November 12, 2013

Context-Based Access Control Model for Ridesharing Service

Nikolay Teslya, Alexey Kashevnik, Michael Pashkin

Laboratory of Computer Aided Integrated Systems

St.Petersburg Institute for Informatics and Automation of RAS (SPIIRAS)

Introduction

- Ridesharing service provides possibilities of a real-time fellow-travelers search
- The service needs information from users:
  - Paths
  - Preferences
  - Users' social profiles
- Most of this information cannot compromise the user's privacy, but there can be some information that has to be protected with access control for being used only by defined persons and services.
- For example, people don't often want to share their locations or social profiles to others.

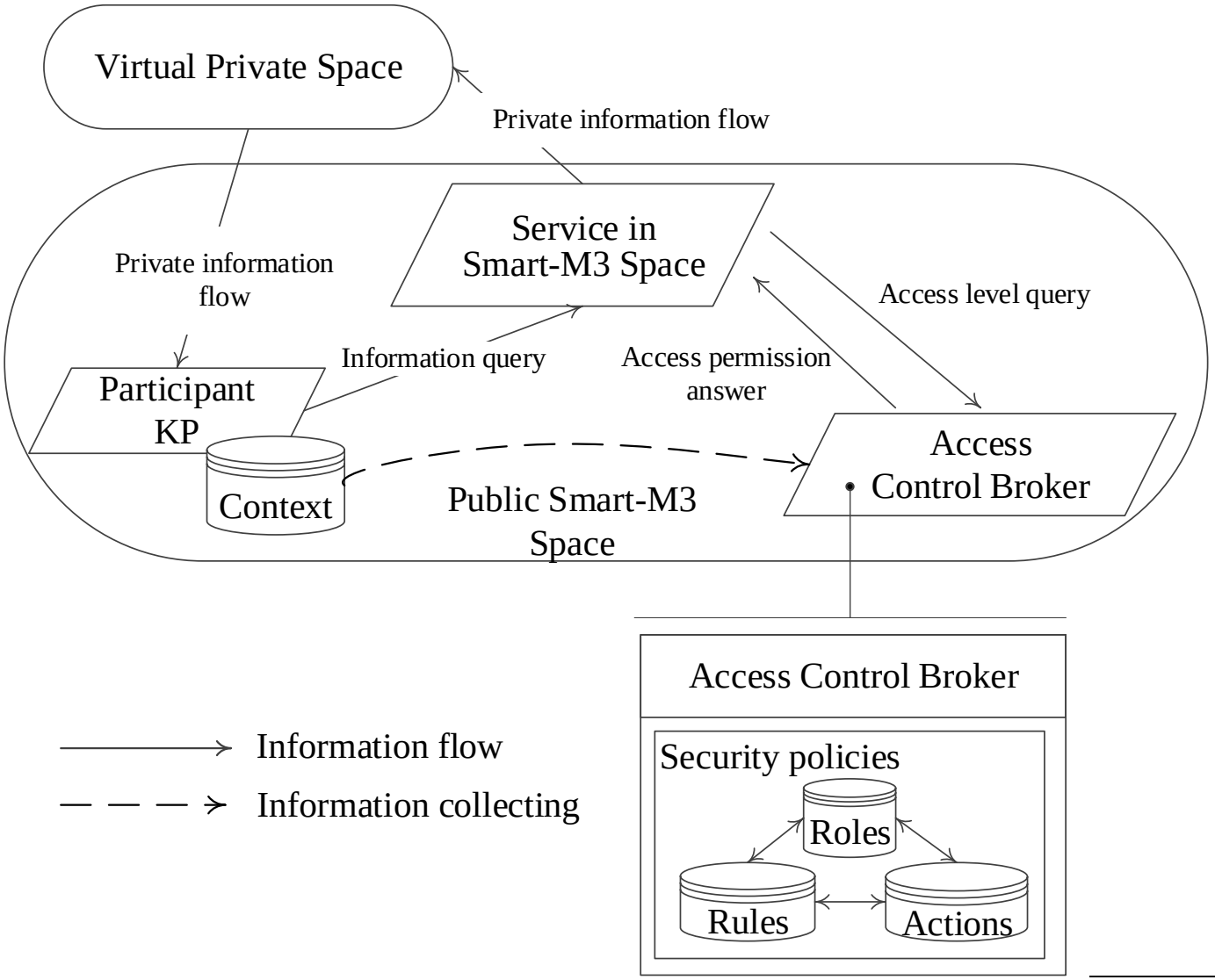
Motivation

- Ridesharing service needs to have an access control to the user's private information:
  - There are no people who want to be tracked (Apple, 2011, storing of user's locations on iPhones)
  - Without the access control, every user can collect locations of other users and predict the future steps for different purposes
  - There are no people who want to share their real names and social networks information for everyone (US Government, 2013, PRISM)
  - Law of any state also restricts collecting any information about any person (e.g. Federal Law № 152 of the Russian Federation)

