The CaPiTo Approach to Protocol Validation

Flemming Nielson

The Technical University of Denmark

Invited talk at NWPT'10
Turku, November 11^{th}, 2010

MT-LAB
A VKR CENTRE OF EXCELLENCE

Abstract

We show how to model service-oriented applications using the process algebra CaPiTo so that, on the one hand, we can achieve an abstract specification without being overwhelmed by the underlying implementation details and, on the other hand, we can obtain a concrete specification respecting the industrial standards used for ensuring security. We consider this development important in order to get a good agreement between the protocols analysed by formal tools and the applications developed by practitioners.

We then show how to transform the concrete specification into the LySa analysis framework, used in the SENSORIA EU project and originally developed in the DEGAS EU project, for analysing cryptographic protocols under a Dolev-Yao attacker. This allows us to perform a control flow analysis for ensuring the authenticity (as well as confidentiality) of messages exchanged between services. The LySa analysis framework is implemented in polynomial time in the size of the protocol specification using the Succinct Solver, that can solve a superset of Datalog clauses.

Joint work with Han Gao and Hanne Riis Nielson.

MT-LAB
A VKR CENTRE OF EXCELLENCE

Roadmap

The CaPiTo Approach to Protocol Validation

- (1) Service Oriented Computing
- (2) Abstract Level of CaPiTo
- (3) Protocol Stack Plug-in Level of CaPiTo
- (4) Concrete Level of CaPiTo
- (5) From CaPiTo to Code Stubs

Linking up with the LySa Approach to Protocol Validation

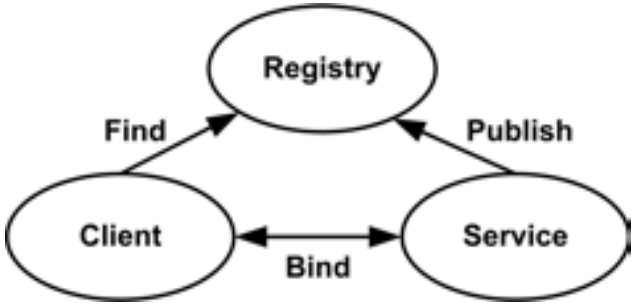
- (6) CaPiTo with Annotations from LySa
- (7) Static Analysis of LySa

Conclusion

MT-LAB
A VKR CENTRE OF EXCELLENCE

(1) Service Oriented Computing

A new computing paradigm that utilises services as the basic constructs to support the development of rapid, low-cost and easy composition of distributed applications even in heterogeneous environments.

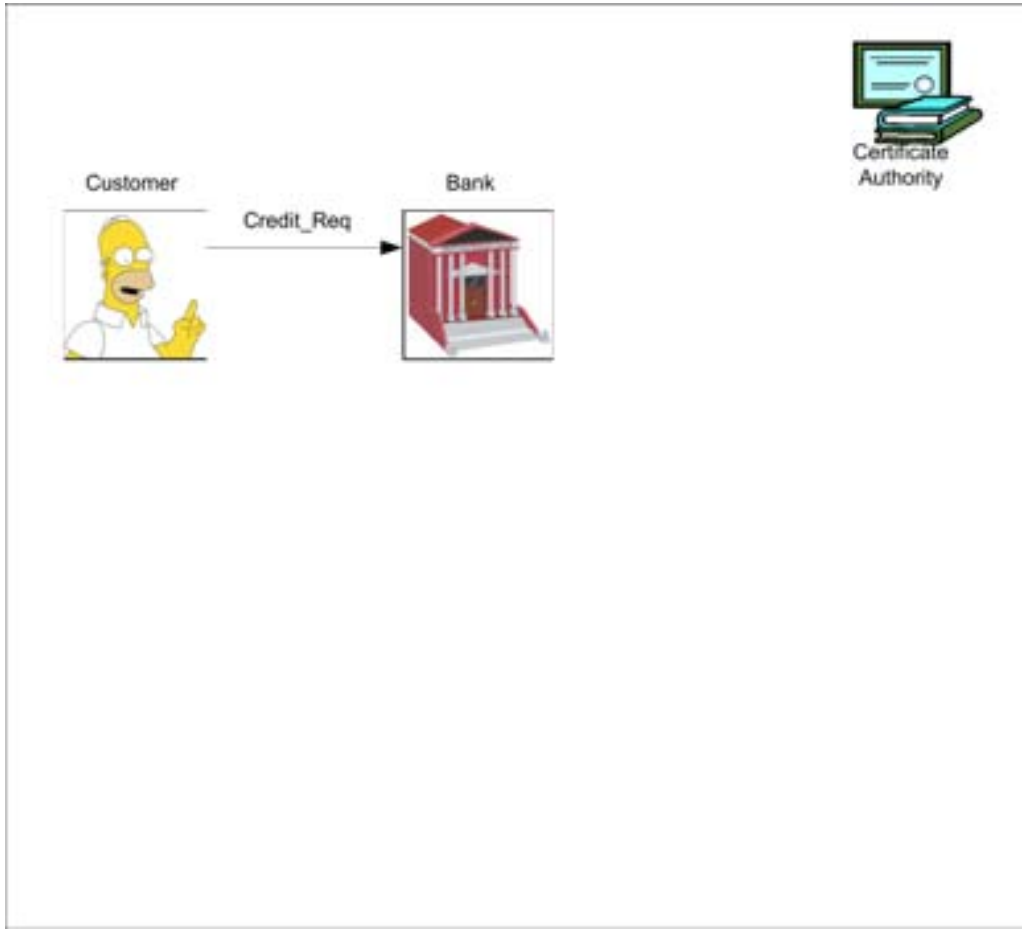


- Greater interoperability
- Loosely coupled
- Easier to integrate
- Increased reuse
- Reduce costs

SENSORIA: An EU project aiming to develop a novel comprehensive approach to the engineering of software systems for service-oriented architectures.

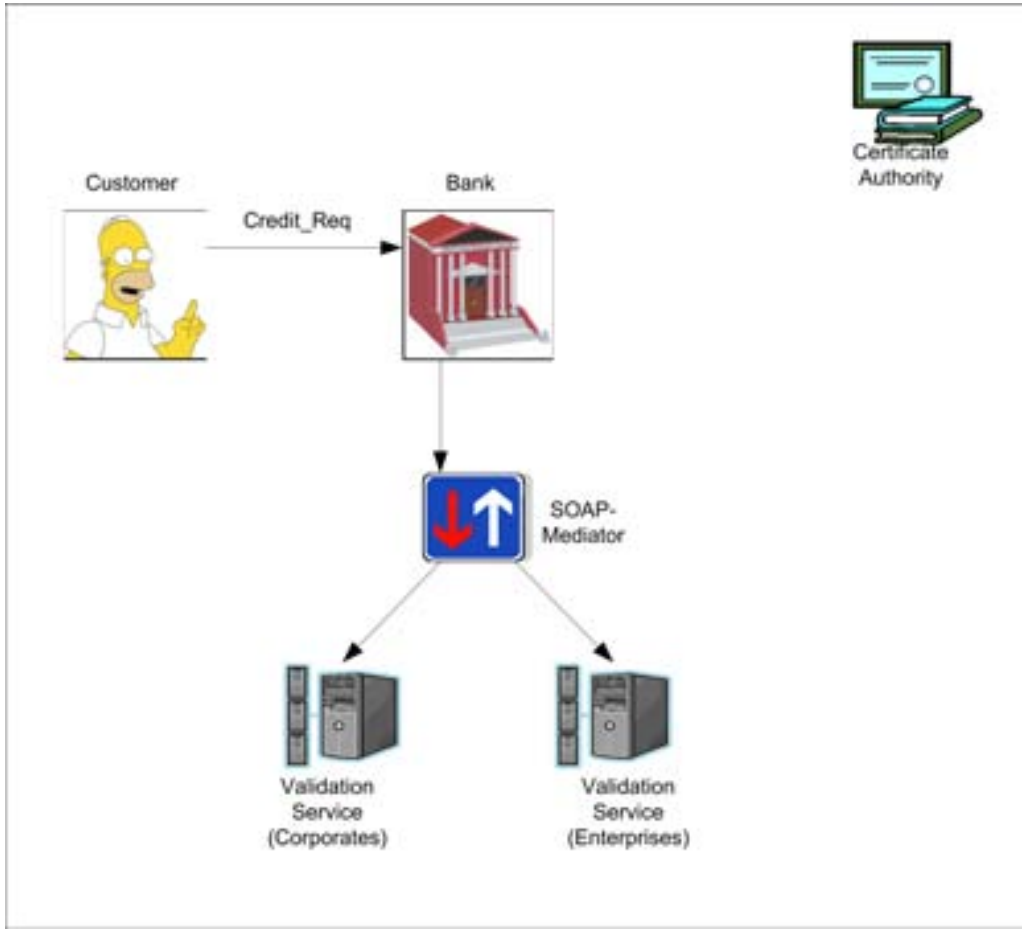
MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Pictures



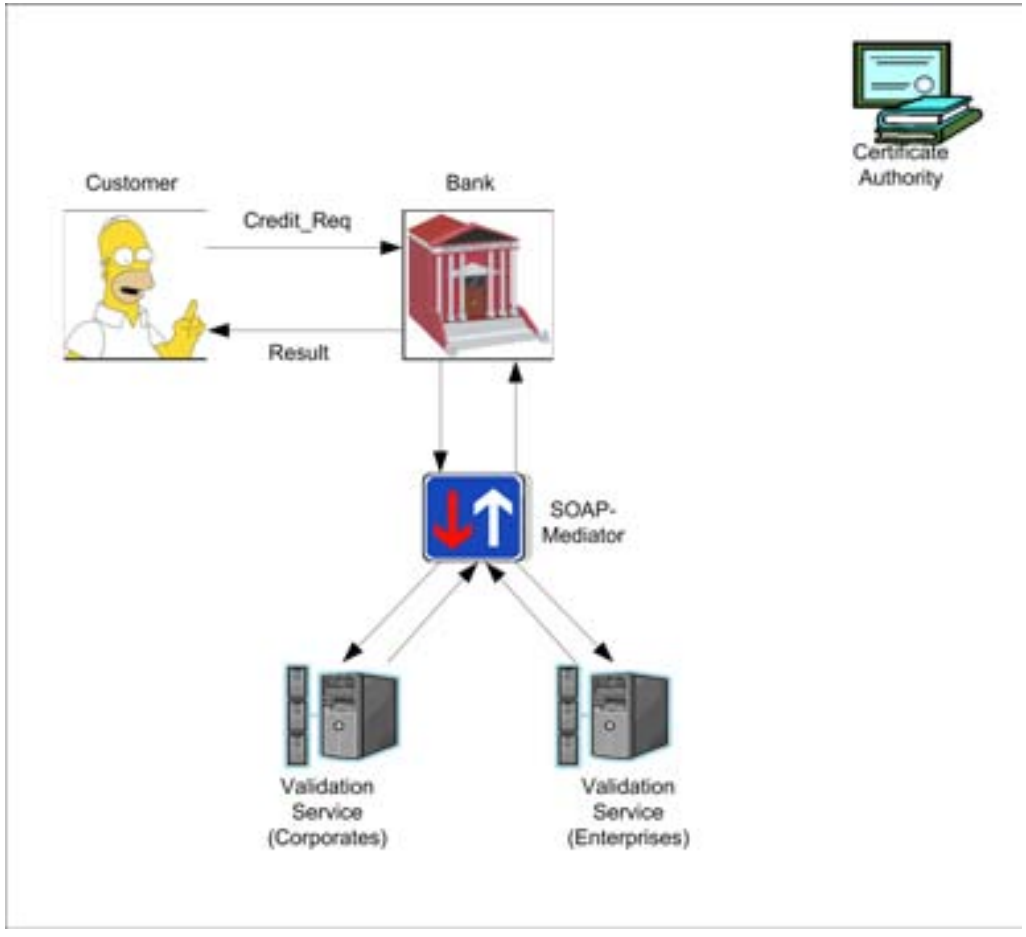
MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Pictures



MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Pictures



MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Narration

(TLS)

1. $C \rightarrow VS : N_C$
2. $VS \rightarrow C : N_{VS}, \mathbf{S}_{K_{CA}^-}(VS, K_{VS}^+)$
3. $C \rightarrow VS : \mathbf{P}_{K_{VS}^+}(N_{PMS})$

(BALANCE VALIDATION REQUEST)

4. $C \rightarrow VS : \mathbf{E}_{\mathbf{H}(N_C, N_{VS}, N_{PMS})}(Bta)$

(SERVICE INVOCATION AND ROUTING)

5. $VS \rightarrow SM : VS, Ser_E, \mathbf{S}_{K_{VS}^-}(\mathbf{P}_{K_{Ser_E}^+}(SN, Bta, \mathbf{S}_{K_{CA}^-}(VS, K_{VS}^+)))$
6. $SM \rightarrow Ser_E : VS, Ser_E, \mathbf{S}_{K_{VS}^-}(\mathbf{P}_{K_{Ser_E}^+}(SN, Bta, \mathbf{S}_{K_{CA}^-}(VS, K_{VS}^+)))$

