TECHNISCHE UNIVERSITÄT WIEN Vienna | Austria

Christian Doppler Forschungsgesellschaft

Security and Quality Improvement in the Production System Lifecycle

Christian Doppler Forschungsgesellschaft

Securing Cyber-Physical Systems through Digital Twins

Matthias A. Eckhart

@MatthiasEckhart

The Digital Twin

> *A digital twin is an integrated […] **simulation** of a […] system that uses the best available **physical models, sensor updates**, […] etc., to **mirror** the life of its […] flying twin.*
>
> *Shafto et al. [7]*

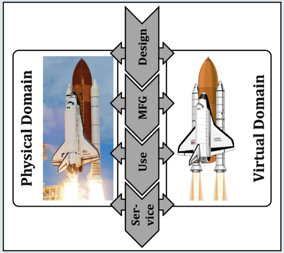


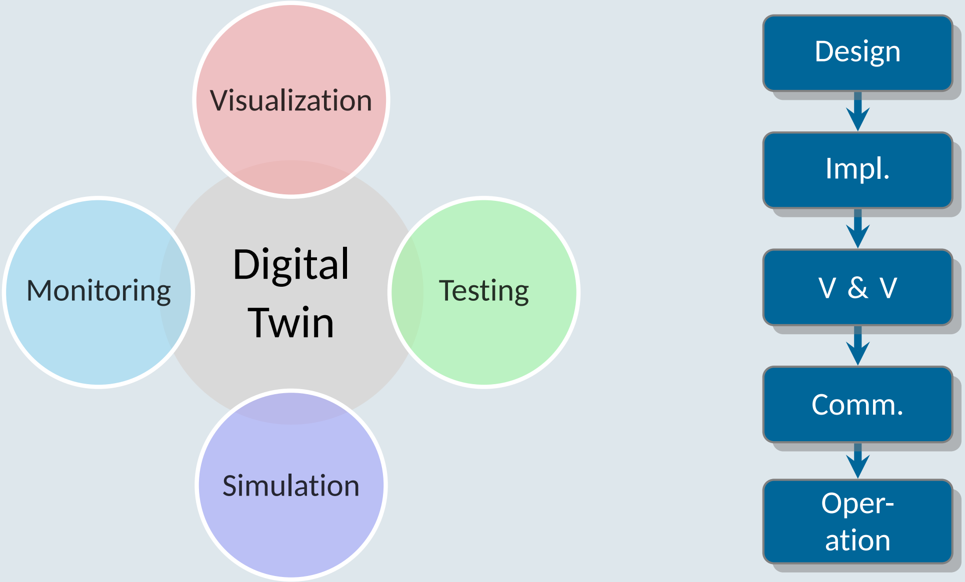
Figure: The vision according to [6].

(a) Nuclear power plant © AlMare, CC BY-SA 3.0

(b) Industrial Robots © Mixabest, CC BY-SA 3.0

(c) Tesla Model S © raneko, CC BY 2.0

Use Cases of the Digital Twin Concept

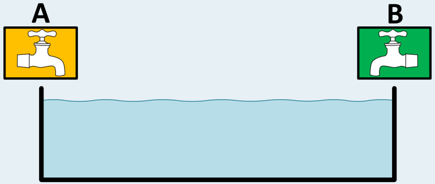


Security-specific Use Cases of the Concept

Intrusion Detection

- Knowledge-based
- Behavior-specification-based
- Process knowledge

Example: Sequence Attacks (e.g., [1])



