Towards a Verified Component Platform

Matthew Fernandez, Ihor Kuz, Gerwin Klein, June Andronick

November 2013
Trustworthy Systems
Trustworthy Systems
Trustworthy Systems

- Development processes
- Testing
- Configuration management
- Certification
- Formal verification
The L4.verified Project

- Specification: 5,000 LOC
- Proof: 200,000 LOC
- C Code: 10,000 LOC

25 person years
The L4.verified Project

25 person years

Specification

5,000 LOC

C Code

10,000 LOC

1,000,000 LOC
Component-Based Development
Component-Based Development
Component-Based Development

kernel + component platform + components
Component-Based Development

seL4

kernel + component platform + components
Component-Based Development
Component-Based Development

- seL4
- CAMkES
- User-provided components

- Kernel
- Component platform
End-to-end Guarantees
End-to-end Guarantees

architecture description → glue code → component code
End-to-end Guarantees
End-to-end Guarantees

- Architecture semantics
- Architecture description
- Glue code
- Userspace image
- Component code
End-to-end Guarantees
End-to-end Guarantees

system semantics

architecture semantics

architecture description

glue semantics

glue code

trusted component specification

component code

userspace image
End-to-end Guarantees

- Architecture Semantics
- Glue Semantics
- Trusted Component Specification
- Architecture Description
- Glue Code
- Component Code
- Userspace Image

System Semantics
End-to-end Guarantees

system semantics

architecture semantics

architecture description

 glue semantics

 glue code

userspace image

trusted component specification

component code
End-to-end Guarantees

system semantics

- architecture semantics
- glue semantics
- trusted component specification

userspace image
End-to-end Guarantees

- architecture semantics
- glue semantics
- trusted component specification

userspace image

system semantics
CAmkES

- Component platform for seL4
- Abstractions for procedures, events, shared memory
- Limited to static architectures

![Diagram of A and B connected with an f label between them.]

[architectural description]
CAmkES

- Component platform for seL4
- Abstractions for procedures, events, shared memory
- Limited to static architectures
CAmkES

- Component platform for seL4
- Abstractions for procedures, events, shared memory
- Limited to static architectures
CAmkES

- Component platform for seL4
- Abstractions for procedures, events, shared memory
- Limited to static architectures
CAmkES

- Component platform for seL4
- Abstractions for procedures, events, shared memory
- Limited to static architectures
CAmkES

- Component platform for seL4
- Abstractions for procedures, events, shared memory
- Limited to static architectures
Architecture Semantics

```
composition {
    component Client A;
    component Server B;

    connection RPC f(from A.i, to B.j);
}
```
Architecture Semantics

composition {
  component Client A;
  component Server B;

  connection RPC f(from A.i, to B.j);
}

definition
  system ::: composition
where
  system ≡ {
    components = [('A', Client), ('B', Server)],
    connections = [('f', (conn_type = RPC,
                           conn_from = ('A', 'i'),
                           conn_to = ('B', 'j')))]
  }
Component Process Language

UserStep \( x = 1 \)

\[
\text{SKIP} \quad \text{REQUEST} \quad \text{RESPONSE}
\]

\[
S_1 ;; S_2
\]

\[
\text{IF } c \text{ THEN } S_1 \text{ ELSE } S_2
\]

\[
\text{WHILE } c \text{ DO } S
\]
Component Behaviour
Component Behaviour
Component Behaviour

\[
\text{definition}
\begin{align*}
  B\_\text{untrusted} &::= \ldots \\
\text{where} &\quad B\_\text{untrusted} \equiv \text{WHILE True DO (} \\
\quad &\quad \begin{aligned}
\text{UserStep} \\
&\quad \begin{aligned}
&\Box \text{ArbitraryRequest} \ f \\
&\Box \text{ArbitraryResponse} \ f
\end{aligned}
\end{aligned}
\end{align*}
\]
Glue Code Behaviour

definition
    Receive_B_f :: ...
where
    Receive_B_f f embed f fproject ≡
        Response ... ;;
        f ;;
    Request ...

wait() unmarshal p
r = f(p)
marshal r
send()
Glue Code Behaviour

```
definition
  Receive_B_f :: ... |
where
  Receive_B_f f embed f fproject ≡
      Response ... ;;
    f ;;
  Request ... |
```

```
wait()
unmarshal p
r = f(p)
marshal r
send()
```
Glue Code Behaviour

definition
  Receive_B_f :: ... 
where
  Receive_B_f f embed f fproject ≡
    Response ... ;;
    f ;;
  Request ...

wait()
unmarshal p
r = f(p)
marshal r
send()
Glue Code Behaviour

definition
  Receive_B_f :: ...
where
  Receive_B_f fEmbed f fproject ≡
    Response ... ;;
    f ;;
    Request ...

wait()
unmarshal p
r = f(p)
marshal r
send()
Introduced Components

a → b
Introduced Components

\[ a \rightarrow b \]

\[ a \leftrightarrow ev \leftrightarrow b \]
Introduced Components

```
definition
  event :: ...  
where
  event c = WHILE True DO
    Response (\q. s'. case s' of Event s ->
      (case q_data q of
        Set -> ...  
        Poll -> ...,  
        _ -> {}}))
```
Introduced Components

```
definition
  event :: ...
where
  event c ≜ WHILE True DO
    Response (λq s'. case s' of Event s →
      (case q.data q of
        Set → ...
        Poll → ...
        _ → {}))
```

```
definition
  memory :: ...
where
  memory c ≜ WHILE True DO
    Response (λq s'. case s' of Memory s →
      (case q.data q of
        Read addr → ...
        Write addr val → ...
        _ → {}))
```
Towards a Verified Component Platform
Towards a Verified Component Platform
Towards a Verified Component Platform

- System semantics
  - Architecture semantics
  - Glue semantics
  - Trusted component specification

- Userspace image