(EVALUATION AND REPLY)

7. $Ser_E \rightarrow SM : Ser_E, VS, \mathbf{S}_{K_{Ser_E}^-}(\mathbf{P}_{K_{VS}^+}(SN, Res))$
8. $SM \rightarrow VS : Ser_E, VS, \mathbf{S}_{K_{Ser_E}^-}(\mathbf{P}_{K_{VS}^+}(SN, Res))$
9. $VS \rightarrow C : \mathbf{E}_{\mathbf{H}(N_C, N_{VS}, N_{PMS})}(Bta, Res)$

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Narration

(TLS)

1. $C \rightarrow VS : N_C$
2. $VS \rightarrow C : N_{VS}, \mathbf{S}_{K_{CA}^-}(VS, K_{VS}^+)$
3. $C \rightarrow VS : \mathbf{P}_{K_{VS}^+}(N_{PMS})$

(BALANCE VALIDATION REQUEST)

4. $C \rightarrow VS : \mathbf{E}_{\mathbf{H}(N_C, N_{VS}, N_{PMS})}(Bta)$

(SERVICE INVOCATION AND ROUTING)

5. $VS \rightarrow SM : VS, Ser_E, \mathbf{S}_{K_{VS}^-}(\mathbf{P}_{K_{Ser_E}^+}(SN, Bta, \mathbf{S}_{K_{CA}^-}(VS, K_{VS}^+)))$
6. $SM \rightarrow Ser_E : VS, Ser_E, \mathbf{S}_{K_{VS}^-}(\mathbf{P}_{K_{Ser_E}^+}(SN, Bta, \mathbf{S}_{K_{CA}^-}(VS, K_{VS}^+)))$

(EVALUATION AND REPLY)

7. $Ser_E \rightarrow SM : Ser_E, VS, \mathbf{S}_{K_{Ser_E}^-}(\mathbf{P}_{K_{VS}^+}(SN, Res))$
8. $SM \rightarrow VS : Ser_E, VS, \mathbf{S}_{K_{Ser_E}^-}(\mathbf{P}_{K_{VS}^+}(SN, Res))$
9. $VS \rightarrow C : \mathbf{E}_{\mathbf{H}(N_C, N_{VS}, N_{PMS})}(Bta, Res)$

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Narration

(TLS)

1. $C \rightarrow VS : N_C$
2. $VS \rightarrow C : N_{VS}, \mathbf{S}_{K_{CA}^-}(VS, K_{VS}^+)$
3. $C \rightarrow VS : \mathbf{P}_{K_{VS}^+}(N_{PMS})$

(BALANCE VALIDATION REQUEST)

4. $C \rightarrow VS : \mathbf{E}_{\mathbf{H}(N_C, N_{VS}, N_{PMS})}(Bta)$

(SERVICE INVOCATION AND ROUTING)

5. $VS \rightarrow SM : VS, Ser_E, \mathbf{S}_{K_{VS}^-}(\mathbf{P}_{K_{Ser_E}^+}(SN, Bta, \mathbf{S}_{K_{CA}^-}(VS, K_{VS}^+)))$
6. $SM \rightarrow Ser_E : VS, Ser_E, \mathbf{S}_{K_{VS}^-}(\mathbf{P}_{K_{Ser_E}^+}(SN, Bta, \mathbf{S}_{K_{CA}^-}(VS, K_{VS}^+)))$

(EVALUATION AND REPLY)

7. $Ser_E \rightarrow SM : Ser_E, VS, \mathbf{S}_{K_{Ser_E}^-}(\mathbf{P}_{K_{VS}^+}(SN, Res))$
8. $SM \rightarrow VS : Ser_E, VS, \mathbf{S}_{K_{Ser_E}^-}(\mathbf{P}_{K_{VS}^+}(SN, Res))$
9. $VS \rightarrow C : \mathbf{E}_{\mathbf{H}(N_C, N_{VS}, N_{PMS})}(Bta, Res)$

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Narration

(TLS)

1. $C \rightarrow VS : N_C$
2. $VS \rightarrow C : N_{VS}, \mathbf{S}_{K^-_{CA}}(VS, K^+_{VS})$
3. $C \rightarrow VS : \mathbf{P}_{K^+_{VS}}(N_{PMS})$

(BALANCE VALIDATION REQUEST)

4. $C \rightarrow VS : \mathbf{E}_{\mathbf{H}(N_C, N_{VS}, N_{PMS})}(Bta)$

(SERVICE INVOCATION AND ROUTING)

5. $VS \rightarrow SM : VS, Ser_E, \mathbf{S}_{K^-_{VS}}(\mathbf{P}_{K^+_{Ser_E}}(SN, Bta, \mathbf{S}_{K^-_{CA}}(VS, K^+_{VS})))$
6. $SM \rightarrow Ser_E : VS, Ser_E, \mathbf{S}_{K^-_{VS}}(\mathbf{P}_{K^+_{Ser_E}}(SN, Bta, \mathbf{S}_{K^-_{CA}}(VS, K^+_{VS})))$

(EVALUATION AND REPLY)

7. $Ser_E \rightarrow SM : Ser_E, VS, \mathbf{S}_{K^-_{Ser_E}}(\mathbf{P}_{K^+_{VS}}(SN, Res))$
8. $SM \rightarrow VS : Ser_E, VS, \mathbf{S}_{K^-_{Ser_E}}(\mathbf{P}_{K^+_{VS}}(SN, Res))$
9. $VS \rightarrow C : \mathbf{E}_{\mathbf{H}(N_C, N_{VS}, N_{PMS})}(Bta, Res)$

MT-LAB
A VKR CENTRE OF EXCELLENCE

(2-4) The CaPiTo Approach: Choice of Models

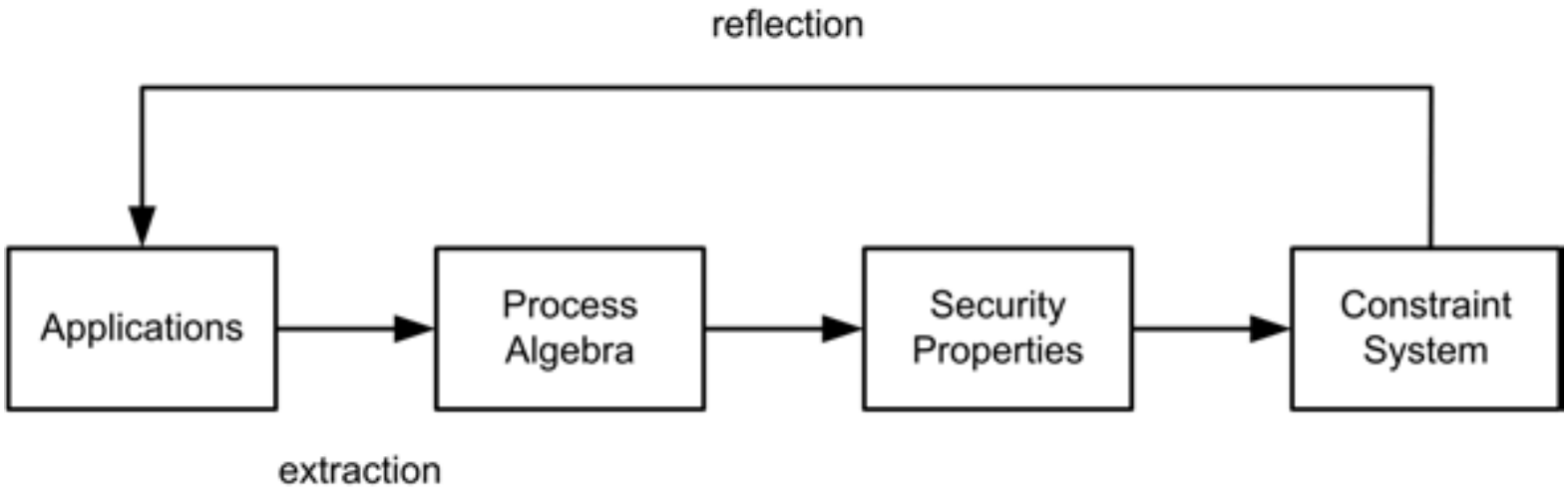
Service Oriented Computing can be described at many levels:

- Software Systems
  - Programming Languages (Java/F#)
- Software Designs
  - Object Oriented Models (UML)
- Software Processes
  - Process Calculi (CCS,π,spi, LySa,...,COWS,CaSPis,...)

What is an appropriate level for bridging the gap between theoretical analysis and practical implementation?

MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Approach: The Vision



MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Approach: Bridging the Gap

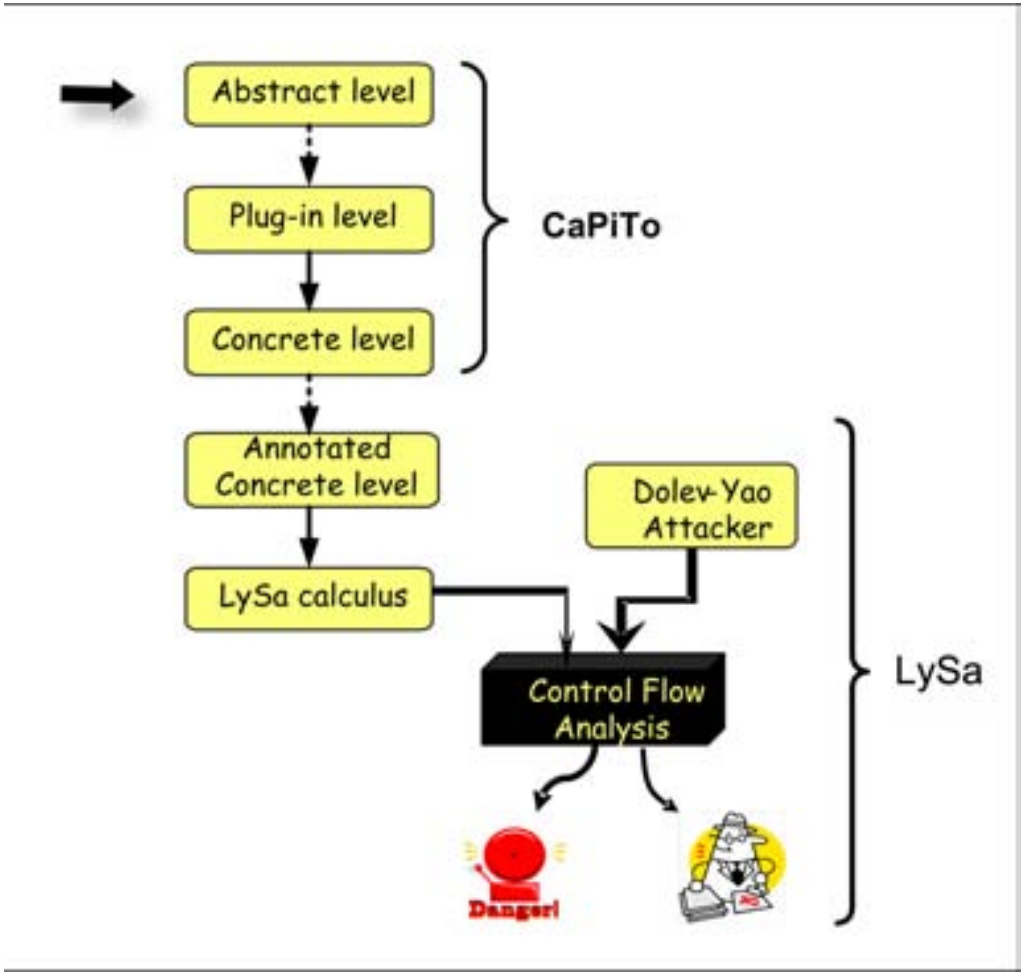
The CaPiTo Approach to Protocol Validation is to model protocols in Service Oriented Computing at different levels

- Abstract level (as often used for theoretical analysis)
- Protocol stack plug-in level (bridge the gap between the two levels)
- Concrete level (as often used for practical implementation)

We then transform the model into LySa calculus and use the static analyser LySa tool to analyse the specification; however, other "backends" can be used as well.