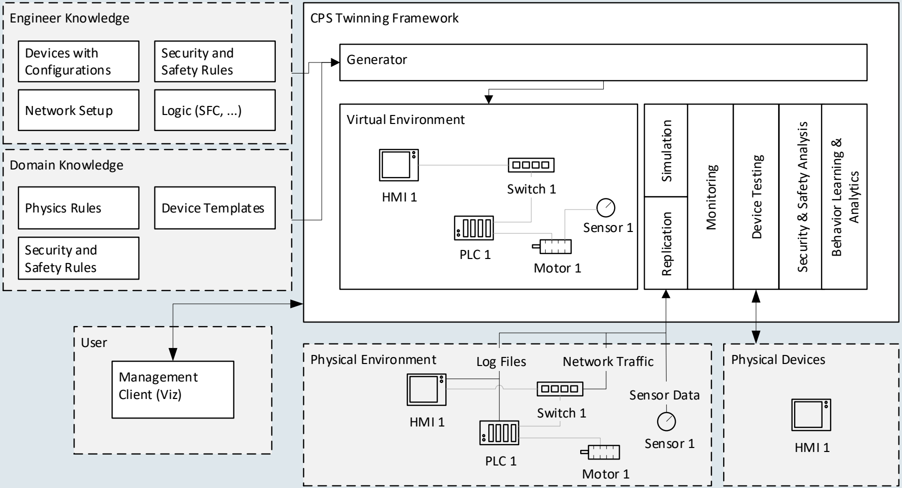
Detecting Misconfigurations

- Manipulation by attacker
- Detect unknown devices

Penetration Testing

- No interference with live system
- No test environment required

Architecture of CPS Twinning

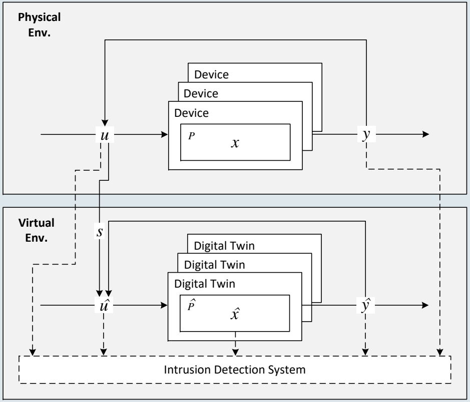


State Replication

A Passive State Replication Approach

A FSM, $P := (X, x_0, U, Y, \delta, \lambda)$

- X is the finite set of states
- $x_0 \in X$ is the initial state
- U is the finite set of inputs
- Y is the finite set of outputs
- δ is the transition function
- λ is the output function



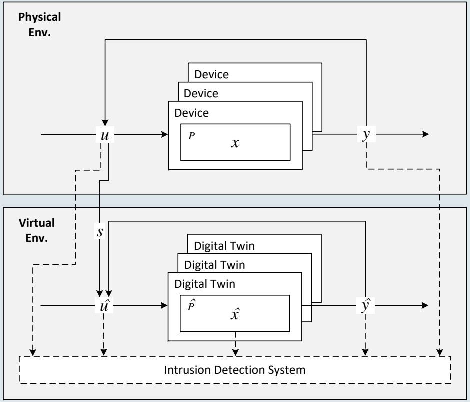
A Passive State Replication Approach

A FSM, $P := (X, x_0, U, Y, \delta, \lambda)$

- X is the finite set of states
- $x_0 \in X$ is the initial state
- U is the finite set of inputs
- Y is the finite set of outputs
- δ is the transition function
- λ is the output function

We expect that $P = \hat{P}$

Thus, $\delta(x, u) = \hat{\delta}(\hat{x}, \hat{u}) \Leftrightarrow x' = \hat{x}'$, provided that $(x = \hat{x}) \wedge (u = \hat{u})$.

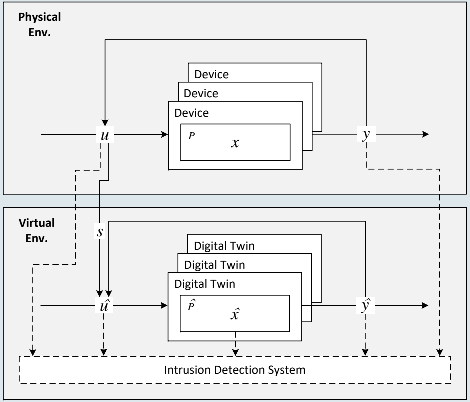


A Passive State Replication Approach

S, denotes the set of stimuli

$$S := \{\, z \in \hat{U} \mid z \in U \land z \notin Y^* \,\}$$

Each digital twin should produce $\hat{y} \in \hat{Y}$ by itself.



