DUCE: Distributed Usage Control Enforcement for Private Data Sharing in Internet of Things

Na Shi, Bo Tang, Ravi Sandhu, and Qi Li

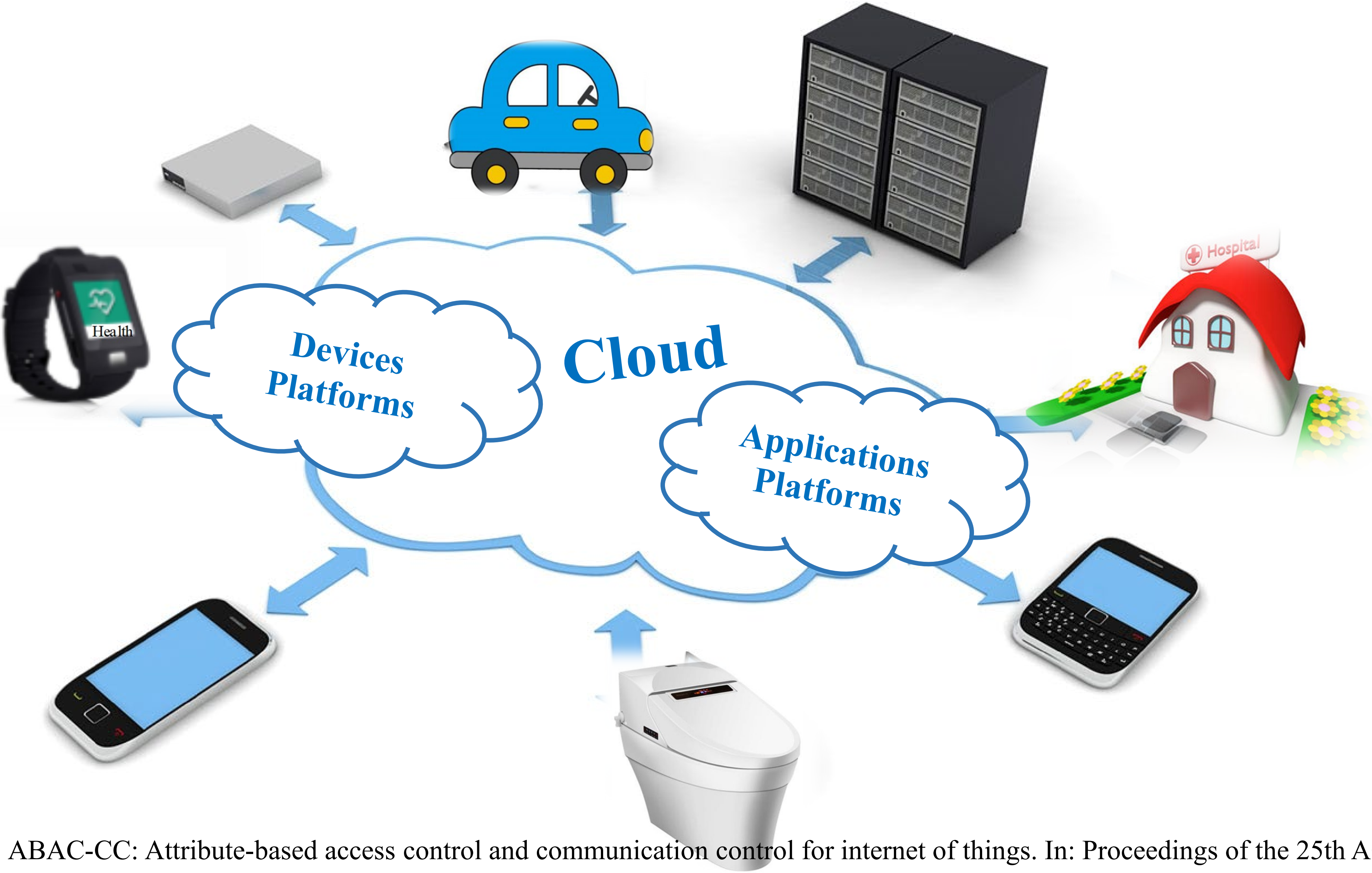