MT-LAB
A VKR CENTRE OF EXCELLENCE

(2) The CaPiTo Abstract Level



MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Abstract Level: Syntax

Process	P	$::=$	$\langle \vec{e} \rangle.P$	output
		$\mid$	$(\vec{p}).P$	input
		$\mid$	$(\nu\ n)P$	restriction
		$\mid$	$P_1 \mid P_2$	parallel
		$\mid$	$!P$	replication
		$\mid$	$P_1 + P_2$	nondeterministic choice
		$\mid$	0	nil
		$\mid$	$\overline{n}[\ ] \cdot P$	service invocation
		$\mid$	$n[\ ] \cdot P$	service provider
		$\mid$	$\uparrow \langle \vec{e} \rangle.P$	return

- $n[\ ] \cdot P$ provides a service n that is to be invoked by $\overline{n}[\ ] \cdot P$
- $\uparrow \langle \vec{e} \rangle.P$ returns $\vec{e}$ to the outside environment

MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Abstract Level: Syntax

value	v, w	$::=$	n	name
		$\mid$	$f(\vec{v})$	function
expr	e	$::=$	x	variable
		$\mid$	n	name
		$\mid$	$f(\vec{e})$	function
pattern	p	$::=$	$?x$	defining occurrence of variable
		$\mid$	x	applied occurrence of variable
		$\mid$	n	name

- Abstract away from both cryptography and industrial communication protocols
- Concentrate on the interaction of the services themselves.
- Distinguish between defining occurrences and applied occurrences of variables.

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Abstract Level

$$System \triangleq Client \mid Bank \mid Ser_E$$

$$Client \triangleq !\,(\nu\ Bta)\ \overline{credit\_req}[\,] \cdot \Big(\langle Bta\rangle.(Bta, ?x_{res}).\uparrow\langle x_{res}\rangle.0\Big)$$

$$Bank \triangleq !\ credit\_req[\,] \cdot \Big((?y_{bta}).\ \overline{validate}[\,] \cdot \langle y_{bta}\rangle.(?y_{res}).\uparrow\langle y_{bta}, y_{res}\rangle.0\Big)$$

$$Ser_E \triangleq !\ validate[\,] \cdot \Big((?z_{bta}).\langle isValid(z_{bta})\rangle.0\Big)$$

Bta : Balance Total Assets

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Abstract Level

$$System \triangleq Client \mid Bank \mid Ser_E$$

$$Client \triangleq !\,(\nu\ Bta)\ \overline{credit\_req}[\,] \cdot \Big(\langle Bta \rangle.(Bta, ?x_{res}).\uparrow \langle x_{res} \rangle.0 \Big)$$

$$Bank \triangleq !\ credit\_req[\,] \cdot \Big((?y_{bta}). \overline{validate}[\,] \cdot \langle y_{bta} \rangle.(?y_{res}).\uparrow \langle y_{bta}, y_{res} \rangle.0 \Big)$$

$$Ser_E \triangleq !\ validate[\,] \cdot \Big((?z_{bta}).\langle isValid(z_{bta}) \rangle.0 \Big)$$

Bta : Balance Total Assets

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Abstract Level

$$System \triangleq Client \mid Bank \mid Ser_E$$

$$Client \triangleq !\,(\nu\ Bta)\ \overline{credit\_req}[\,] \cdot \Big(\langle Bta \rangle.(Bta, ?x_{res}).\uparrow \langle x_{res} \rangle.0 \Big)$$

$$Bank \triangleq !\ credit\_req[\,] \cdot \Big((?y_{bta}).\ \overline{validate}[\,] \cdot \langle Bta \rangle.(?y_{res}).\uparrow \langle Bta, y_{res} \rangle.0 \Big)$$

$$Ser_E \triangleq !\ validate[\,] \cdot \Big((?z_{bta}).\langle isValid(z_{bta}) \rangle.0 \Big)$$

Bta : Balance Total Assets

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Abstract Level

$$System \triangleq Client \mid Bank \mid Ser_E$$

$$Client \triangleq !\,(\nu\ Bta)\ \overline{credit\_req}[\,] \cdot \Big(\langle Bta \rangle . (Bta, ?x_{res}) . \uparrow \langle x_{res} \rangle . 0 \Big)$$

$$Bank \triangleq !\ credit\_req[\,] \cdot \Big((?y_{bta}) . \overline{validate}[\,] \cdot \langle Bta \rangle . (?y_{res}) . \uparrow \langle Bta, y_{res} \rangle . 0 \Big)$$

$$Ser_E \triangleq !\ validate[\,] \cdot \Big((?z_{bta}) . \langle isValid(Bta) \rangle . 0 \Big)$$

Bta : Balance Total Assets

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Abstract Level

$$System \triangleq Client \mid Bank \mid Ser_E$$

$$Client \triangleq !\,(\nu\ Bta)\ \overline{credit\_req}[\,] \cdot \Big(\langle Bta\rangle.(Bta, ?x_{res}).\uparrow\langle x_{res}\rangle.0\Big)$$

$$Bank \triangleq !\ credit\_req[\,] \cdot \Big((?y_{bta}).\ \overline{validate}[\,] \cdot \langle Bta\rangle.(?y_{res}).\uparrow\langle Bta, isValid(Bta)\rangle.0\Big)$$

$$Ser_E \triangleq !\ validate[\,] \cdot \Big((?z_{bta}).\langle isValid(Bta)\rangle.0\Big)$$

Bta : Balance Total Assets

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Abstract Level

$$System \triangleq Client \mid Bank \mid Ser_E$$

$$Client \triangleq !\,(\nu\ Bta)\ \overline{credit\_req}[\,] \cdot \Big(\langle Bta\rangle.(Bta, ?x_{res}).\uparrow\langle x_{res}\rangle.0\Big)$$

$$Bank \triangleq !\ credit\_req[\,] \cdot \Big((?y_{bta}).\ \overline{validate}[\,]\cdot\langle Bta\rangle.(?y_{res}).\uparrow\langle Bta, isValid(Bta)\rangle.0\Big)$$

$$Ser_E \triangleq !\ validate[\,] \cdot \Big((?z_{bta}).\langle isValid(Bta)\rangle.0\Big)$$

Bta : Balance Total Assets

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Abstract Level

$$System \triangleq Client \mid Bank \mid Ser_E$$

$$Client \triangleq !\,(\nu\ Bta)\ \overline{credit\_req}[\,] \cdot \Big(\langle Bta\rangle.(Bta, ?x_{res}).\uparrow \langle isValid\,Bta\rangle.0\Big)$$

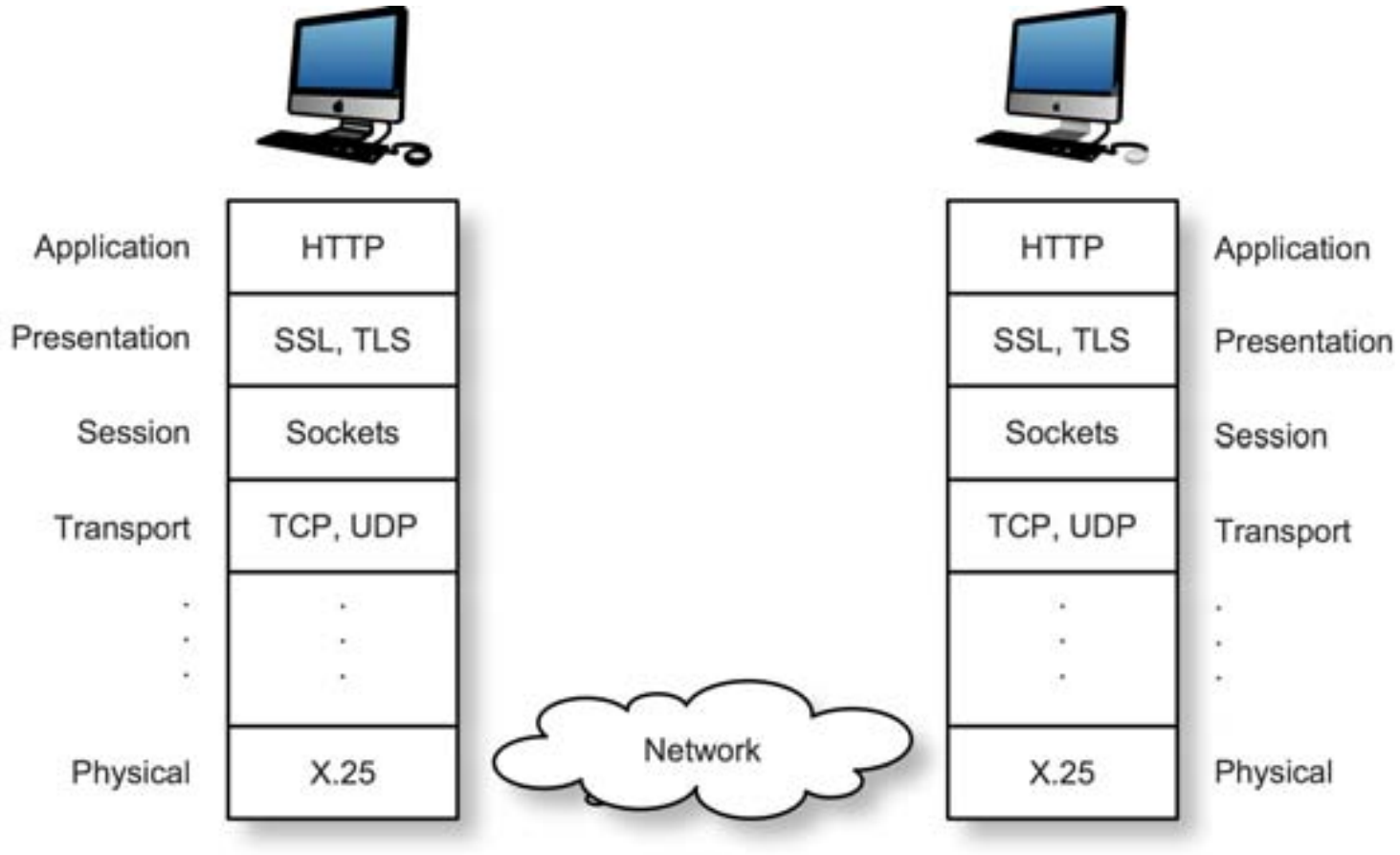
$$Bank \triangleq !\ credit\_req[\,] \cdot \Big((?y_{bta}).\ \overline{validate}[\,] \cdot \langle Bta\rangle.(?y_{res}).\uparrow \langle Bta, isValid(Bta)\rangle.0\Big)$$

$$Ser_E \triangleq !\ validate[\,] \cdot \Big((?z_{bta}).\langle isValid(Bta)\rangle.0\Big)$$

Bta : Balance Total Assets

MT-LAB
A VKR CENTRE OF EXCELLENCE

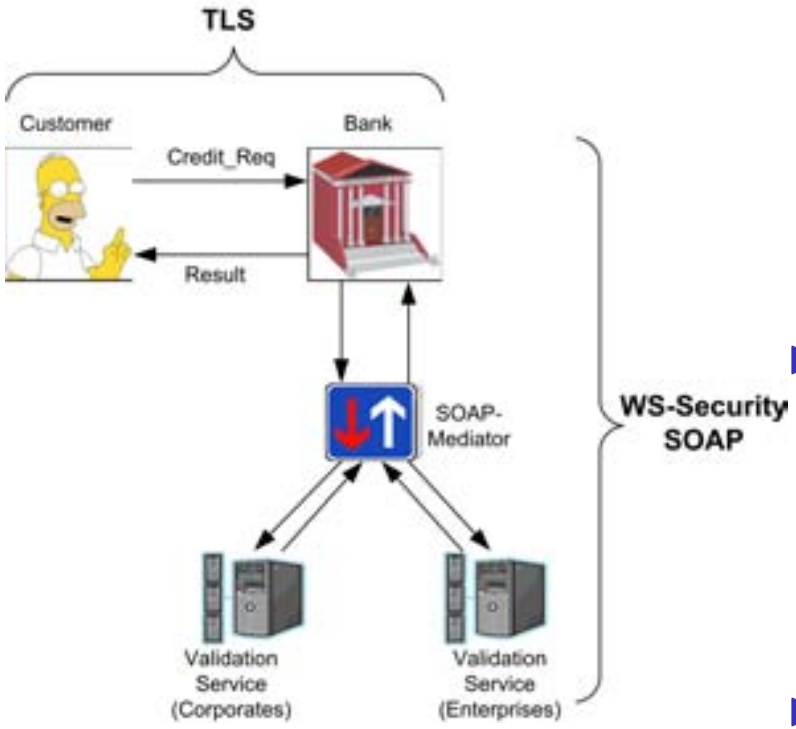
(3) Background: Protocol Stacks



MT-LAB
A VKR CENTRE OF EXCELLENCE

Background: The Protocols

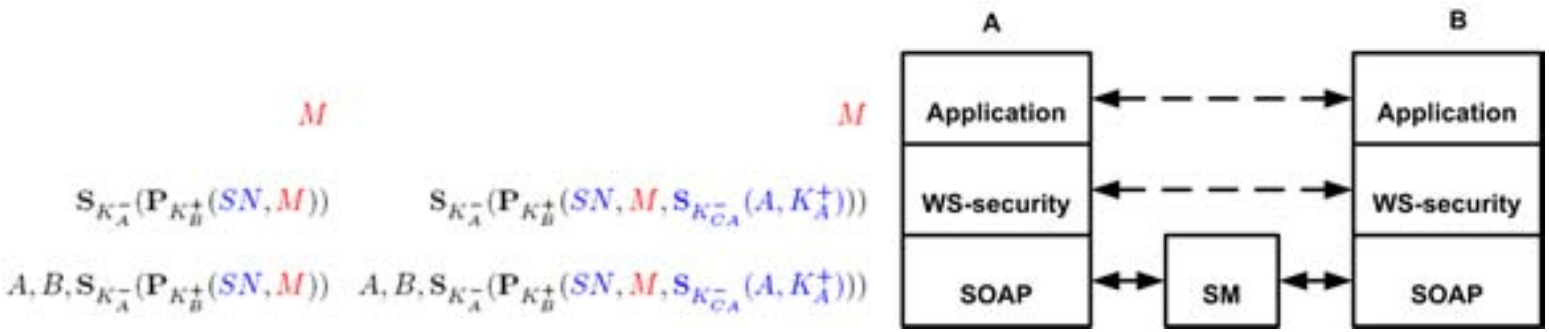
Go one step further and consider the various protocols needed to secure the communication.