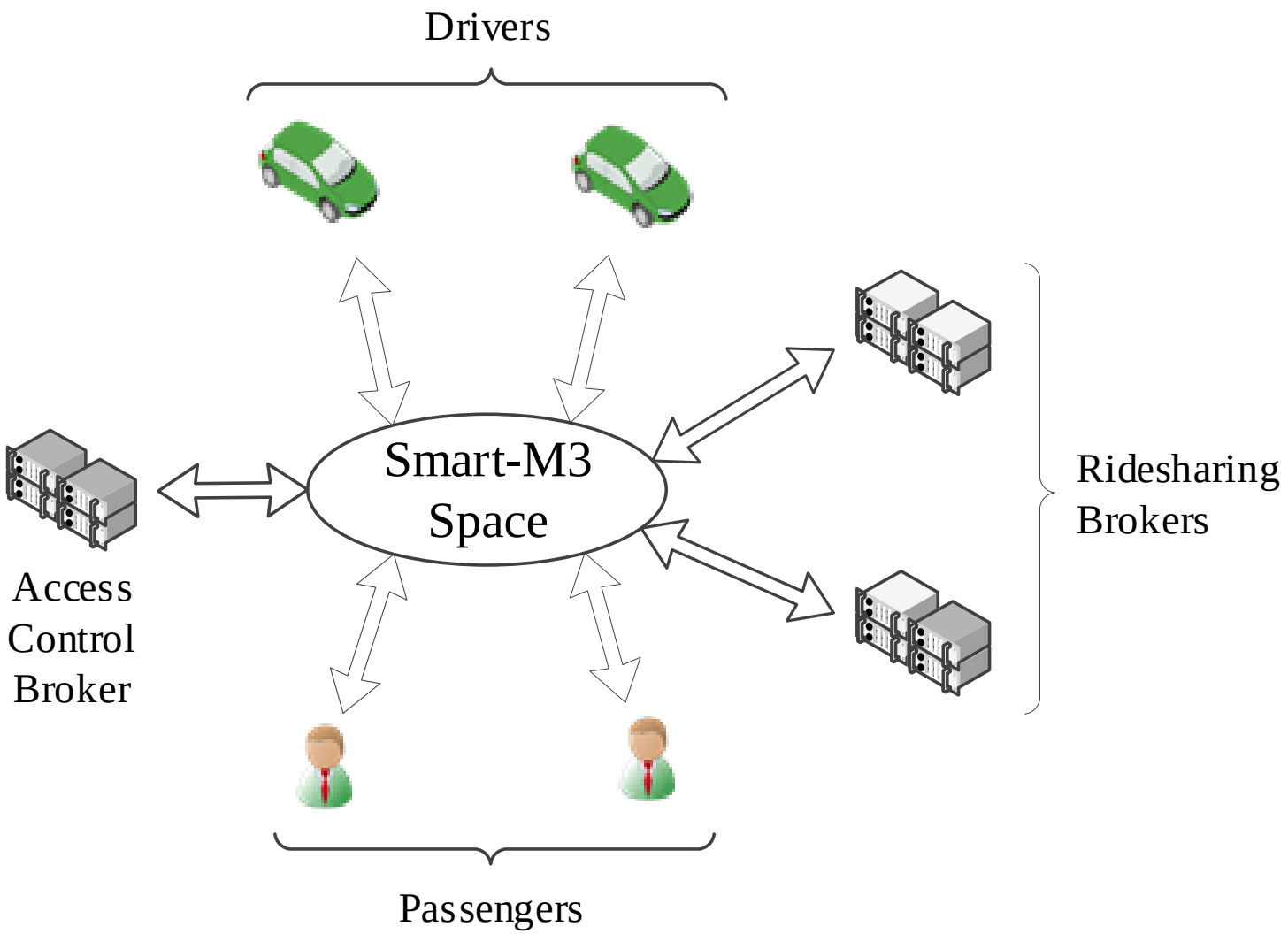
Table of contents

- Conceptual model of the context-based access control module and the ridesharing service
- Context formalization for the ridesharing service
- Rules configuration for access control module

Conceptual Model of Smart Space Access Control Module

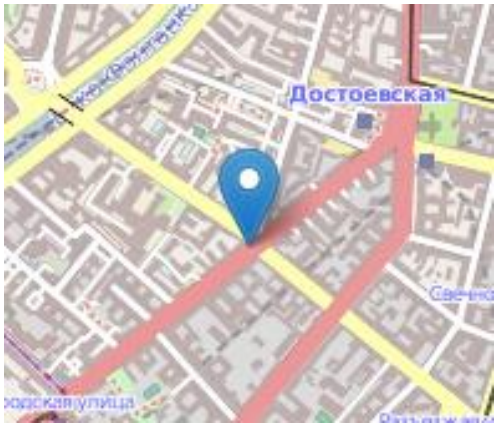


Conceptual Model of Ridesharing Service With Access Control



Context Formalization: Physical Part

- User Location. Latitude & longitude. Building routes and defining place to access information from



