t$ , it is possible to give an output as a term whose variables are input variables occurring before the output
- permits to englobe all known decidability results

# RESTRICTIONS (2/2)

On a symbolic derivations  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  :

- For every variable  $v \in \mathcal{V}$ , either :
  - There exists an equation  $v = k$  in  $\mathcal{S}$  with  $k \in \mathcal{K}$
  - $v$  is an input position
  - There exists equations in  $\mathcal{S}$  ensuring that  $v$  is the rhs of a deduction rule whose lhs consists of variables occurring before  $v$  in  $\mathcal{V}$
  - Note : no restrictions on In/Out
- Note : If rules  $Var(t) \rightarrow t$ , it is possible to give an output as a term whose variables are input variables occurring before the output
- permits to englobe all known decidability results

# RESTRICTIONS (2/2)

On a symbolic derivations  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  :

- For every variable  $v \in \mathcal{V}$ , either :
  - There exists an equation  $v = k$  in  $\mathcal{S}$  with  $k \in \mathcal{K}$
  - $v$  is an input position
  - There exists equations in  $\mathcal{S}$  ensuring that  $v$  is the rhs of a deduction rule whose lhs consists of variables occurring before  $v$  in  $\mathcal{V}$
  - Note : no restrictions on In/Out
- Note : If rules  $Var(t) \rightarrow t$ , it is possible to give an output as a term whose variables are input variables occurring before the output
- permits to englobe all known decidability results

# RESTRICTIONS (2/2)

On a symbolic derivations  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  :

- For every variable  $v \in \mathcal{V}$ , either :
  - There exists an equation  $v = k$  in  $\mathcal{S}$  with  $k \in \mathcal{K}$
  - $v$  is an input position
  - There exists equations in  $\mathcal{S}$  ensuring that  $v$  is the rhs of a deduction rule whose lhs consists of variables occurring before  $v$  in  $\mathcal{V}$
  - Note : no restrictions on In/Out
- Note : If rules  $Var(t) \rightarrow t$ , it is possible to give an output as a term whose variables are input variables occurring before the output
- permits to englobe all known decidability results

# CLOSED SYMBOLIC DERIVATIONS

- A symbolic derivation  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  is closed if  $In = Out = \emptyset$
- Permits to define the result of an execution englobing the deduction of all participants
- One compose symbolic derivations by identifying the input of one with the output of the other

# CLOSED SYMBOLIC DERIVATIONS

- A symbolic derivation  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  is closed if  $In = Out = \emptyset$
- Permits to define the result of an execution englobing the deduction of all participants
- One compose symbolic derivations by identifying the input of one with the output of the other

# CLOSED SYMBOLIC DERIVATIONS

- A symbolic derivation  $(\mathcal{V}, \mathcal{S}, \mathcal{K}, In, Out)$  is closed if  $In = Out = \emptyset$
- Permits to define the result of an execution englobing the deduction of all participants
- One compose symbolic derivations by identifying the input of one with the output of the other

# DECISION PROBLEM

Analysis of protocols : Given a symbolic derivation  $\mathcal{D}$  and an initial knowledge set of the intruder, find if there exists a symbolic derivation with the same set of ground terms

- Given a deduction system  $\langle \mathcal{F}, \mathcal{S}, \mathcal{E} \rangle$
- Given a symbolic derivation  $\mathcal{D}$  (for honest agents) and a finite set of knowledge  $\mathcal{K}$  for the intruder
- Find if there exists a symbolic derivation  $\mathcal{D}'$  such that :

# DECISION PROBLEM

Analysis of protocols : Given a symbolic derivation  $\mathcal{D}$  and an initial knowledge set of the intruder, find if there exists a symbolic derivation with the same set of ground terms

- Given a deduction system  $\langle \mathcal{F}, \mathcal{S}, \mathcal{E} \rangle$
- Given a symbolic derivation  $\mathcal{D}$  (for honest agents) and a finite set of knowledge  $\mathcal{K}$  for the intruder
- Find if there exists a symbolic derivation  $\mathcal{D}'$  such that :
  - The composition of  $\mathcal{D}$  and  $\mathcal{D}'$  is a closed symbolic derivation
  - The set of initial knowledge of  $\mathcal{D}'$  is  $\mathcal{K}$

# DECISION PROBLEM

Analysis of protocols : Given a symbolic derivation  $\mathcal{D}$  and an initial knowledge set of the intruder, find if there exists a symbolic derivation with the same set of ground terms