- WS-Security (Web Service Security): a communication protocol suite providing security to Web services, while guaranteeing point-to-point (P2P) integrity and authenticity.
- SOAP (Simple Object Access Protocol): a protocol specification for exchanging structured information in computer networks.
- TLS (Transport Layer Security): a key establishment protocol.

MT-LAB
A VKR CENTRE OF EXCELLENCE

Background: The WS-Security and SOAP Protocols



- A is sending a message M to B
- WS-Security ($P2P$ or $P2P^+$) signs the encrypted messages to ensure integrity and confidentiality
  - SN: a sequence number for correlation
  - $P2P^+$ also inserts A's certificate in the message in case it is unknown to B, to allow B to reply.
- SOAP gives a standardised format for the messages and adds the message header to allow SOAP-routing

MT-LAB
A VKR CENTRE OF EXCELLENCE

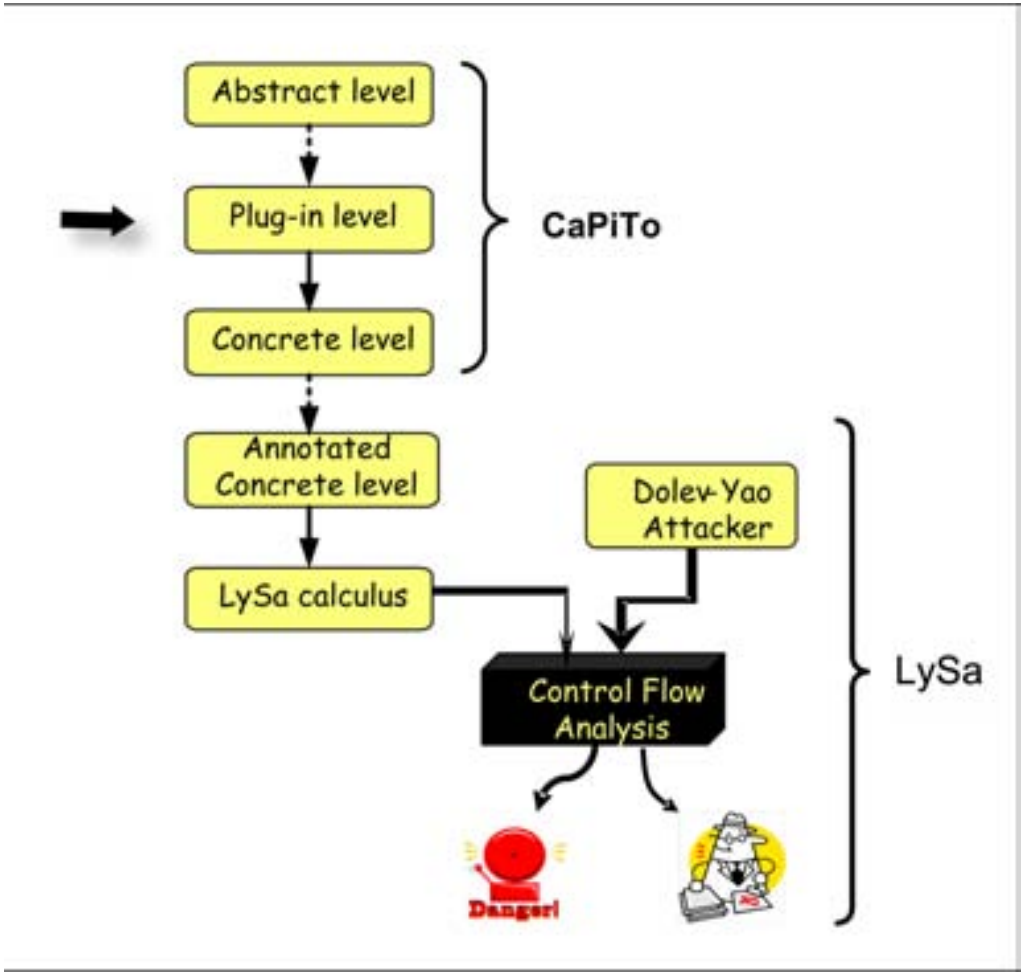
Background: The TLS Protocol

The TLS protocol is summarised by the following protocol narration taking place between the client C and a server (Bank) S holding a certificate $\mathbf{S}_{K_{CA}^-}(S, K_S^+)$ issued by a mutually trusted Certificate Authority CA:

$$
\begin{array}{llcll}
1. & C & \rightarrow & S & : \quad N_C \\
2. & S & \rightarrow & C & : \quad N_S, \mathbf{S}_{K_{CA}^-}(S, K_S^+) \\
3. & C & \rightarrow & S & : \quad \mathbf{P}_{K_S^+}(N) \\
4. & C & \rightarrow & S & : \quad \mathbf{E}_{\mathbf{H}(N_C, N_S, N)}(M)
\end{array}
$$

MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Plug-in Level



MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Plug-in Level: Syntax

We wish to be precise about which protocols we intend to use.

$$
\begin{array}{lcl}
v, w & ::= & n \mid f(\vec{v}) \\
e & ::= & x \mid n \mid f(\vec{e}) \\
p & ::= & ?x \mid x \mid n \\
P & ::= & \langle \vec{e} \rangle.P \mid (\vec{p}).P \mid \\
 & & (\nu\ n)P \mid P_1 | P_2 \mid !P \mid P_1 + P_2 \mid 0 \mid \\
 & & \overline{n}[ps].P \mid n[ps].P \mid \uparrow \langle \vec{e} \rangle.P \\
ps & ::= & pi \mid pi; ps \\
pi & ::= & name, param_1, \cdots, param_k
\end{array}
$$

- ps is a protocol stack containing a list of protocols to be used.
- each protocol pi is identified by its name and a list of parameters

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Plug-in Level

$$Client \triangleq\ !\,(\nu\ Bta)\overline{credit\_req}[TLS, Client, Bank].\Big(\langle Bta\rangle.(Bta, ?x_{res}).\uparrow\langle x_{res}\rangle.0\Big)$$

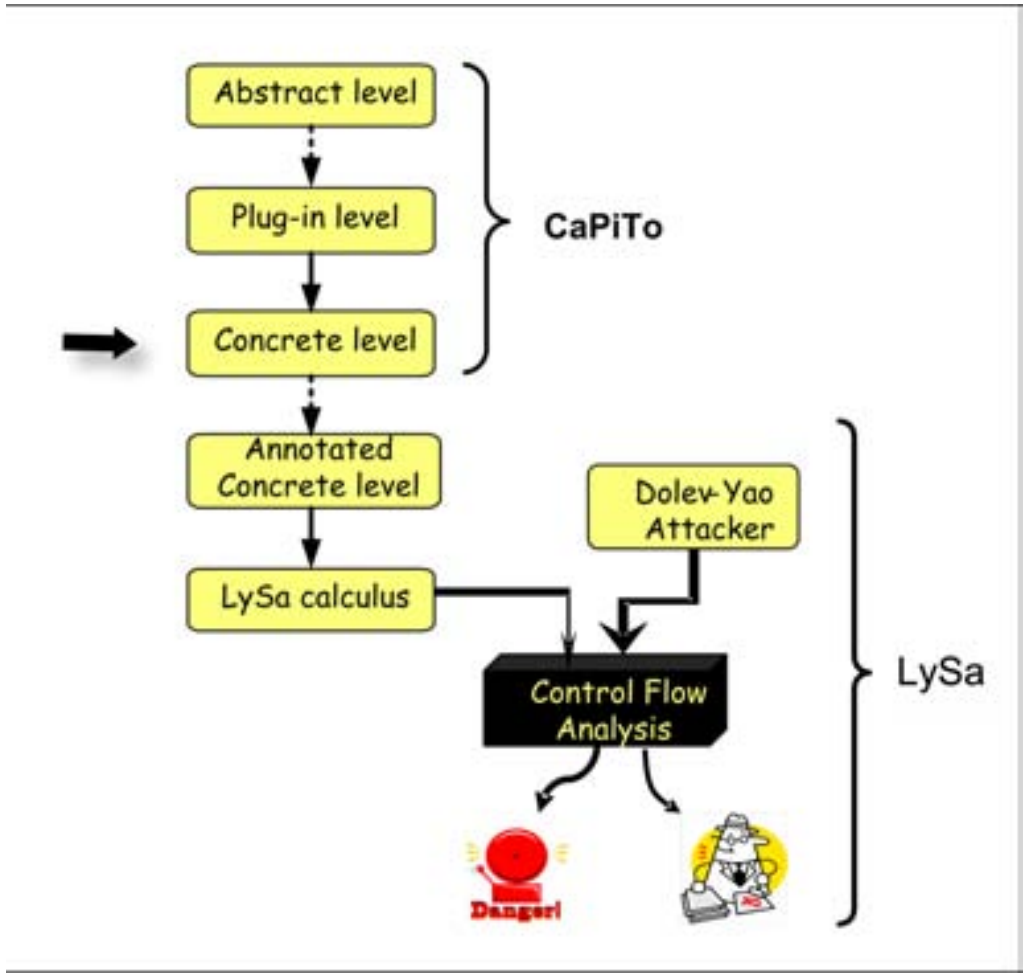
$$Bank \triangleq\ !\ credit\_req[TLS, Client, Bank].(?y_{bta}).\overline{validate}[P2P^{+}, Bank, Ser_E; SOAP, Bank, SM, Ser_E].\Big(\langle y_{bta}\rangle.(?y_{res}).\uparrow\langle y_{bta}, y_{res}\rangle.0\Big)$$

$$Ser_E \triangleq\ !\ validate[P2P^{+}, Bank, Ser_E; SOAP, Bank, SM, Ser_E].\Big((?z_{bta}).\langle isValid(z_{bta})\rangle.0\Big)$$

$$SM \triangleq\ !\,(?A, SM, A, ?B, ?M).\langle SM, B, A, B, M\rangle$$

MT-LAB
A VKR CENTRE OF EXCELLENCE

(4) The CaPiTo Concrete Level



MT-LAB
A VKR CENTRE OF EXCELLENCE

Concrete Level Syntax

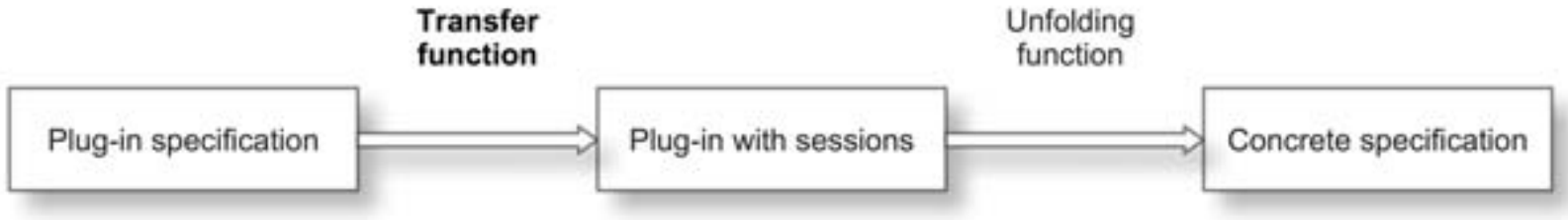
- Processes:

$$
\begin{array}{lll}
P & ::= & \langle r, \vec{e} \rangle.P \mid (r, \vec{p}).P \mid (\nu\ n)P \mid \\
 & & !P \mid P_1 + P_2 \mid P_1 | P_2 \mid 0 \mid \\
 & & \overline{n}[\ ].P \mid n[\ ].P \mid \uparrow \langle \vec{e} \rangle.P \mid \\
 & & (\nu_{\pm}\ n)P \mid \\
 & & r : e \blacktriangleright P
\end{array}
$$