- Date and time. Access permission in specified time interval

Context Formalization: Virtual Part

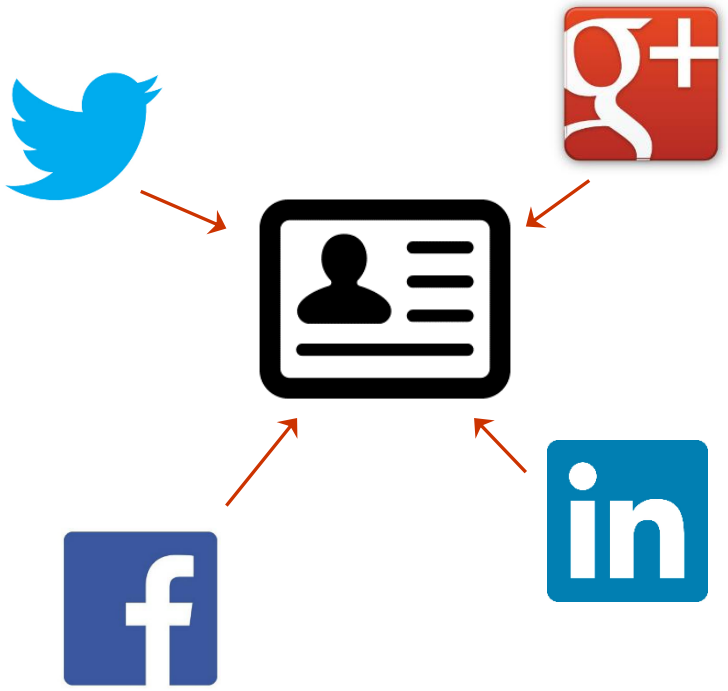
- ID. Unique user's identifier in the service

- Different kinds of authentication information:
  - Password;
  - Private key;
  - Certificate;
  - Device fingerprint (e.g. H. Bojinov, "fingerprinting" a smart phone through its accelerometer, sensor-id.com);
  - IMEI;
  - etc.

Context Formalization: Social Part

- Role in the ridesharing service

- Information from the different social networks:
  - Name, surname;
  - Date of the birth;
  - Birth place;
  - Interests;
  - Friends;
  - etc.



Access Control Policy Rules

- Policy consists of 3 rule's types:
  1) *TrustValue* rules. Used to assign the numeric trust value to the context component.
  2) *Assign_role* rules. Used at the time of access request for role assignment.
  3) *Permissions* rules. These rules contain access control policies, which determine whether a participant with a certain role is allowed to access a particular resource type or not.

TrustValue Rules Examples

- TrustValue(isPassenger == true) = 1;
- TrustValue(isPassenger == false) = 0;
- TrustValue(‘08:00’ < *currentTime* < ‘17:00’) = 0.8;
- TrustValue(‘08:00’ > *currentTime* > ‘17:00’) = 0.2;
- TrustValue(currentLocation *‘in set’* [Russia, Finland]) = 0.9;
- TrustValue(currentLocation *‘in set’* [China, North Korea]) = 0.1;
- TrustValue(commonInterests *‘more than’* 1/2) = 0.7;
- TrustValue(commonInterests *‘less than’* 1/4) = 0.3;
- TrustValue(birthDateAccept *‘before’* 1985) = 0.8;
- TrustValue(birthDateAccept *‘after’* 1985) = 0.2;

Assign_role Rules (1/2)

- Five main roles: owner, trustedPassenger, untrustedPassenger, trustedDriver, untrustedDriver.

- Example:

  - A user has a trustedPassenger role only if:
    - The user is a passenger,
    - His/her current location is Russia or Finland,
    - Current time is from 8 am to 5 pm,
    - His/her birth date is before 1985.

Assign_role Rules (2/2)

- According to the trust values presented before, the rules are set by the following way:

  - AssignRole(trustedPassenger) =
    (TrustValue(isPassenger) = 1) &
    (TrustValue(currentLocation $\in$ (0.7, 1)) &
    (TrustValue(currentTime) $\in$ (0.6, 1)) &
    (TrustValue(birthDateAccept) $\in$ (0.7, 1))

  - AssignRole(untrustedPassenger) =
    (TrustValue(isPassenger) = 0) &
    (TrustValue(currentLocation $\in$ (0.1, 0.5)) &
    (TrustValue(currentTime) $\in$ (0, 0.6)) &
    (TrustValue(birthDateAccept) $\in$ (0, 0.7))

Permissions Rules

- Determines whether a participant with a certain role is allowed to access a particular resource type or not.

- Examples:

  - Permission(trustedPassenger) = "readCommon", "readPrivate";

  - Permission(untrustedPassenger) = "readCommon".

Conclusion

- Access control in the ridesharing service allows users to share their private information only with persons they trust via using the easy configurable preferences
- The ontology of the ridesharing service has been used for defining the context of the service users
- The rules of the access control broker are based on the context of the service users
- The access control broker for ridesharing service helps to prevent the collecting information about users of this service by criminals

Thank you for Attention
Questions are Welcome

E-mail: teslya@iias.spb.su