CHANGHONG长虹

I·C·S
The Institute for Cyber Security

清華大學 TSINGHUA UNIVERSITY -1911-

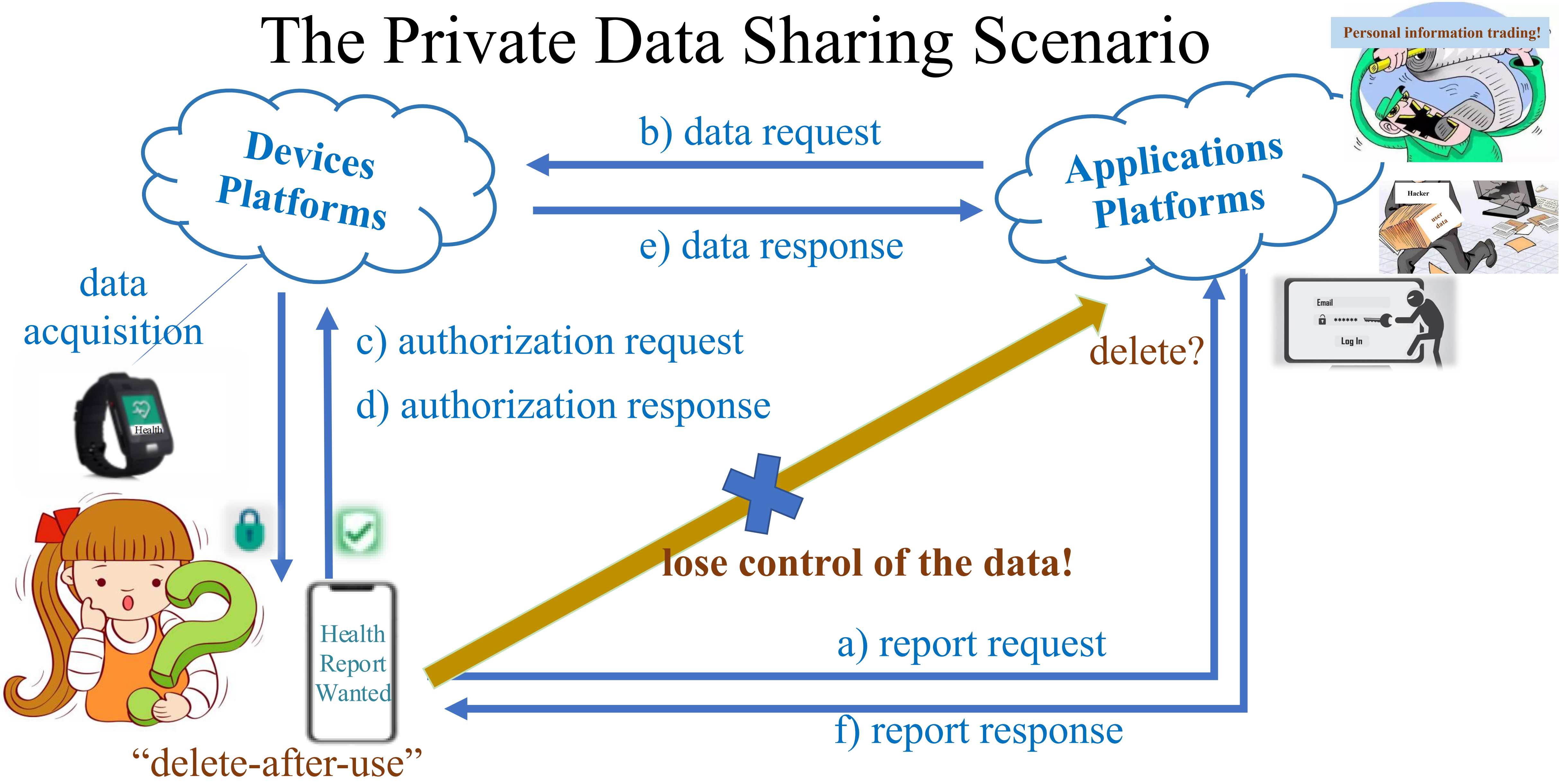
July 20, 2021

The Cloud-Enabled IoT[5, 6]