- Cryptographic features:
  - Asymmetric encryption $\mathbf{P}_{v^+}(\vec{v})$ and decryption $\underline{\mathbf{P}}_{v^-}(\vec{v})$
  - Digital signature $\mathbf{S}_{v^-}(\vec{v})$ and verification $\underline{\mathbf{S}}_{v^+}(\vec{v})$
  - Hash function $\mathbf{H}(\vec{v})$
  - Symmetric <u>tunnel</u> $r : e \blacktriangleright P$ where e is the symmetric key

MT-LAB
A VKR CENTRE OF EXCELLENCE

From Plug-in Level to Concrete Level (1)



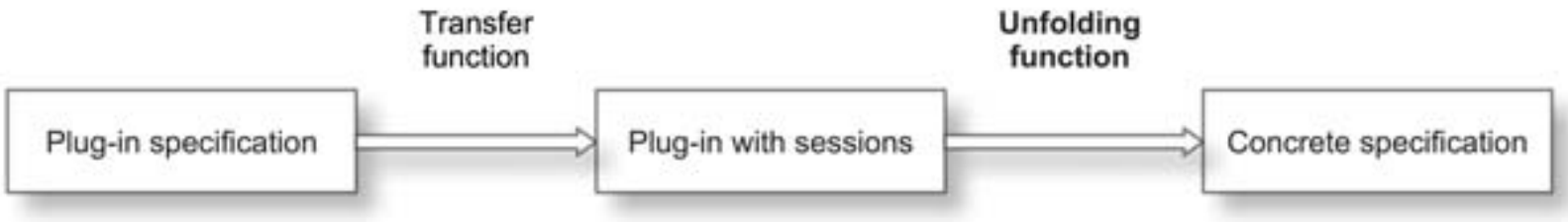
- Distinguish between different services as well as different sessions of each service by using a unique session identifier r

$$
\begin{array}{lcl}
\mathcal{T}(\bar{s}[ps] \cdot P, rl) & \triangleq & (\nu\ r)\langle r_{env}, s, r\rangle.\bar{r}[ps] \rhd \mathcal{T}(P, r :: rl) \\
\mathcal{T}(s[ps] \cdot P, rl) & \triangleq & (r_{env}, s, ?r).r[ps] \rhd \mathcal{T}(P, r :: rl) \\
\mathcal{T}(\langle \vec{e} \rangle.P, r :: rl) & \triangleq & \langle r, \vec{e} \rangle.\mathcal{T}(P, r :: rl) \\
\mathcal{T}((\vec{p}).P, r :: rl) & \triangleq & (r, \vec{p}).\mathcal{T}(P, r : rl) \\
\mathcal{T}(\uparrow \langle \vec{e} \rangle.P, r_1 :: r_2 :: rl) & \triangleq & \langle r_2, \vec{e} \rangle.\mathcal{T}(P, r_1 :: r_2 :: rl)
\end{array}
$$

MT-LAB
A VKR CENTRE OF EXCELLENCE

- Intermediate Result: $s[pi; pi'] \cdot P \Rightarrow r[pi; pi'] \rhd P$

From Plug-in Level to Concrete Level (2)



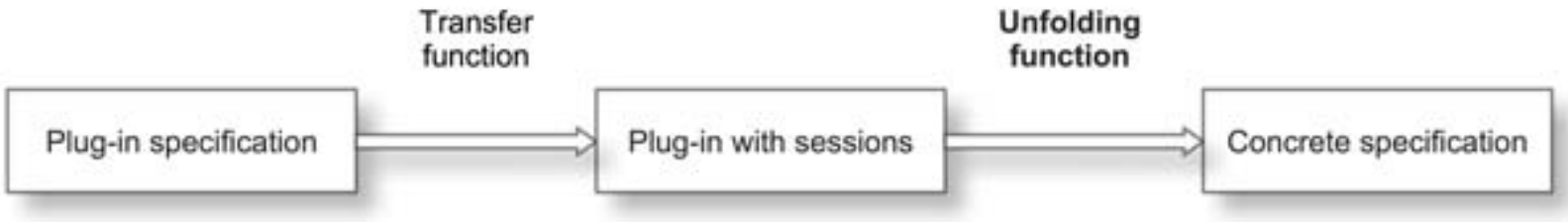
- When a protocol stack contains a sequence of protocols we first expand away the leftmost (topmost) protocol in the stack and then continue with the subsequent layers

$$\overline{r}[pi_1; \ldots; pi_k] \rhd P \triangleq \overline{r}[pi_k] \rhd (\ldots(\overline{r}[pi_1] \rhd P))$$
$$r[pi_1; \ldots; pi_k] \rhd P \triangleq r[pi_k] \rhd (\ldots(r[pi_1] \rhd P))$$

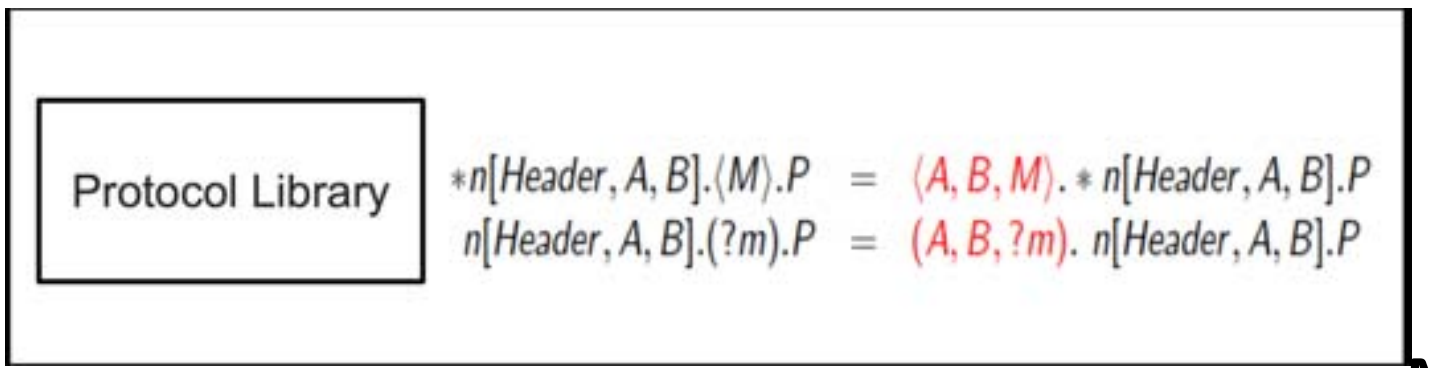
- Intermediate Result: $s[pi; pi'] \cdot P \Rightarrow r[pi] \rhd (r[pi'] \rhd P)$

MT-LAB
A VKR CENTRE OF EXCELLENCE

From Plug-in Level to Concrete Level (3)



- The individual protocols are taken from a library.



MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Concrete TLS Protocol

A reminder of the TLS protocol narration:

- Explicit about the creation of nonces and the checking of certificates
- After the completion, the symmetric key $\mathbf{H}(N_C, N_S, N)$ is computed

$$
\begin{array}{lcccl}
1. & C & \rightarrow & S & : \quad N_C \\
2. & S & \rightarrow & C & : \quad N_S, \mathbf{S}_{K^-_{CA}}(S, K^+_S) \\
3. & C & \rightarrow & S & : \quad \mathbf{P}_{K^+_S}(N) \\
4. & C & \rightarrow & S & : \quad \mathbf{E}_{\mathbf{H}(N_C,N_S,N)}(M)
\end{array}
$$

MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Concrete TLS Protocol

The TLS protocol in the CaPiTo Library:

$$\begin{aligned}\overline{r}[TLS, C, S, CA] \triangleright (\langle M\rangle.P) \triangleq\ & (\nu\ N_C)\langle r, C, S, N_C\rangle.\\ & (r, S, C, ?n_S, \underline{\mathbf{S}}_{K^+_{CA}}(S, ?x_{ks})).\\ & (\nu\ N)\langle r, C, S, \mathbf{P}_{x_{ks}}(N)\rangle.\\ & r : \mathbf{H}(N_C, n_S, N) \blacktriangleright (\langle M\rangle.P)\end{aligned}$$

$$\begin{aligned}r[TLS, C, S, CA] \triangleright ((?m).P) \triangleq\ & (r, C, S, ?n_C).\\ & (\nu\ N_S)\langle r, S, C, N_S, \mathbf{S}_{K^-_{CA}}(S, K^+_S)\rangle.\\ & (r, C, S, \underline{\mathbf{P}}_{K^-_S}(?n)).\\ & r : \mathbf{H}(n_C, N_S, n) \blacktriangleright ((?m).P)\end{aligned}$$

parameterised on the session identifier r

MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Concrete TLS Protocol

The TLS protocol in the CaPiTo Library:

$$\begin{aligned}\overline{r}[TLS, C, S, CA] \triangleright (\langle M\rangle.P) \triangleq\ & (\nu\ N_C)\langle r, C, S, N_C\rangle.\\ & (r, S, C, ?n_S, \underline{\mathbf{S}}_{K^+_{CA}}(S, ?x_{ks})).\\ & (\nu\ N)\langle r, C, S, \mathbf{P}_{x_{ks}}(N)\rangle.\\ & r : \mathbf{H}(N_C, n_S, N) \blacktriangleright (\langle M\rangle.P)\end{aligned}$$

$$\begin{aligned}r[TLS, C, S, CA] \triangleright ((?m).P) \triangleq\ & (r, C, S, ?n_C).\\ & (\nu\ N_S)\langle r, S, C, N_S, \mathbf{S}_{K^-_{CA}}(S, K^+_S)\rangle.\\ & (r, C, S, \underline{\mathbf{P}}_{K^-_S}(?n)).\\ & r : \mathbf{H}(n_C, N_S, n) \blacktriangleright ((?m).P)\end{aligned}$$

parameterised on the session identifier r

MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Concrete TLS Protocol

The TLS protocol in the CaPiTo Library:

$$\begin{aligned}\overline{r}[TLS, C, S, CA] \triangleright (\langle M\rangle.P) \triangleq\ & (\nu\ N_C)\langle r, C, S, N_C\rangle.\\ & (r, S, C, ?n_S, \underline{\mathbf{S}}_{K^+_{CA}}(S, ?x_{ks})).\\ & (\nu\ N)\langle r, C, S, \mathbf{P}_{x_{ks}}(N)\rangle.\\ & r : \mathbf{H}(N_C, n_S, N) \blacktriangleright (\langle M\rangle.P)\end{aligned}$$

$$\begin{aligned}r[TLS, C, S, CA] \triangleright ((?m).P) \triangleq\ & (r, C, S, ?n_C).\\ & (\nu\ N_S)\langle r, S, C, N_S, \mathbf{S}_{K^-_{CA}}(S, K^+_S)\rangle.\\ & (r, C, S, \underline{\mathbf{P}}_{K^-_S}(?n)).\\ & r : \mathbf{H}(n_C, N_S, n) \blacktriangleright ((?m).P)\end{aligned}$$

parameterised on the session identifier r

MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Concrete TLS Protocol

The TLS protocol in the CaPiTo Library:

$$\begin{aligned}
\overline{r}[TLS, C, S, CA] \triangleright (\langle M \rangle.P) \triangleq\ & (\nu\ N_C)\langle r, C, S, N_C \rangle. \\
& (r, S, C, ?n_S, \underline{\mathbf{S}}_{K^+_{CA}}(S, ?x_{ks})). \\
& (\nu\ N)\langle r, C, S, \mathbf{P}_{x_{ks}}(N) \rangle. \\
& r : \mathbf{H}(N_C, n_S, N) \blacktriangleright (\langle M \rangle.P)
\end{aligned}$$

$$\begin{aligned}
r[TLS, C, S, CA] \triangleright ((?m).P) \triangleq\ & (r, C, S, ?n_C). \\
& (\nu\ N_S)\langle r, S, C, N_S, \mathbf{S}_{K^-_{CA}}(S, K^+_S) \rangle. \\
& (r, C, S, \underline{\mathbf{P}}_{K^-_S}(?n)). \\
& r : \mathbf{H}(n_C, N_S, n) \blacktriangleright ((?m).P)
\end{aligned}$$

parameterised on the session identifier r

MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Concrete WS-Security Protocol

Sender sends out message M: $A \longrightarrow B : M$.