- Given a deduction system  $\langle \mathcal{F}, \mathcal{S}, \mathcal{E} \rangle$
- Given a symbolic derivation  $\mathcal{D}$  (for honest agents) and a finite set of knowledge  $\mathcal{K}$  for the intruder
- Find if there exists a symbolic derivation  $\mathcal{D}'$  such that :
  - The composition of  $\mathcal{D}$  and  $\mathcal{D}'$  is a closed symbolic derivation
  - The set of initial knowledge of  $\mathcal{D}'$  is  $\mathcal{K}$
  - The equations in the obtained closed derivation are satisfiable by a ground substitution  $\sigma$

# DECISION PROBLEM

Analysis of protocols : Given a symbolic derivation  $\mathcal{D}$  and an initial knowledge set of the intruder, find if there exists a symbolic derivation with the same set of ground terms

- Given a deduction system  $\langle \mathcal{F}, \mathcal{S}, \mathcal{E} \rangle$
- Given a symbolic derivation  $\mathcal{D}$  (for honest agents) and a finite set of knowledge  $\mathcal{K}$  for the intruder
- Find if there exists a symbolic derivation  $\mathcal{D}'$  such that :
  - The composition of  $\mathcal{D}$  and  $\mathcal{D}'$  is a closed symbolic derivation
  - The set of initial knowledge of  $\mathcal{D}'$  is  $\mathcal{K}$
  - The equations in the obtained closed derivation are satisfiable by a ground substitution  $\sigma$

# DECISION PROBLEM

Analysis of protocols : Given a symbolic derivation  $\mathcal{D}$  and an initial knowledge set of the intruder, find if there exists a symbolic derivation with the same set of ground terms

- Given a deduction system  $\langle \mathcal{F}, \mathcal{S}, \mathcal{E} \rangle$
- Given a symbolic derivation  $\mathcal{D}$  (for honest agents) and a finite set of knowledge  $\mathcal{K}$  for the intruder
- Find if there exists a symbolic derivation  $\mathcal{D}'$  such that :
  - The composition of  $\mathcal{D}$  and  $\mathcal{D}'$  is a closed symbolic derivation
  - The set of initial knowledge of  $\mathcal{D}'$  is  $\mathcal{K}$
  - The equations in the obtained closed derivation are satisfiable by a ground substitution  $\sigma$

# DECISION PROBLEM

Analysis of protocols : Given a symbolic derivation  $\mathcal{D}$  and an initial knowledge set of the intruder, find if there exists a symbolic derivation with the same set of ground terms

- Given a deduction system  $\langle \mathcal{F}, \mathcal{S}, \mathcal{E} \rangle$
- Given a symbolic derivation  $\mathcal{D}$  (for honest agents) and a finite set of knowledge  $\mathcal{K}$  for the intruder
- Find if there exists a symbolic derivation  $\mathcal{D}'$  such that :
  - The composition of  $\mathcal{D}$  and  $\mathcal{D}'$  is a closed symbolic derivation
  - The set of initial knowledge of  $\mathcal{D}'$  is  $\mathcal{K}$
  - The equations in the obtained closed derivation are satisfiable by a ground substitution  $\sigma$

# MAIN RESULT

## THEOREM

*If satisfiability of symbolic derivations with lcr are decidable for two deduction systems, they are decidable for their union*

- Permits to add Dolev-Yao style encryption (syntactic or equational), xor, ... to an AC operator and free unary symbols
- Thus permits to decide the existence of an attack on the instance of a Web Service
- First step, note that the operations performed by a firewall may depend on the content of a message ("forall" operation)
- no extraction of an arbitrary subterm (W-I-P)

# MAIN RESULT

## THEOREM

*If satisfiability of symbolic derivations with lcr are decidable for two deduction systems, they are decidable for their union*

- Permits to add Dolev-Yao style encryption (syntactic or equational), xor, ... to an AC operator and free unary symbols
- Thus permits to decide the existence of an attack on the instance of a Web Service
- First step, note that the operations performed by a firewall may depend on the content of a message (“forall” operation)
- no extraction of an arbitrary subterm (W-I-P)

# MAIN RESULT

## THEOREM

*If satisfiability of symbolic derivations with lcr are decidable for two deduction systems, they are decidable for their union*

- Permits to add Dolev-Yao style encryption (syntactic or equational), xor, ... to an AC operator and free unary symbols
- Thus permits to decide the existence of an attack on the instance of a Web Service
- First step, note that the operations performed by a firewall may depend on the content of a message (“forall” operation)
- no extraction of an arbitrary subterm (W-I-P)

# MAIN RESULT

## THEOREM

*If satisfiability of symbolic derivations with lcr are decidable for two deduction systems, they are decidable for their union*

- Permits to add Dolev-Yao style encryption (syntactic or equational), xor, ... to an AC operator and free unary symbols
- Thus permits to decide the existence of an attack on the instance of a Web Service
- First step, note that the operations performed by a firewall may depend on the content of a message (“forall” operation)
- no extraction of an arbitrary subterm (W-I-P)