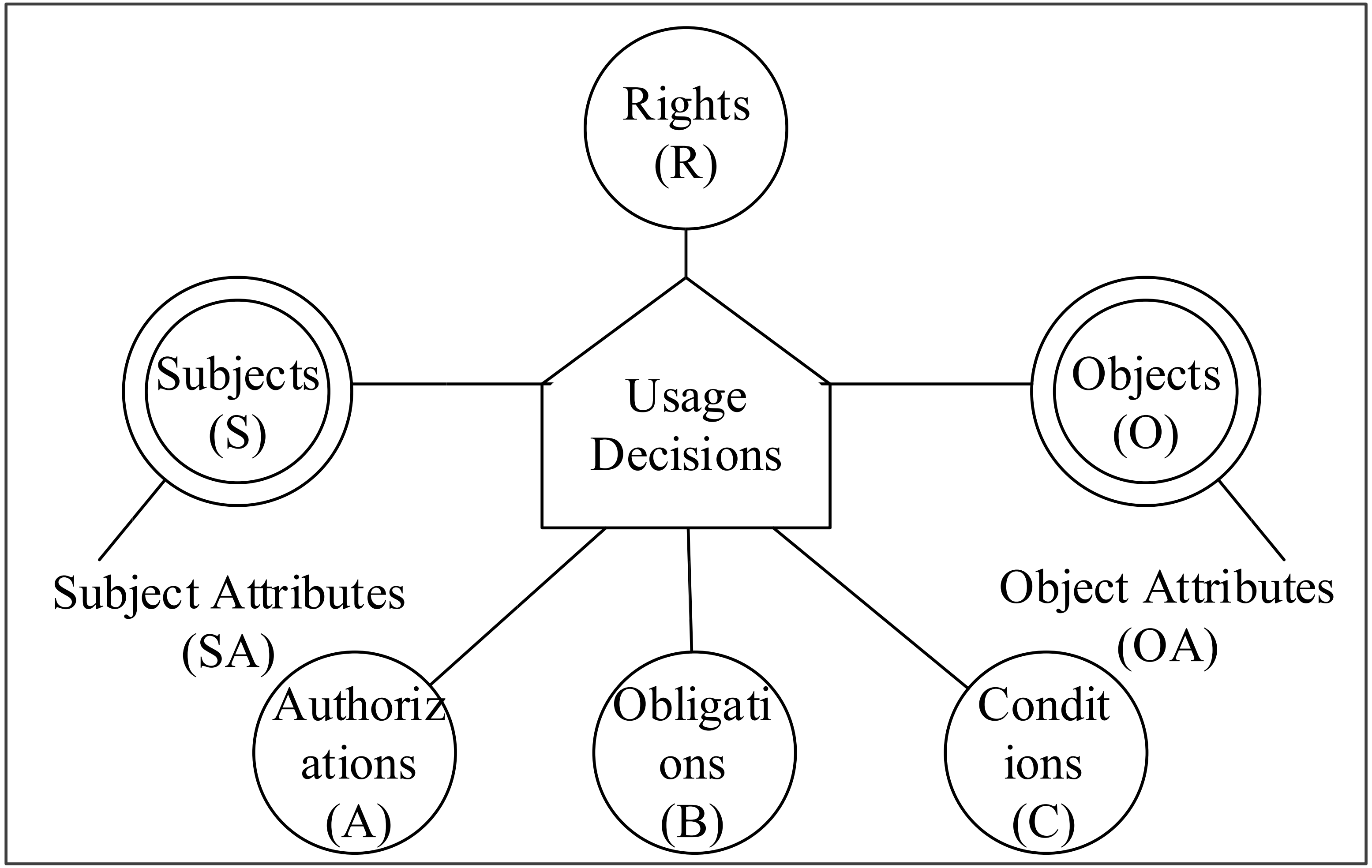
[5] Bhatt, S., Sandhu, R.: ABAC-CC: Attribute-based access control and communication control for internet of things. In: Proceedings of the 25th ACM Symposium on Access Control Models and Technologies, pp. 203–212 (2020).
[6] Chen, R., et al.: Trust-based service management for mobile cloud IoT systems. IEEE Trans. Netw. Serv. Manag. 16(1), 246–263 (2018).

The Private Data Sharing Scenario



Motivations: A distributed usage control enforcement model with privacy preserving for private data sharing.