- $P2P^{+}$: encrypts and signs the message together with a sequence number SN and sender's certificate $\mathbf{S}_{K_{CA}^{-}}(A, K_A^{+})$ to allow receiver to reply.

$$\overline{r}[P2P^{+}, S, R, CA] \triangleright (\langle r, \vec{e} \rangle.P) \triangleq$$
$$(\nu\ SN)\langle r, \mathbf{S}_{K_A^{-}}(\mathbf{P}_{K_B^{+}}(SN, \vec{e}, \mathbf{S}_{K_{CA}^{-}}(A, K_A^{+})))\rangle.$$
$$\overline{r}[P2P, S, R, SN] \triangleright P$$

- $P2P$: encrypts and signs the message together with a sequence number SN.

$$\overline{r}[P2P, S, R, SN] \triangleright (\langle r, \vec{e} \rangle.P) \triangleq$$
$$\langle r, \mathbf{S}_{K_A^{-}}(\mathbf{P}_{K_B^{+}}(SN, \vec{e}))\rangle.$$
$$\overline{r}[P2P, S, R, SN] \triangleright P$$

MT-LAB
A VKR CENTRE OF EXCELLENCE

The CaPiTo Concrete *SOAP* Protocol

- SOAP: appends additional fields (headers) to messages.

$$\overline{r}[SOAP, S, SM, R] \rhd (\langle r, \vec{e} \rangle.P) \triangleq \langle r, S, SM, S, R, \vec{e} \rangle. \overline{r}[SOAP, S, SM, R] \rhd P$$

$$\overline{r}[SOAP, S, SM, R] \rhd ((r, \vec{p}).P) \triangleq (r, SM, S, R, S, \vec{p}). \overline{r}[SOAP, S, SM, R] \rhd P$$

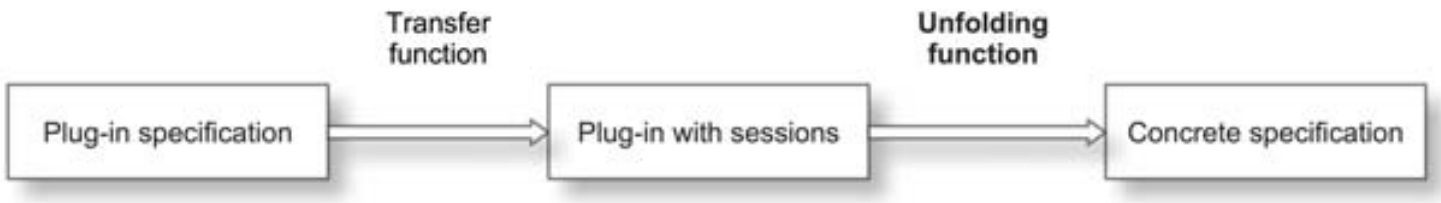
MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Concrete Level

Recall the Plug-in Level specification of the Bank:

$$Bank \triangleq\ !\ \overline{credit\_req}[TLS, Client, Bank].(?y_{bta}).\ \overline{validate}[WS, Bank, Ser_E; SOAP, Bank, SM, Ser_E].\ \Big(\langle y_{bta}\rangle.(?y_{res}).\uparrow \langle y_{bta}, y_{res}\rangle.0\Big)$$

We next transform it into the Concrete Level:



MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Concrete Level

The resulting Concrete specification of the Bank:

$$
\begin{array}{l}
(r_{env}, req, ?r_1).(r_1, Client, Bank, ?n_C). \\
\quad (\nu\ N_S)\langle r_1, Bank, Client, N_S, \mathbf{S}_{K^-_{CA}}(Bank, K^+_{Bank})\rangle \\
\quad\quad (r_1, Client, Bank, \underline{\mathbf{P}}_{K^-_{Bank}}(?n)). \\
\quad\quad\quad r_1 : \mathbf{H}(n_C, N_S, n) \blacktriangleright (\\
\quad\quad\quad\quad (r_1, ?y_{bta}). \\
\quad\quad\quad\quad\quad (\nu\ r_2)\langle r_{env}, val, r_2\rangle.(\nu\ SN) \\
\quad\quad\quad\quad\quad\quad \langle r_2, Bank, SM, Bank, Ser_E, \\
\quad\quad\quad\quad\quad\quad\quad\quad \mathbf{S}_{K^-_{Bank}}(\mathbf{P}_{K^+_{Ser_E}}(SN, y_{bta}, \mathbf{S}_{K^-_{CA}}(Bank, K^+_{Bank})))\rangle. \\
\quad\quad\quad\quad\quad\quad\quad (r_2, SM, Bank, Ser_E, Bank, \underline{\mathbf{S}}_{K^+_{Ser_E}}(\underline{\mathbf{P}}_{K^-_{Bank}}(SN, ?y_r))). \\
\quad\quad\quad\quad\quad\quad\quad \langle r_1, y_{bta}, y_r\rangle.0)
\end{array}
$$

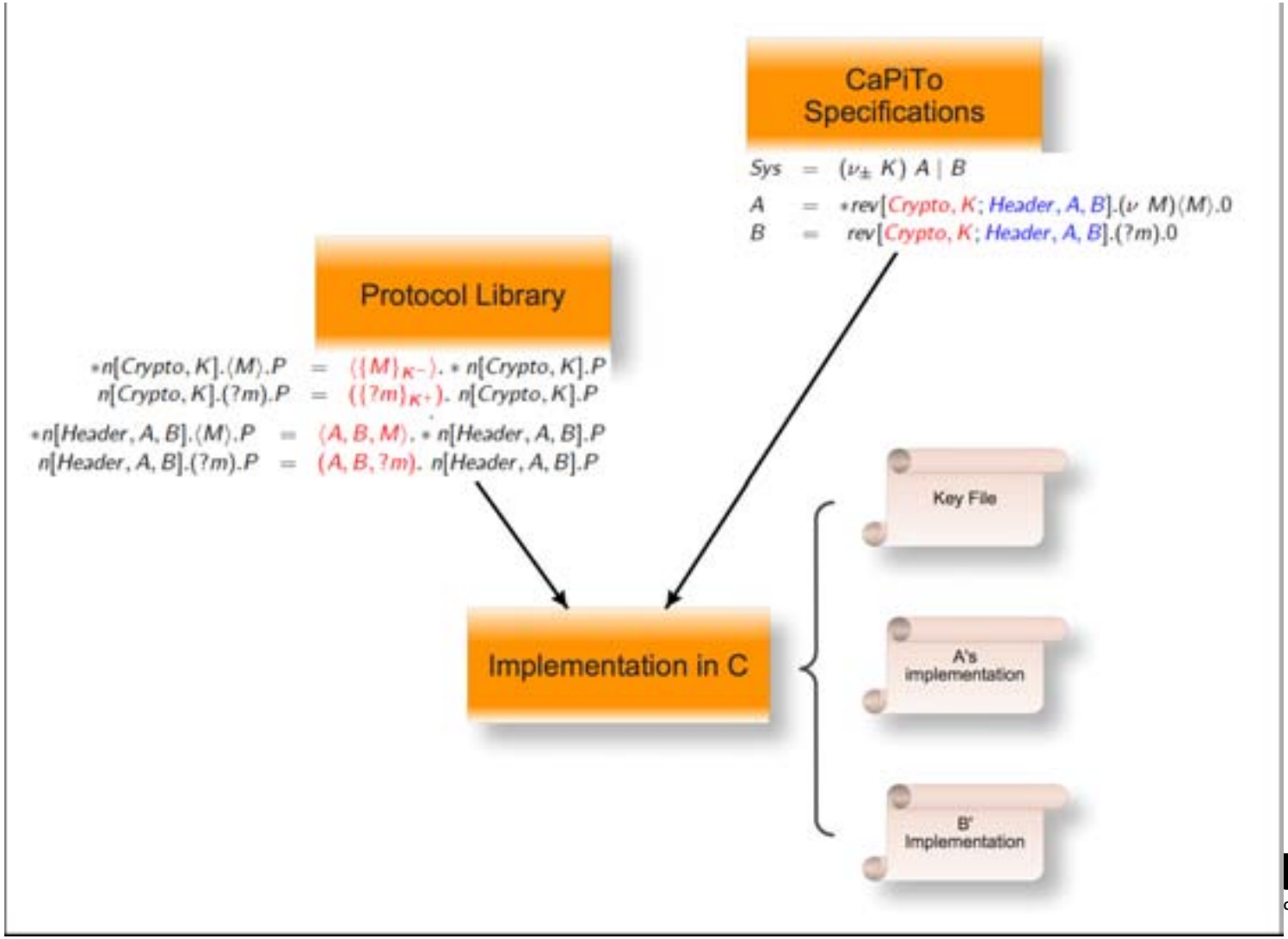
MT-LAB
A VKR CENTRE OF EXCELLENCE

Note that tunnels are still present.