A Passive State Replication Approach

S, denotes the set of stimuli

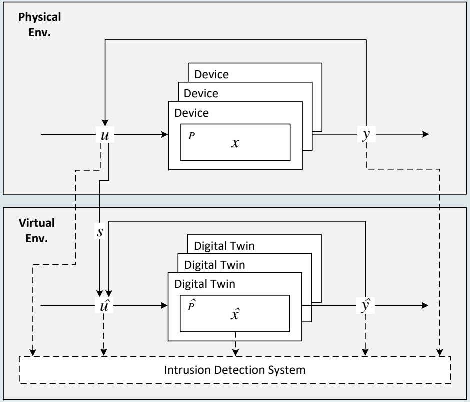
$$S := \{\, z \in \hat{U} \mid z \in U \wedge z \notin Y^* \,\}$$

Each digital twin should produce $\hat{y} \in \hat{Y}$ by itself.

Use specification of CPS to identify stimuli

Let $f\colon\ U^* \cup Y^* \rightharpoonup S^*$ be a partial function, then I is defined as follows:

$$I := \{\, j \in U^* \cup Y^* \mid f(j) \in S^* \,\}.$$



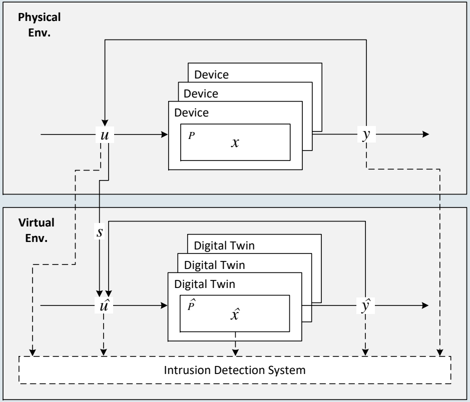
A Passive State Replication Approach

Replicate stimuli

Next, $j \in U^* \cup Y^*$ will be observed and checked whether $j \in I$.

Since $j \in I \Leftrightarrow f(j)\downarrow$, $s \in S^*$, the value of f of j, is fed to the respective digital twin.

Hence, $\hat{\delta}(\hat{x}, s) = \hat{x}'$.



Example

- Conveyor belt
- HMI & PLC digital twins exist
- Communication via Modbus TCP/IP
- Definition of *f*
- AutomationML [2]

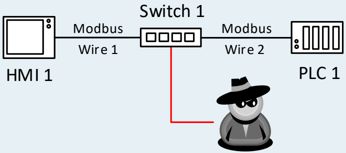
```
<InternalElement Name="LogicalNetwork" ID="c51...">
  <InternalElement Name="ModbusRequests" ID="ce1...">
    <InternalElement
      ↪     Name="StartConveyorBeltModbusReadRequest"
      ↪     ID="0e5...">
      <Attribute Name="functionCode"
        ↪     AttributeDataType="xs:integer">
        <Value>3</Value>
      </Attribute>
      ...
      <InternalLink Name="HMI1 StartConveyorBelt -
        ↪     PLC1 Modbus 400001" RefPartner-
        ↪     SideA="{068...}:StartConveyorBelt"
        ↪     RefPartnerSideB="{29b...}:1" />
      <RoleRequirements RefBaseRoleClass-
        ↪     Path="/ModbusReadHoldingRegisters"
        ↪     />
    </InternalElement>
    ...
```

Intrusion Detection

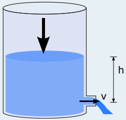
Knowledge-based IDS

Implicit

Network Layout



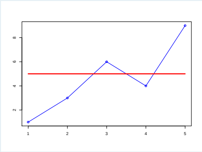
Laws of Physics



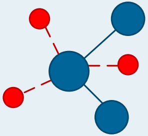
Torricelli's Law © LimoWreck, CC BY-SA 3.0

Explicit

Thresholds for Variables



Relationship Between States



Behavior-specification-based IDS

Assumptions

- Specification of CPS defines the correct behavior
- Digital twin follows state of its physical counterpart

Inner workings

- Comparison between $p \in U^* \cup Y^*$ and $v \in \hat{U}^* \cup \hat{Y}^*$
- Predefined features (e.g., Modbus FC)