What is UCON$_{ABC}$ [24]?



continuity of control on usage of digital resource

[24] Park, J., Sandhu, R.: The uconabc usage control model. ACM Trans. Inf. Syst. Secur. (TISSEC) 7(1), 128–174 (2004).

Things Need to Be Done in the Model

- **Privacy Preserving**

requires that the shared user data and the keys in authorization used to decrypt this data should be protected.

- **Integrity Protection**

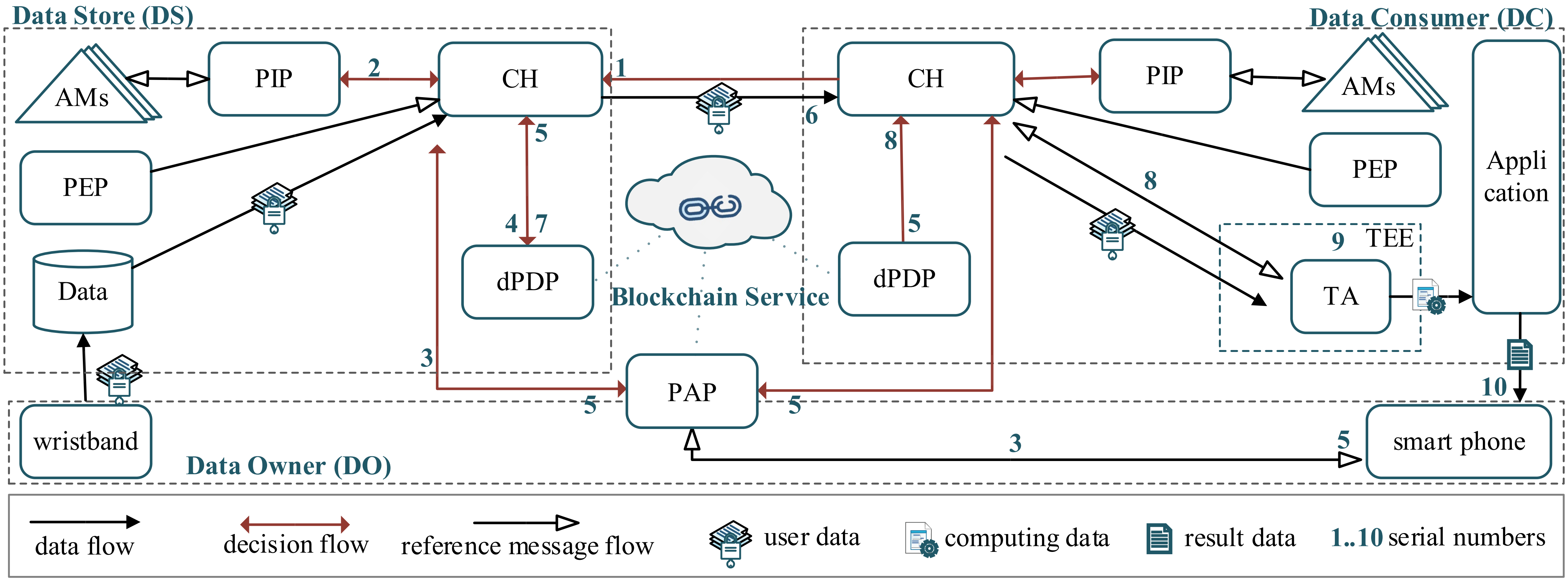
requires that the policy defined by users and enforcement records should not be tampered with.

- **Traceability**

requires that violations must be able to be traced through enforcement records, and are visible to users.

Our Approach: DUCE

A distribute usage control enforcement model for private data sharing by utilizing blockchain [19] technology and TEE.



[19] Maesa, D., Mori, P., Ricci, L.: A blockchain based approach for the definition of auditable access control systems. Comput. Secur. 84, 93–119 (2019).

