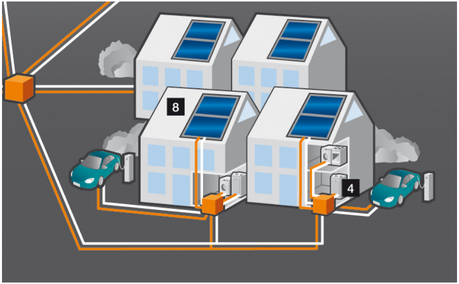
Architectural design for secure smart grids

Denis Bytschkow, Jean Quilbeuf, Georgeta Igna and Harald Ruess

fortiss

February 26, 2014

Smart Grid



Set of cooperating *prosumers* (PROducers – conSUMERS) coordinated through a central component

Challenges:

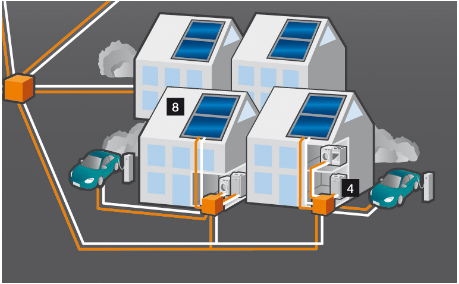
- stability
- safety
- security

We focus on security issues

Prosumer

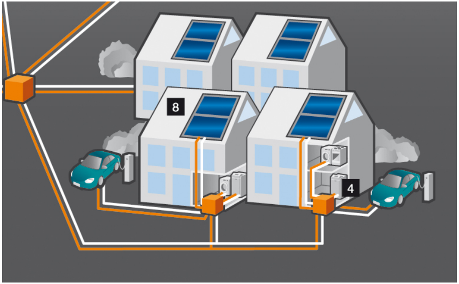
- Produces energy (solar panel)
- Stores energy (battery)
- Consumes energy (smart building)
- Plans consumption and production based on weather forecast and history
- Negotiates plans for the next day with the grid

Security Property



Privacy The power consumption of a household or a factory may reveal sensitive informations.

Security Property



Privacy The power consumption of a household or a factory may reveal sensitive informations.

SEC: *No prosumer knows the consumption plan of another prosumer.*

D-MILS Approach

D-MILS (FP7 project) approach relies on two levels:

1. Enforcing an information flow at the platform level
2. Checking higher-level security properties based on the information flow

D-MILS Approach

D-MILS (FP7 project) approach relies on two levels:

1. Enforcing an information flow at the platform level
2. Checking higher-level security properties based on the information flow

No prosumer knows the consumption plan of another prosumer.

becomes:

1. No prosumer can directly access the consumption plan of another prosumer.
2. Based on the information flow, no prosumer can deduce the consumption plan of another prosumer.

D-MILS Approach

Information flow is obtained by:

- separation of software components running on the same machine
  - time and space partitioning provided by a separation kernel (Rushby, 1981)
  - several separation kernel exists (LynxSecure is used in D-MILS)
  - MILS approach

D-MILS Approach

Information flow is obtained by:

- separation of software components running on the same machine
  - time and space partitioning provided by a separation kernel (Rushby, 1981)
  - several separation kernel exists (LynxSecure is used in D-MILS)
  - MILS approach
- separation of communication channels between components
  - within the same machine: handled by the separation kernel
  - between different machines: different techniques depending on the network: time-triggered ethernet, cryptography ...
  - extension of MILS approach to distributed systems (D-MILS)

D-MILS Approach

Information flow is obtained by:

- separation of software components running on the same machine
  - time and space partitioning provided by a separation kernel (Rushby, 1981)
  - several separation kernel exists (LynxSecure is used in D-MILS)
  - MILS approach
- separation of communication channels between components
  - within the same machine: handled by the separation kernel
  - between different machines: different techniques depending on the network: time-triggered ethernet, cryptography ...
  - extension of MILS approach to distributed systems (D-MILS)

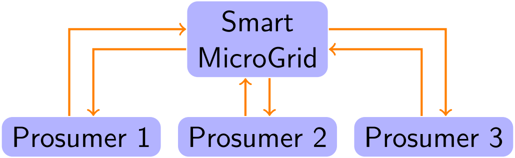
Verification of high-level properties builds on the information flow.