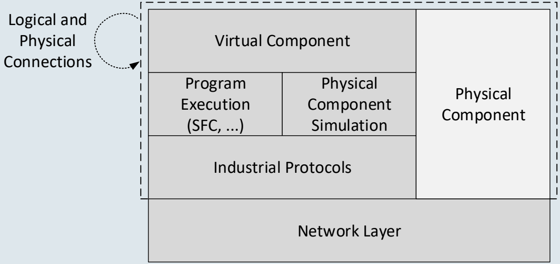
Benefits & drawbacks

- Automatic in-depth checks without causing any risks of interference
- Risk of replicating malicious stimuli

Proof of Concept

Prototype

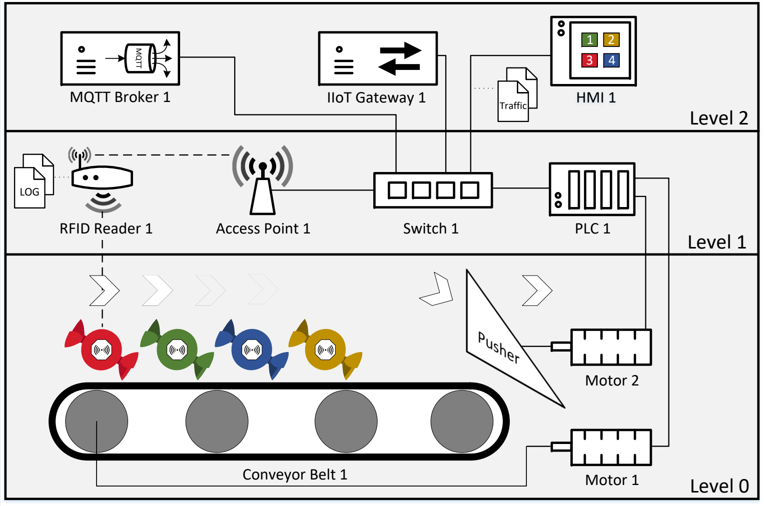
- Based on Mininet [5]
- Integration of MatIEC transcompiler
- GitHub Repos:
    - CPS Twinning
    - CPS State Replication



CPS Twinning CLI

```
mininet> twinning /home/user/ConveyorSystem.aml
mininet> nodes
available nodes are:
HMI1 PLC1 Switch1 c0
mininet> links
Switch1-eth1<->HMI1-eth0 (OK OK)
Switch1-eth2<->PLC1-eth0 (OK OK)
mininet> show_tags PLC1
Name        |Class       |Type
---------------------------------------
ENABLE      |var         |bool
PTO         |var         |bool
Q10         |out         |bool
Q00         |out         |bool
START       |mem         |bool
STOP        |mem         |bool
VELOCITY    |mem         |int
...
mininet> get_tag PLC1 START
False
mininet> set_tag PLC1 START True
mininet> get_tag PLC1 START
True
```

Scenario



MQTT
MQTT Broker 1
IIoT Gateway 1
1
2
3
4
HMI 1
Traffic
Level 2
LOG
RFID Reader 1
Access Point 1
Switch 1
PLC 1
Level 1
Pusher
Motor 2
Conveyor Belt 1
Motor 1
Level 0

Evaluation: Detection of Attacks (1)

```
<ExternalInterface Name="Velocity"
↪     RefBase="/VariableInterface">
  <Attribute Name="refURI">
    <Value>file:///SFC.xml#Velocity</Value>
    <Constraint Name="Safety Rule Motor">
      <OrdinalScaledType>
        <RequiredMaxValue>
            60
        </RequiredMaxValue>
      </OrdinalScaledType>
            ...
```

```
<InternalElement Name="VelocityConstraint"
  ↪     ID="e0b...">
  <Attribute Name="operator"
    ↪     AttributeDataType="xs:string">
    <Value>equals</Value>
  </Attribute>
  <InternalLink Name="VelocityConstraint
    ↪     PLC1 - HMI1"
    ↪     RefPartnerSideA="{133...}:Velocity"
    ↪     RefPartnerSideB="{068...}:Velocity"
    ↪     />
  ...
</InternalElement>
```

```
INFO:root:'Velocity' value changed 0 -> 20 in device 'HMI1'.
INFO:root:'VELOCITY' value changed 0 -> 100 in device 'PLC1'.
WARNING:root:ALERT! 'PLC1' tag [Velocity=100] exceeds max value of 60.
WARNING:root:ALERT! 'HMI1' tag [Velocity=20] does not equal 'PLC1' tag [Velocity=100].
```