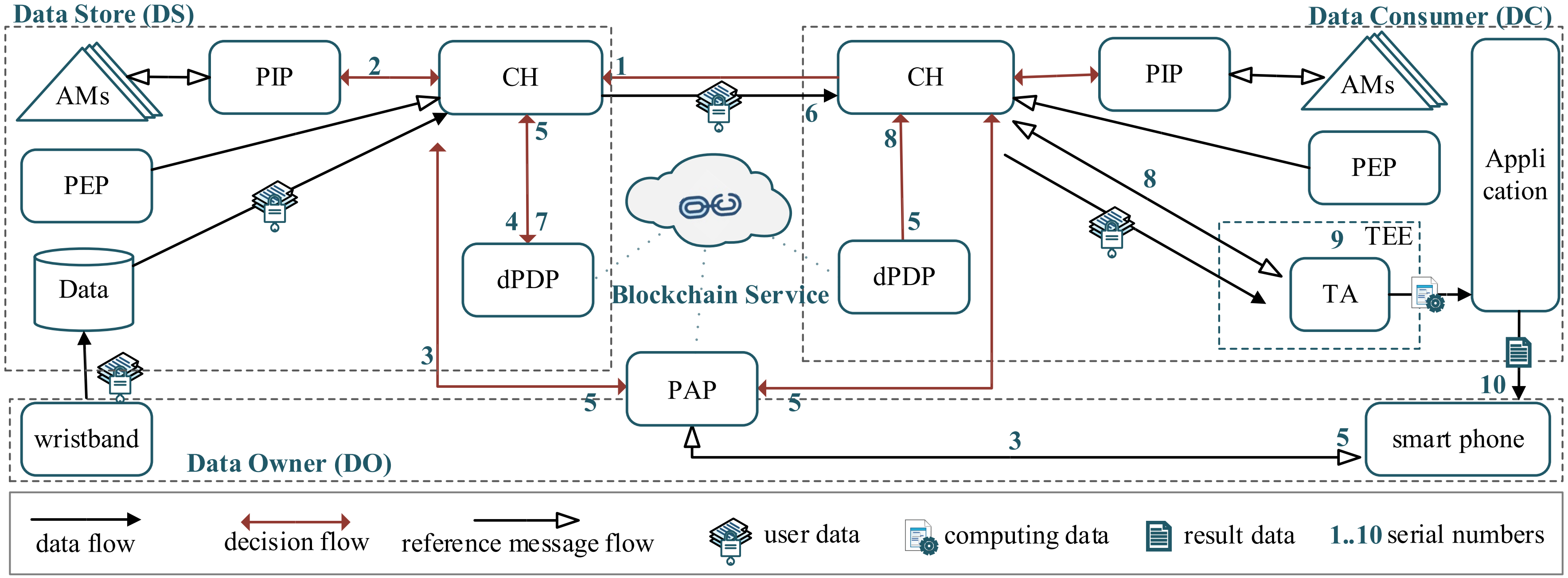
Our Approach: DUCE

- **Enforcement process**

1) Initialization phase.

2) Enforcement phase.