Outline

1. Introduction
2. Model and Platform
3. Ensuring Information Flow
4. Security as hyperproperties
5. Conclusion and Future work

Outline

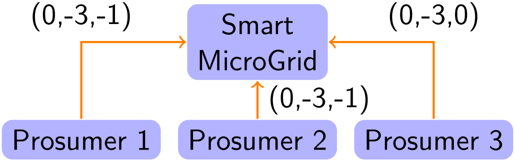
Logical Architecture



Negotiation between the smart micro grid and the prosumers:

- prosumers send their plan (consumption and production)
- if admissible, the SMG validates the plans
  (SMG sends back an acknowledgement)
- else, prosumers have to modify their plans
  (SMG indicates that consumption or production has to be reduced)

Model Execution



	1		
Pr1	(0,-3,-1)		...
Pr2	(0,-3,-1)		...
Pr3	(0,-3,0)		...
SMG			...

Each prosumer indicates its (production,consumption,battery usage).

Model Execution

Smart MicroGrid

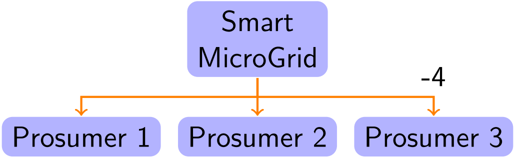
Admissible: $[-7, 4]$
Total: -11
Overconsumption: -4

Prosumer 1 Prosumer 2 Prosumer 3

	1		
Pr1	(0,-3,-1)		...
Pr2	(0,-3,-1)		...
Pr3	(0,-3,0)		...
SMG			...

SMG checks whether total production or consumption is admissible.

Model Execution



	1	2	
Pr1	(0,-3,-1)		...
Pr2	(0,-3,-1)		...
Pr3	(0,-3,0)		...
SMG		-4	...

SMG returns over consumption/production amount. (0 admissible)

Model Execution

Admissible: $[-7, 4]$
Total: -6
OK

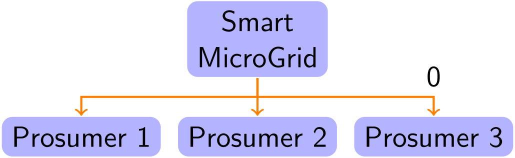
Smart MicroGrid

Prosumer 1 | Prosumer 2 | Prosumer 3

	1	2	3	
Pr1	(0,-3,-1)		(0,-3,1)	...
Pr2	(0,-3,-1)		(0,-3,1)	...
Pr3	(0,-3,0)		(0,-2,0)	...
SMG		-4		...

Negotiation continues until plans are admissible

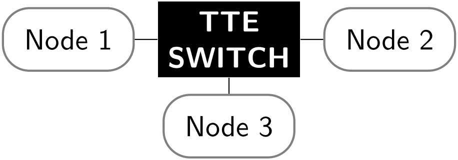
Model Execution



	1	2	3	4	
Pr1	(0,-3,-1)		(0,-3,1)		...
Pr2	(0,-3,-1)		(0,-3,1)		...
Pr3	(0,-3,0)		(0,-2,0)		...
SMG		-4		0	...

Platform

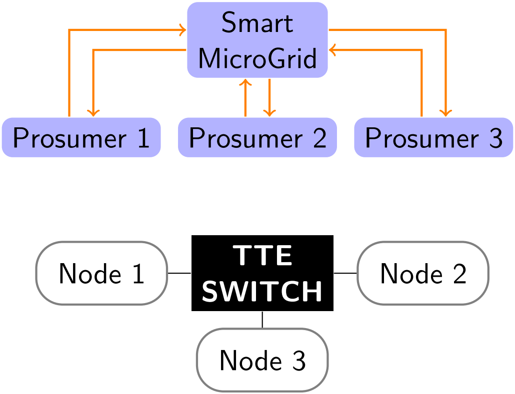
Set of D-MILS nodes connected through a time-triggered ethernet network.



- Each node hosts a separation kernel (separation of components)
- The network is statically scheduled, allowing time partitionning (separation of communication channels)

Outline

Ensuring information flow



Logical architecture = information flow to ensure

Providing a mapping:

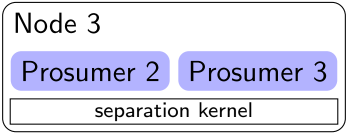
Smart MicroGrid $\mapsto$ Node 2
Prosumer 1 $\mapsto$ Node 1
Prosumer 2 $\mapsto$ Node 3
Prosumer 3 $\mapsto$ Node 3

