Distributed service environment (smart spaces) security model development

Kirill Yudenok, Kirill Krinkin
FRUCT LETI Lab,
Open Source & Linux Lab

FRUCT 12th, Oulu, November, 2012
Agenda

- Motivation;
- Goal and tasks;
- Current Smart-M3 security;
- Security model development;
- Smart-M3 security realization:
  - HIP-agent;
  - smart space RDF-graph mapping to the virtual file system (VFS);
- What was done?
- Future research and development;
Motivation

- access control mechanism for the smart space platform, for example Smart-M3;

- protection information mechanism of the space;

- research information security within the smart space area.
Goal and Tasks

The project goal

- Development a security model for distributed service environment (smart spaces, SS), access control algorithms and test developed components as a part of the SS Smart-M3 platform;

The main tasks of the project

- investigation of the basic security models and creation own security solutions;
- development a security model for Smart Spaces;
- modeling and development security model components for the Smart-M3 platform;
- testing developed components and algorithms within the Smart-M3 platform;
Smart-M3 security

What do we have?

- access control at triple level [1];
- context-based and access control policies;
- security objects as triple patterns;

What do we want?

- identification and authentication mechanism of the SS subjects;
- authorization and access control mechanism of SS subjects;
- data privacy;

Security model development

Identification and authentication of space subjects:
  ▶ HIP, PAM;

Authorization and access control of space subjects:
  ▶ discretionary security model;
  ▶ smart space RDF-graph mapping to the virtual file system (VFS);
  ▶ named graphs;
  ▶ access control ontology;
  ▶ security extensions for smart space database.
Smart-M3 security realization

Identification and authentication mechanisms

▶ prospective architecture of HIP-agent;
▶ interaction of HIP-agent components.

Authorization and access control mechanisms

▶ smart space RDF-graph mapping to the VFS;
▶ intermediate solution of the graph mapping;
▶ implementation mechanism to the Smart-M3 platform.
Prospective architecture of HIP-agent

Identification and authentication of the client:

1. Client connection request to the SS;
2. Request intercepting by the HIP-agent;
Interaction of HIP-agent components

The process of connecting the client to the space:

1. Transmission the client hash key to HIP-agent;
2. Checking validity of the hash key;
3. Identification and authentication of the client;
4. Connection to the SS.
Smart Space RDF-graph mapping

- information of SS is stored in a relational database, smart space database (SQLite);
- information of SS is presented in triple form (S, P, O);
- set of triples stored in specific database tables;

**Solution:** The virtual FS, that mapping information of SS in a certain directory structure.

```
SIBFS
  └── objects── p1, p2, ..., pn── s1, s2, ..., sn
  └── subjects── p1, p2, ..., pn── o1, o2, ..., on
  └── predicates── p1, p2, ..., pn
```
The updated directory structure of VFS

- provide more accuracy right to triplets (information) of the space;

```
SIBFS
  └── objects       o1, o2, ..., on ─── p1, p2, ..., pn ─── s1, s2, ..., sn
  └── subjects     s1, s2, ..., sn ─── p1, p2, ..., pn ─── o1, o2, ..., on
  └── predicates   p1, p2, ..., pn
```
The intermediate solution of the graph mapping

- Working with SS database: get all triples and save them in memory of data structure (SQLite):
  - receiving all objects, subjects, predicates and their values;
- Creating a VFS directory structure based on the data:
  - creating of virtual FS using FUSE technology (fusekit), setting permissions;
Implementation mechanism to the Smart-M3 platform

- modification of Smart-M3 platform piglet module:
  - piglet proxy creation for new extensions;
  - replacement of all smart space database operations to mapping FS operations;
  - determine and verify client access permissions;
- testing operations on the client side.
What was done?

- analyzed and designed the HIP protocol-based mechanism of identification and authentication;

- the mechanism of authorization and SS subjects access control by mapping RDF-graph to the virtual file system is developed; mechanism tested in the Smart-M3 platform;

- the implementation process of HIP-agent and mapping mechanism to the Smart-M3 platform is started;
Future research and development

Main
- HIP-agent development;
- implementation of mapping model to Smart-M3 platform;
- set permissions tool development for mapping FS;

Additional
- named graph authorization system development;
- adding developed mechanisms to new version of Smart-M3 platform (Redland);
Questions & Answers

Kirill Yudenok, Kirill Krinkin
{kirill.yudenok, kirill.krinkin}@gmail.com
Open Source & Linux Lab,
http://osll.fruct.org, osll@fruct.org

FRUCT 12th, Oulu, November, 2012