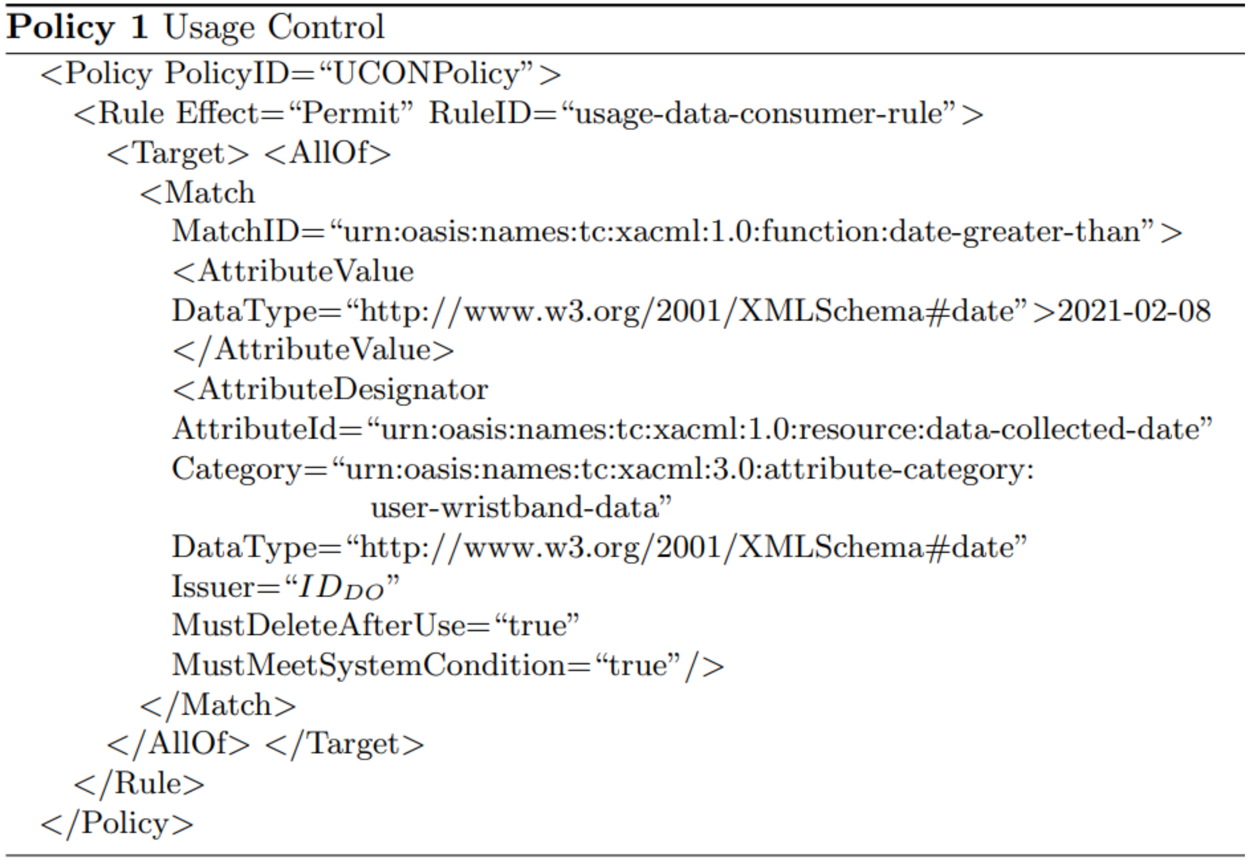
   - Authorization
   - Operation
   - Evaluation
   - Notification