Configure the platform to enforce information flow

Ensuring information flow: Inside a node

Component deployed on the same node are run in distinct partitions of the separation kernel. (MILS approach)

In our example, Prosumer 2 and 3 are deployed on node 3:

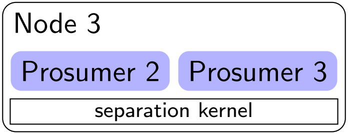


The separation kernel handles communication between component deployed on the same node. (none here)

Ensuring information flow: Inside a node

Component deployed on the same node are run in distinct partitions of the separation kernel. (MILS approach)

In our example, Prosumer 2 and 3 are deployed on node 3:

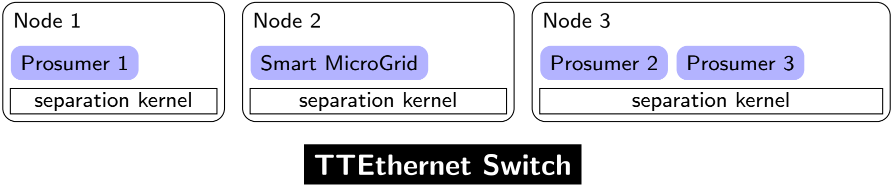


The separation kernel handles communication between component deployed on the same node. (none here)

The configuration compiler provides a configuration file for the kernel.

Separation of communications channels

Each channel is allocated a fixed time partition.

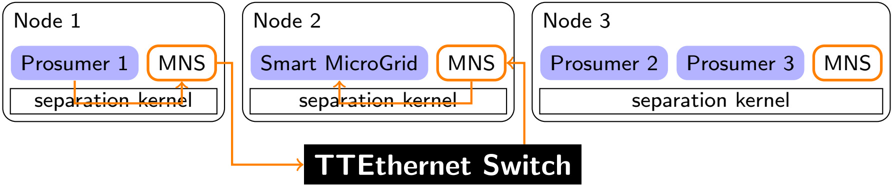


Time partitionning ensured by:

- A MILS Network Server component in each node
- Configuration of the switch

Separation of communications channels

Each channel is allocated a fixed time partition.

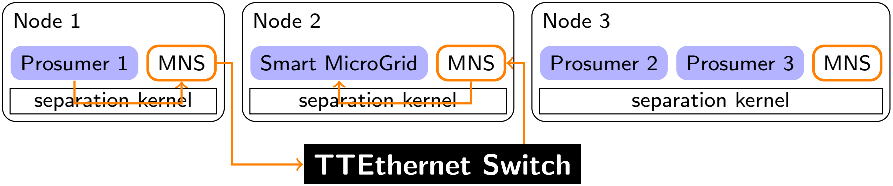


Time partitionning ensured by:

- A MILS Network Server component in each node
- Configuration of the switch

Separation of communications channels

Each channel is allocated a fixed time partition.

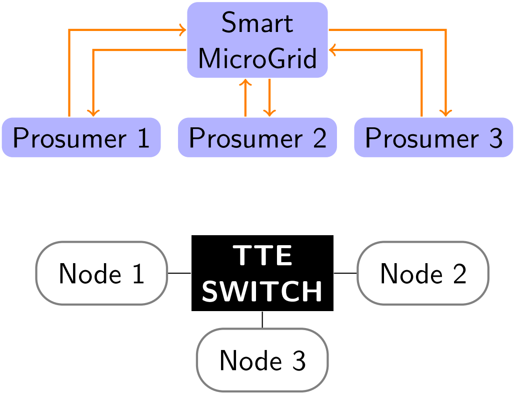


Time partitionning ensured by:

- A MILS Network Server component in each node
- Configuration of the switch

The configuration compiler provides a configuration file for each switch.

Configuration compiler



Based on

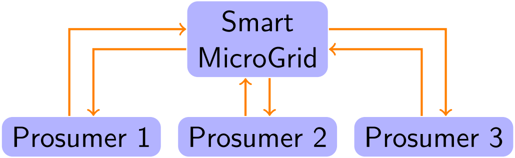
- the logical architecture,
- the platform model,
- the mapping,

the configuration compiler generates configuration files:

- for the separation kernels
- for the network switches

Outline

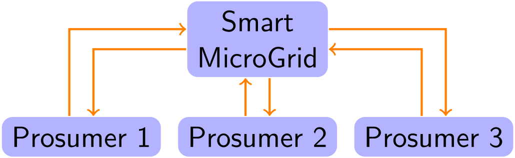
Knowledge



Information flow ensures that:
A component can only observe its inputs and outputs