(5) Code Stubs from CaPiTo

MT-LAB
A VKR CENTRE OF EXCELLENCE

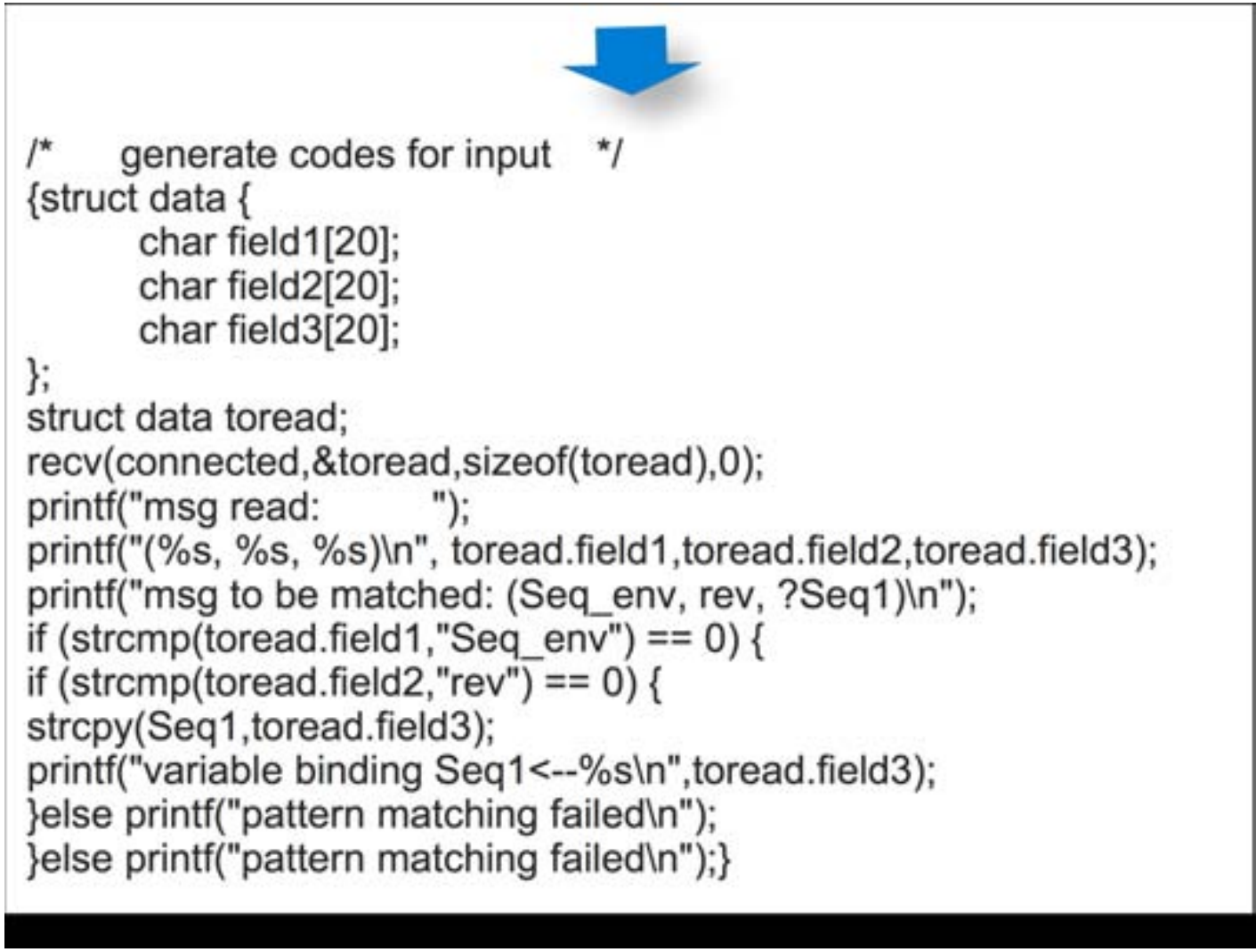
Code Stubs from CaPiTo



IT-LAB
CENTRE OF EXCELLENCE

Code Stubs from CaPiTo

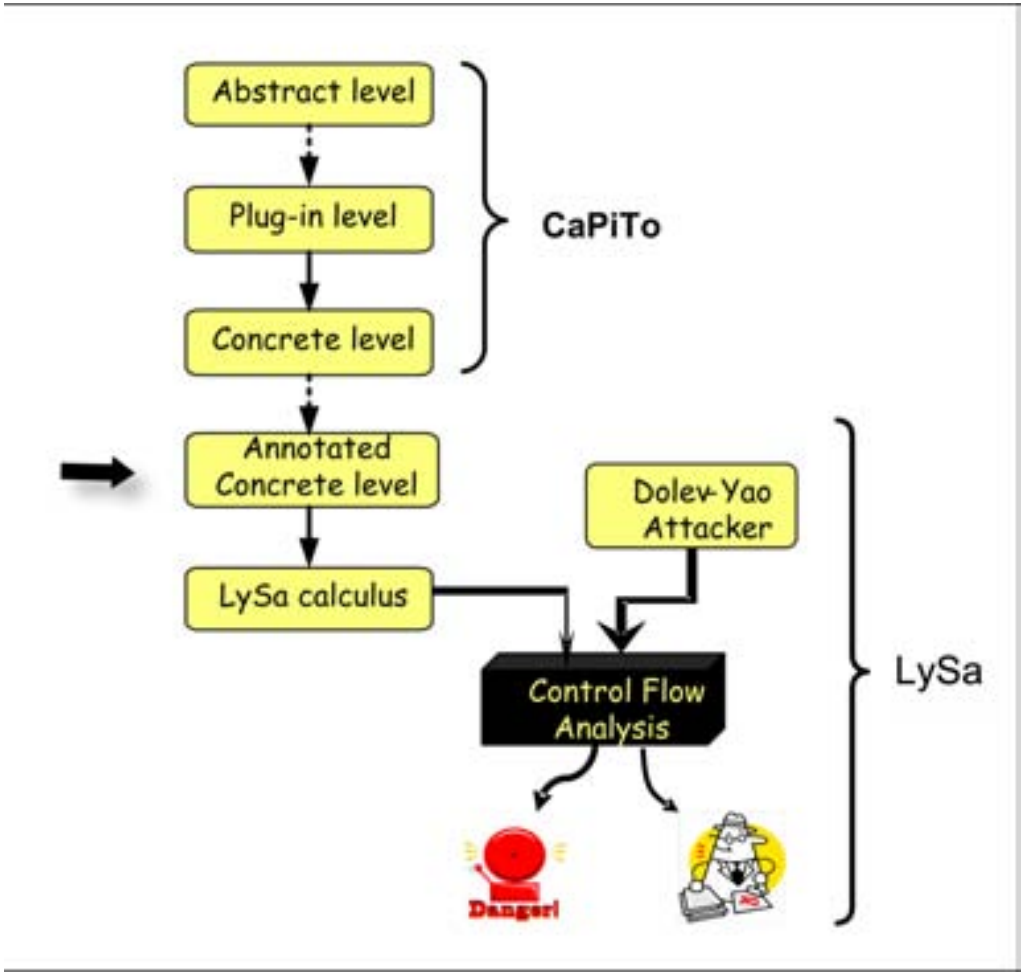
$\ldots(Seq_{env}, rev, ?Seq1).0$



```
/*      generate codes for input    */
{struct data {
        char field1[20];
        char field2[20];
        char field3[20];
};
struct data toread;
recv(connected,&toread,sizeof(toread),0);
printf("msg read:          ");
printf("(%s, %s, %s)\n", toread.field1,toread.field2,toread.field3);
printf("msg to be matched: (Seq_env, rev, ?Seq1)\n");
if (strcmp(toread.field1,"Seq_env") == 0) {
if (strcmp(toread.field2,"rev") == 0) {
strcpy(Seq1,toread.field3);
printf("variable binding Seq1<--%s\n",toread.field3);
}else printf("pattern matching failed\n");
}else printf("pattern matching failed\n");}
```

MT-LAB
A VKR CENTRE OF EXCELLENCE

(6) CaPiTo with Annotations from LySa



MT-LAB
A VKR CENTRE OF EXCELLENCE

CaPiTo with Annotations from LySa

The focus is ensuring the authenticity:

- When encryptions are created: specify where they are intended to be decrypted.

$$r : e \blacktriangleright \langle V_1, V_2 \rangle^{l_1} [dest\ \mathcal{L}_2].P$$

- When encryptions are decrypted: specify where they are expected to have been encrypted.

$$r : e \blacktriangleright (V_1, x)^{l_2} [orig\ \mathcal{L}_1].P$$

Here l_1 and l_2 are crypto-points and $\mathcal{L}_1$ and $\mathcal{L}_2$ are sets of crypto-points. Authenticity amounts to $l_1 \in \mathcal{L}_2$ and $l_2 \in \mathcal{L}_1$.

Similar annotations are made for asymmetric encryption, signature, decryption and signature validation.

MT-LAB
A VKR CENTRE OF EXCELLENCE

CaPiTo with Annotations from LySa

Review of LySa:

- Expressions

$$\begin{array}{rcl} E & ::= & n \\ & | & x \\ & | & \{E_1, \ldots, E_k\}^{l}_{E_0}[dest\ \mathcal{L}] \end{array}$$

- Processes

$$\begin{array}{rcl} P & ::= & \langle E_1, \ldots, E_k \rangle.P \\ & | & (E_1, \ldots, E_j; x_{j+1}, \ldots, x_k).P \\ & | & \text{decrypt } E \text{ as } \{E_1, \ldots, E_j; x_{j+1}, \ldots, x_k\}^{l}_{E_0}[orig\ \mathcal{L}] \text{ in } P \\ & | & (\nu\ n)P \\ & | & P_1 | P_2 \\ & | & !P \\ & | & 0 \end{array}$$

MT-LAB
A VKR CENTRE OF EXCELLENCE

CaPiTo with Annotations from LySa

From Annotated Concrete Level to LySa:

- Unfold CaPiTo tunnels $(r : e \blacktriangleright P)$
- into LySa's encryptions $(\{M\}_K)$ and decryptions (decrypt V as $\{; m\}_K$)

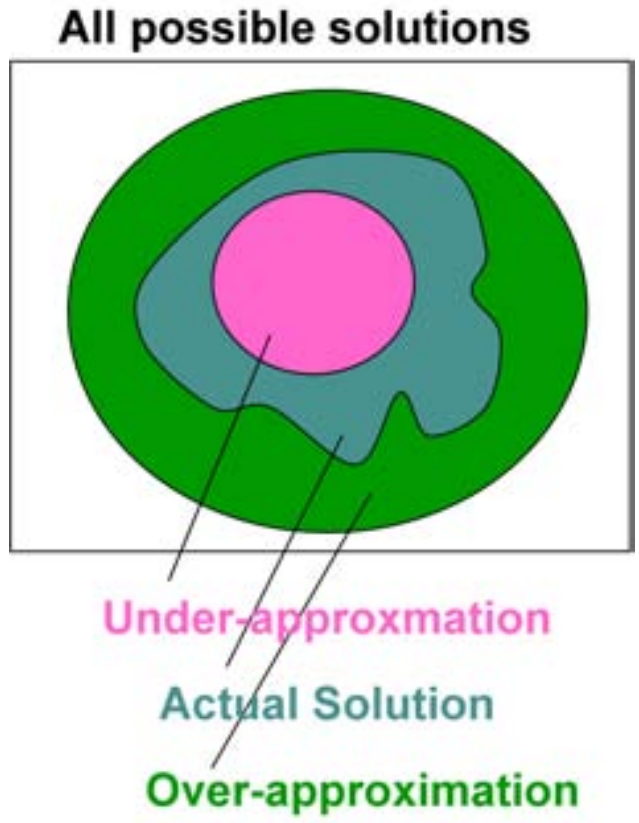
$$r : e_0 \blacktriangleright \langle r', \vec{e}\rangle^{l_0}[\mathit{dest}\ \mathcal{L}].P \rightsquigarrow \langle r', \{\vec{e}\}^{l_0}_{e_0}[\mathit{dest}\ \mathcal{L}]\rangle.r : e_0 \blacktriangleright P \quad \text{if } r = r'$$
$$r : e_0 \blacktriangleright \langle r', \vec{e}\rangle^{l_0}[\mathit{dest}\ \mathcal{L}].P \rightsquigarrow \langle r', \vec{e}\rangle^{l_0}[\mathit{dest}\ \mathcal{L}].r : e_0 \blacktriangleright P \quad \text{if } r \neq r'$$

$$r : e_0 \blacktriangleright (r', \vec{p})^{l_0}[\mathit{orig}\ \mathcal{L}].P \rightsquigarrow (r', \{\vec{p}\}^{l_0}_{e_0}[\mathit{orig}\ \mathcal{L}]).r : e_0 \blacktriangleright P$$
$$\text{if } r = r' \text{ and } \mathrm{fn}(e_0) \cap \mathrm{fn}(\vec{p}) = \emptyset$$
$$r : e_0 \blacktriangleright (r', \vec{p})^{l_0}[\mathit{orig}\ \mathcal{L}].P \rightsquigarrow (r', \vec{p})^{l_0}[\mathit{orig}\ \mathcal{L}].r : e_0 \blacktriangleright P \quad \text{if } r \neq r'$$

MT-LAB
A VKR CENTRE OF EXCELLENCE

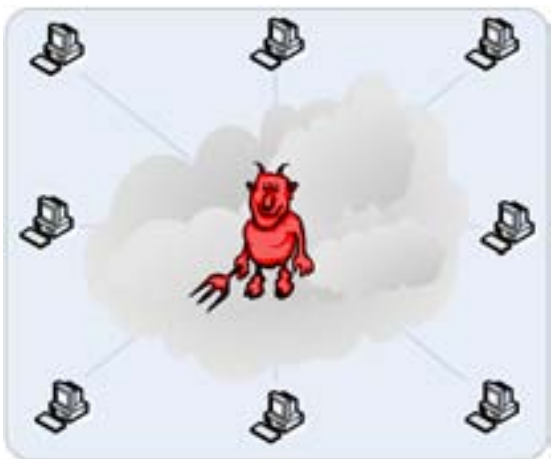
(7) Static Analysis of LySa

- Properties of Static Program Analysis
  - Semantically correct
    - Over-Approximation (e.g. this work)
    - Under-Approximation (e.g. AVISPA)
- Efficient algorithms



MT-LAB
A VKR CENTRE OF EXCELLENCE

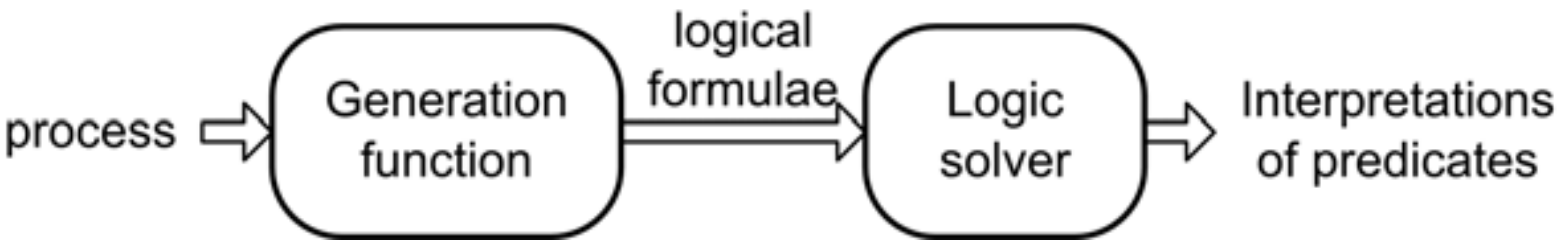
Static Analysis of LySa: The Dolev-Yao Attacker



- Challenge: the existence of the attacker
- Capabilities
  - receive and send messages on the network
  - encrypt and decrypt messages using known keys
  - create new keys, nonces, messages, etc
- Assumption: Prefect cryptography

MT-LAB
A VKR CENTRE OF EXCELLENCE

Static Analysis of LySa: Implementation Outline



- We adopt Succinct Solver as the Logic Solver.
- The results are computed in low polynomial time in the size of the system. In practice, less than two seconds.
- Alternatively, there are other candidates for the position of Logic Solver.
  - Datalog
  - H1, H3, ...

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Static Analysis

The overall scenario of the case study takes the form

$$\begin{aligned}&((\nu_{\pm}\ K_{VS})(\nu_{\pm}\ K_{Ser_E})(\nu_{\pm}\ K_{Ser_C})\\ &\qquad |_{i=1}^{n}\ Client_i \mid VS \mid SM \mid Ser_E \mid Ser_C)\\ &\mid\ Attacker\end{aligned}$$

The Scenario:

- i Clients may simultaneously request services from VS and Ser_E (or Ser_C). The index i is added to all variables, crypto-points and constants
- An active Dolev-Yao attacker. The attacker has no knowledge of the principals' asymmetric keys.

In our experiment, the analysis is carried out for $n = 2$, thus allows the analysis to see if the two instance of communications can interfere with each other.