Our Approach: DUCE

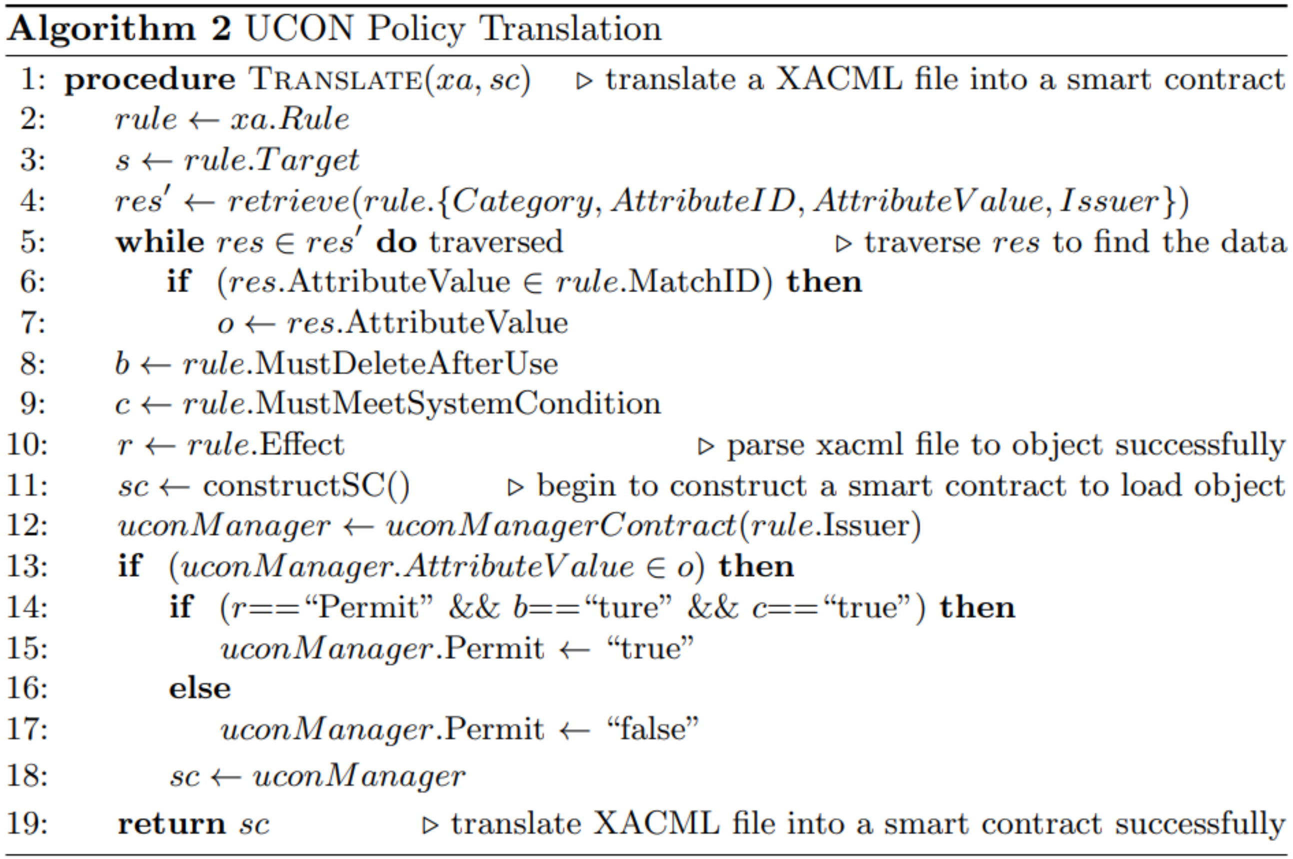
- **Policy Administration**

XACML[3] file -----(translate)----->> Smart Contracts[19]



Policy 1 Usage Control

```
<Policy PolicyID="UCONPolicy">
  <Rule Effect="Permit" RuleID="usage-data-consumer-rule">
    <Target> <AllOf>
      <Match
        MatchID="urn:oasis:names:tc:xacml:1.0:function:date-greater-than">
        <AttributeValue
        DataType="http://www.w3.org/2001/XMLSchema#date">2021-02-08
        </AttributeValue>
        <AttributeDesignator
        AttributeId="urn:oasis:names:tc:xacml:1.0:resource:data-collected-date"
        Category="urn:oasis:names:tc:xacml:3.0:attribute-category:
                    user-wristband-data"
        DataType="http://www.w3.org/2001/XMLSchema#date"
        Issuer="ID_DO"
        MustDeleteAfterUse="true"
        MustMeetSystemCondition="true"/>
      </Match>
    </AllOf> </Target>
  </Rule>
</Policy>
```



Algorithm 2 UCON Policy Translation

```
procedure TRANSLATE($xa, sc$)    ▷ translate a XACML file into a smart contract
    $rule \leftarrow xa.Rule$
    $s \leftarrow rule.Target$
    $res' \leftarrow retrieve(rule.\{Category, AttributeID, AttributeValue, Issuer\})$
    while $res \in res'$ do traversed    ▷ traverse $res$ to find the data
        if ($res$.AttributeValue $\in$ $rule$.MatchID) then
            $o \leftarrow res$.AttributeValue
    $b \leftarrow rule$.MustDeleteAfterUse
    $c \leftarrow rule$.MustMeetSystemCondition
    $r \leftarrow rule$.Effect    ▷ parse xacml file to object successfully
    $sc \leftarrow$ constructSC()    ▷ begin to construct a smart contract to load object
    $uconManager \leftarrow uconManagerContract(rule$.Issuer$)$
    if ($uconManager.AttributeValue \in o$) then
        if ($r$=="Permit" && $b$=="ture" && $c$=="true") then
            $uconManager$.Permit $\leftarrow$ "true"
        else
            $uconManager$.Permit $\leftarrow$ "false"
        $sc \leftarrow uconManager$
    return $sc$    ▷ translate XACML file into a smart contract successfully
```

[3] Anderson, A., et al.: eXtensible access control markup language (XACML) version 1.0. OASIS (2003).