	1	2	3	4
Pr1	(0,-3,-1)		(0,-3,1)	
Pr2	(0,-3,-1)		(0,-3,1)	
Pr3	(0,-3,0)		(0,-2,0)	
SMG		-4		0

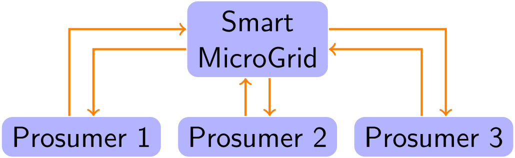
Knowledge



Information flow ensures that:
A component can only observe its inputs and outputs

		1	2	3	4
	Pr1	(0,-3,-1)		(0,-3,1)	
	Pr2	(0,-3,-1)		(0,-3,1)	
Observed by Prosumer 3	Pr3	(0,-3,0)		(0,-2,0)	
	SMG		-4		0

Knowledge



Information flow ensures that:
A component can only observe its inputs and outputs

		1	2	3	4
	Pr1	(0,-3,-1)		(0,-3,1)	
	Pr2	(0,-3,-1)		(0,-3,1)	
Observed by Prosumer 3	Pr3	(0,-3,0)		(0,-2,0)	
	SMG		-4		0

Each observation can be completed in a global trace.

A components knows that a given fact is true
if this fact holds in all traces completing its local observation.

Application to our example

We assume that for each prosumer:

- Production is between 0 and 2
- Consumption is between -3 and 0
- Battery usage is between -1 (loading battery) and 1 (using battery)

Furthermore, admissible values for the global consumption or production are between -7 and 4.

Application to our example

We assume that for each prosumer:

- Production is between 0 and 2
- Consumption is between -3 and 0
- Battery usage is between -1 (loading battery) and 1 (using battery)

Furthermore, admissible values for the global consumption or production are between -7 and 4.

Observation from Prosumer 3:

	1	2
Pr3	(0,-3,0)	
SMG		-4

Prosumer 3 can infer:

- global consumption is -11

- Prosumers 1 and 2 consume -8

Only one possible trace:

both consume -3 and charge battery

Application to our example

We assume that for each prosumer:

- Production is between 0 and 2
- Consumption is between -3 and 0
- Battery usage is between -1 (loading battery) and 1 (using battery)

Furthermore, admissible values for the global consumption or production are between -7 and 4.

Observation from Prosumer 3:

	1	2
Pr3	(0,-3,0)	
SMG		-4

Prosumer 3 can infer:
- global consumption is -11
- Prosumers 1 and 2 consume -8

Only one possible trace:
both consume -3 and charge battery

Prosumer 3 knows consumption plans of Prosumers 1 and 2.

Application to our example

Returning the exact amount of energy overproduced/overconsumed is not secure !

Need to change implementation of the smart micro grid.
Possible solution: return values in
$\{$overconsumption, overproduction, admissible$\}$

Application to our example

Returning the exact amount of energy overproduced/overconsumed is not secure !

Need to change implementation of the smart micro grid.
Possible solution: return values in
{overconsumption, overproduction, admissible}

"No prosumer knows the consumption plan of another prosumer"
is an hyperproperty (Clarkson and Schneider, 2010)

Checked by self-composition of the system (Barthe et al. 2011)
Ongoing work...

Outline

References

D-MILS project: `www.d-mils.org`

(Rushby 1981) *The Design and Verification of Secure Systems*

(Clarkson and Schneider 2010) *Hyperproperties*

(Barthe et al. 2011) *Secure information flow by self-composition*

(van der Meyden, 2007) *What, indeed, is intransitive noninterference?*

(Balliu, 2013) *A logic for information flow analysis of distributed programs*

Conclusion and future work

We applied the D-MILS approach to the negociation phase of a smart grid model

1. Information to secure cannot be directly acessed
2. Hyperproperty formalizes the fact that the secure information cannot be deduced from local observation

Conclusion and future work

We applied the D-MILS approach to the negociation phase of a smart grid model

1. Information to secure cannot be directly acessed
2. Hyperproperty formalizes the fact that the secure information cannot be deduced from local observation

Ongoing and Future Work

- Formalization and verification of hyperproperties
- More precise model of the prosumer (include smart building and forecast components)
- Extension of D-MILS to handle different types of network
  - Wifi for components of smart buildings
  - regular ethernet

Thank You

Any questions ?