IDS Output

Evaluation: Detection of Attacks (2)

```
14:04:55.178 - Count [pCandy=1,vCandy=1].
+-------+
| candy|
+-------+
|Cherry|
+-------+
14:06:06.392 - Count [pMQTT=8,vMQTT=1].
+---------+---------+------------+------------+---+---+---+---+---------+----------+--------+
|  eth.src|  eth.dst|      ip.src|      ip.dst|...|...|...|...|mqtt.len|mqtt.topic|mqtt.msg|
+---------+- -------+------------+------------+---+---+---+---+---------+----------+--------+
|08:00:...|f8:1e:...|192.168.0.61|192.168.0.32|  3|  0|  0|  0|       11|     candy|    Mint|
...
|08:00:...|f8:1e:...|192.168.0.61|192.168.0.32|  3|  0|  0|  0|       11|     candy|    Mint|
+---------+---------+------------+------------+---+---+---+---+---------+----------+--------+
```

```
15:07:21.065 - Count [pCandy=1,vCandy=1].
+------+
|candy|
+------+
| Mint|
+------+
```

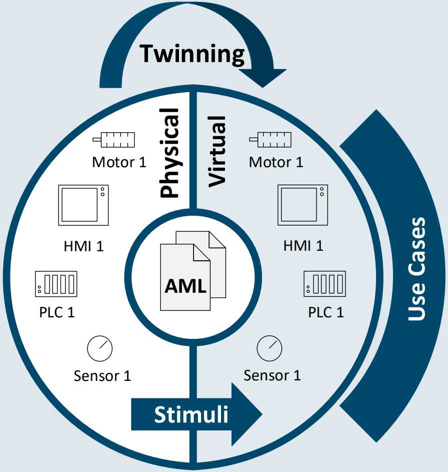
Conclusion

Contribution

- Generation of digital twins from specification
- State replication

Challenges

- Specification often non-existent or incomplete
- Performance issues
- High overhead, even though automatic generation is feasible



References I

[1] Marco Caselli, Emmanuele Zambon, and Frank Kargl.
Sequence-aware intrusion detection in industrial control systems.
In *Proceedings of the 1st ACM Workshop on Cyber-Physical System Security*, CPSS '15, pages 13–24, New York, NY, USA, 2015. ACM.

[2] R. Drath, A. Luder, J. Peschke, and L. Hundt.
Automationml - the glue for seamless automation engineering.
In *2008 IEEE International Conference on Emerging Technologies and Factory Automation*, pages 616–623, Sept 2008.

[3] Matthias Eckhart and Andreas Ekelhart.
A specification-based state replication approach for digital twins.
In *Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy*, CPS-SPC '18, pages 36–47, New York, NY, USA, 2018. ACM.

References II

[4] Matthias Eckhart and Andreas Ekelhart.
Towards security-aware virtual environments for digital twins.
In *Proceedings of the 4th ACM Workshop on Cyber-Physical System Security*, CPSS '18, pages 61–72, New York, NY, USA, 2018. ACM.

[5] Bob Lantz, Brandon Heller, and Nick McKeown.
A network in a laptop: Rapid prototyping for software-defined networks.
In *Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks*, Hotnets-IX, pages 19:1–19:6, New York, NY, USA, 2010. ACM.

[6] Benjamin Schleich, Nabil Anwer, Luc Mathieu, and Sandro Wartzack.
Shaping the digital twin for design and production engineering.
CIRP Annals, 66(1):141 – 144, 2017.

[7] Mike Shafto, Mike Conroy, Rich Doyle, Ed Glaessgen, Chris Kemp, Jacqueline LeMoigne, and Lui Wang.
Draft modeling, simulation, information technology & processing roadmap.
Technology Area, 11, 2010.

TECHNISCHE UNIVERSITÄT WIEN
Vienna | Austria

Christian Doppler Forschungsgesellschaft

Matthias A. Eckhart

Christian Doppler Labor für die Verbesserung von Sicherheit und Qualität in Produktionssystemen (CDL SQI)

Technische Universität Wien
Favoritenstraße 9–11, 1040 Wien
+43 664 4483435
Matthias.Eckhart@tuwien.ac.at