Evaluation

The implementation of DUCE

the baseline: OAuth 2.0

- ✓ Blockchain Service ---- FISCO BCOS[1]
- ✓ TEE ---- SGX [2]
- ✓ Cloud ---- Alibaba Cloud Elastic Compute Service[3]
- ✓ MySQL, Redis

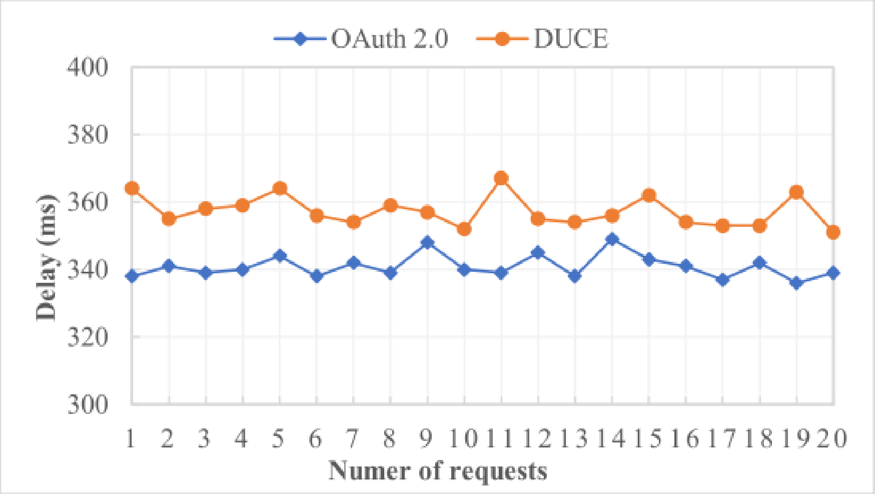
The results (authorization and authentication)

✓ Delay.

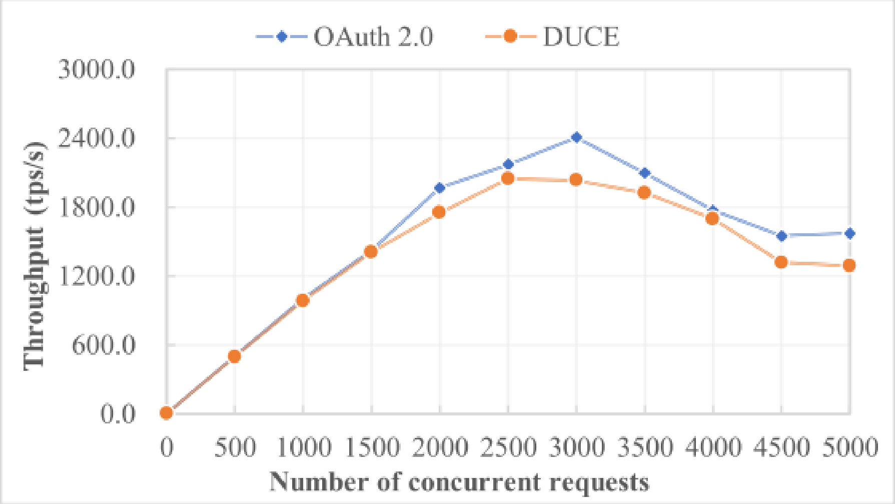
The time required for communication messages transmitting from one network end to another.

✓ Throughput.

the maximum request number that the system can handle per unit time.



(a) Delay performance



(b) Throughput performance

[1] http://www.fisco-bcos.org. [2] https://www.intel.com/content/www/us/en/architecture-and-technology/software-guard-extensions.html. [3] https://www.aliyun.com/.

Summary

A design overview is given with the distributed PDPs and PEPs.

- DUCE leverages permissioned blockchain technology to build a trusted relationship between data-sharing parties, whereby the rules and enforcement records are tamper-proof and visible to users.
- A Trusted Execution Environment (TEE) is used to ensure that the enforcement process of the rules and the usage of user data are trustworthy and controllable by users.

The policy administration model of DUCE is provided.

- A policy example of "delete-after-use" in XACML
- And the policy translation algorithm into Solidity language for smart contracts.

A prototype system is implemented.

- This system is deployed along with an OAuth 2.0 benchmark system.
- The end-to-end delay and throughput are evaluated and analyzed to demonstrate the viability of DUCE.

Thank you for your time !