MT-LAB
A VKR CENTRE OF EXCELLENCE

The Service Oriented Case Study: Analysis Results

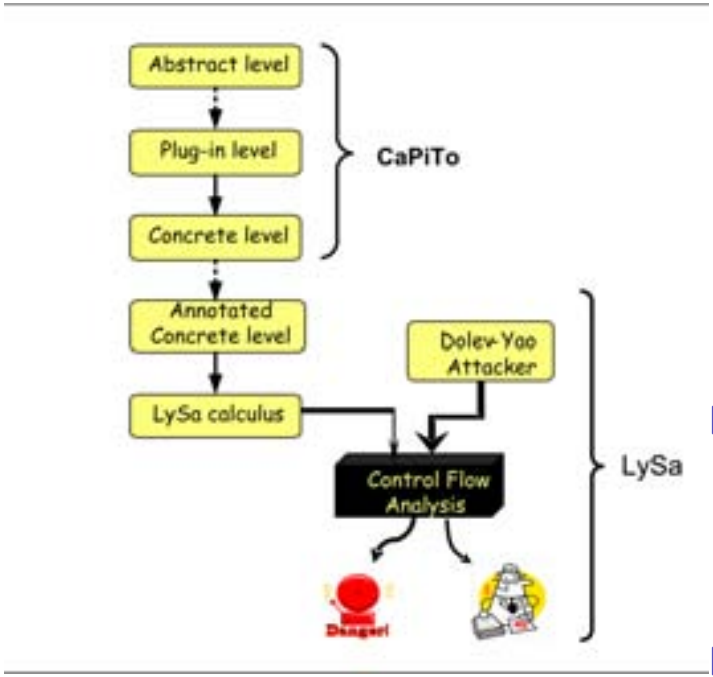
$$
\begin{aligned}
\mathit{Client} &\triangleq\ !\,(\nu\ \mathit{Bta})\overline{\mathit{credit\_req}}[\,]\cdot\Big(\langle \mathit{Bta}\rangle.(\mathit{Bta}, ?x_{res}).\uparrow\langle x_{res}\rangle.0\Big)\\
\mathit{Bank} &\triangleq\ !\,\mathit{credit\_req}[\,]\cdot\Big((?y_{bta}).\overline{\mathit{validate}}[\,]\cdot\langle y_{bta}\rangle.(?y_{res}).\uparrow\langle y_{bta}.y_{res}\rangle.0\Big)\\
\mathit{Ser}_E &\triangleq\ !\,\mathit{validate}[\,]\cdot\Big((?z_{bta}).\langle \mathit{isValid}(z_{bta})\rangle.0\Big)
\end{aligned}
$$

Analysing the case study gives an empty error (ψ) component

- no authentication annotations are violated

MT-LAB
A VKR CENTRE OF EXCELLENCE

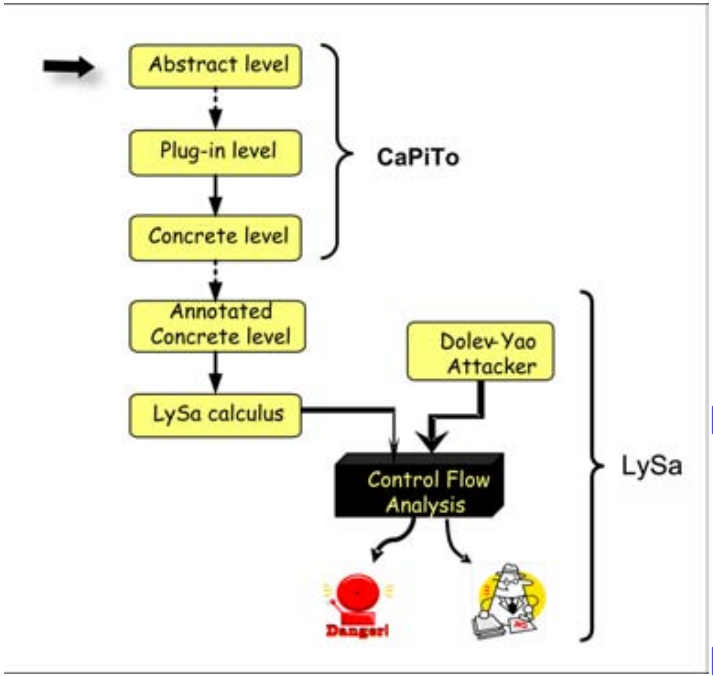
Summary



- The CaPiTo approach models Services at different levels of abstraction:
  - The Abstract Level focuses on the understanding of the application.
  - The Plug-in Level adds focus on existing protocols.
  - The Concrete Level focuses on the implementation of the application.
- The transformation from the Concrete Level to an implementation – for example in C – allows us to generate code stubs.
- The transformation from the Concrete Level to an analysis package – like the LySa tool – allows us to validate protocols.

MT-LAB
A VKR CENTRE OF EXCELLENCE

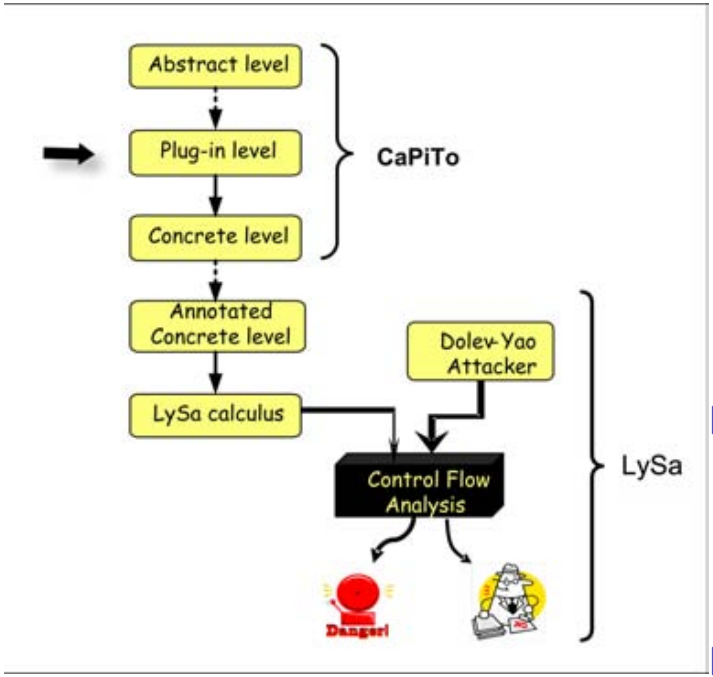
Summary



- The CaPiTo approach models Services at different levels of abstraction:
  - The Abstract Level focuses on the understanding of the application.
  - The Plug-in Level adds focus on existing protocols.
  - The Concrete Level focuses on the implementation of the application.
- The transformation from the Concrete Level to an implementation – for example in C – allows us to generate code stubs.
- The transformation from the Concrete Level to an analysis package – like the LySa tool – allows us to validate protocols.

MT-LAB
A VKR CENTRE OF EXCELLENCE

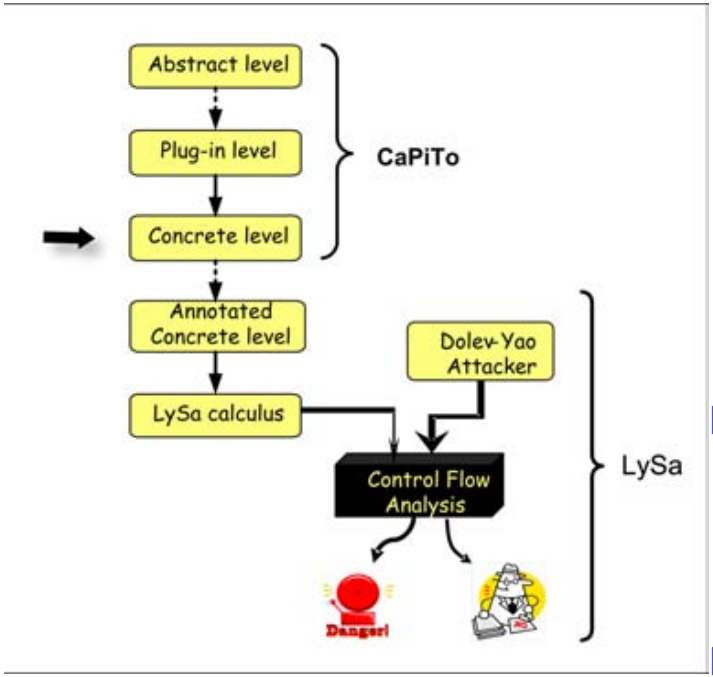
Summary



- The CaPiTo approach models Services at different levels of abstraction:
  - The Abstract Level focuses on the understanding of the application.
  - The Plug-in Level adds focus on existing protocols.
  - The Concrete Level focuses on the implementation of the application.
- The transformation from the Concrete Level to an implementation – for example in C – allows us to generate code stubs.
- The transformation from the Concrete Level to an analysis package – like the LySa tool – allows us to validate protocols.

MT-LAB
A VKR CENTRE OF EXCELLENCE

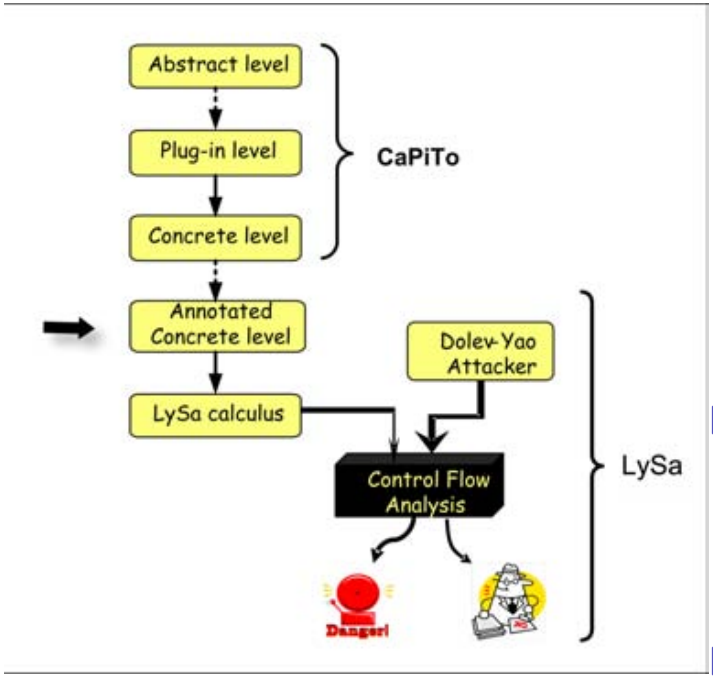
Summary



- The CaPiTo approach models Services at different levels of abstraction:
  - The Abstract Level focuses on the understanding of the application.
  - The Plug-in Level adds focus on existing protocols.
  - The Concrete Level focuses on the implementation of the application.
- The transformation from the Concrete Level to an implementation – for example in C – allows us to generate code stubs.
- The transformation from the Concrete Level to an analysis package – like the LySa tool – allows us to validate protocols.

MT-LAB
A VKR CENTRE OF EXCELLENCE

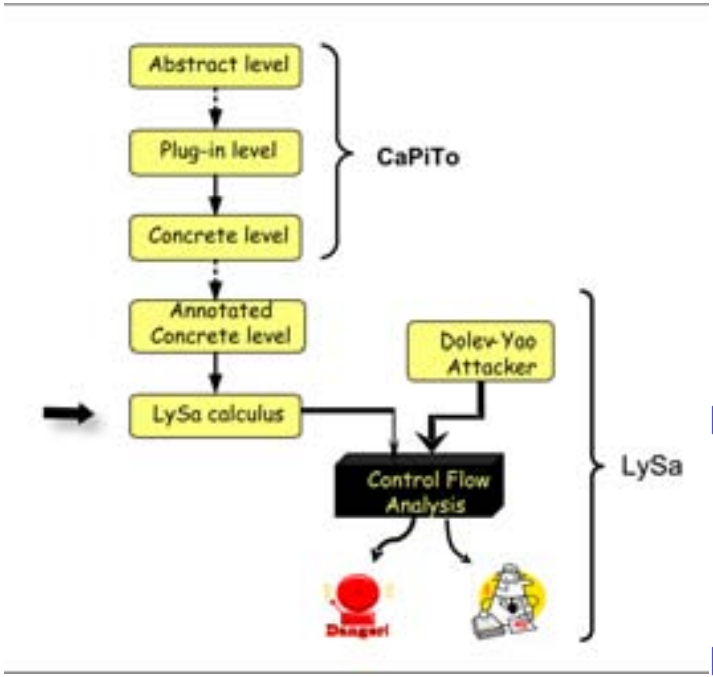
Summary



- The CaPiTo approach models Services at different levels of abstraction:
  - The Abstract Level focuses on the understanding of the application.
  - The Plug-in Level adds focus on existing protocols.
  - The Concrete Level focuses on the implementation of the application.
- The transformation from the Concrete Level to an implementation – for example in C – allows us to generate code stubs.
- The transformation from the Concrete Level to an analysis package – like the LySa tool – allows us to validate protocols.

MT-LAB
A VKR CENTRE OF EXCELLENCE

Summary



- The CaPiTo approach models Services at different levels of abstraction:
  - The Abstract Level focuses on the understanding of the application.
  - The Plug-in Level adds focus on existing protocols.
  - The Concrete Level focuses on the implementation of the application.
- The transformation from the Concrete Level to an implementation – for example in C – allows us to generate code stubs.
- The transformation from the Concrete Level to an analysis package – like the LySa tool – allows us to validate protocols.

MT-LAB
A VKR CENTRE OF EXCELLENCE