Secure Dissemination of EHR in Untrusted Cloud

Denis Ulybyshev[1] , Bharat Bhargava[1] , Leon Li[2] , Jason Kobes[2], Donald Steiner[2], Harry Halpin[4] , ByungChan An[1] , Miguel Villarreal[1], Rohit Ranchal[3], Tim Vincent[1]

[1]Computer Science and CERIAS, Purdue University;
[2]NGC Research Consortium;
[3]IBM Watson Health Cloud;
[4]MIT

Tutorial

Purdue University, 2016

Table of Contents

Initial Setup

Supported OS: Linux

1. Create a <Project_Folder>, go to that folder
   $ cd Project_Folder

and clone the source code from the repository:
   $ git clone https://github.com/Denis-Ulybysh/absoa16/

2. go to folder *absoa16/console* and install ***npm***:
   $ cd console
   $ sudo npm install

node_modules folder must appear in the project directory. Then do the global update of "Express" framework :
  $ npm update -g express

3. Go to *Project_Folder/absoa16/scenarios/webcrypto* and run 'install' script. It will install npm for every service: doctor, insurance, researcher, hospital, authenticator.
   $ ./install

4. Install ***mysql*** :
   $ apt-get install mysql-server-5.6

5. Set up MySQL database:
  $ mysql -u root -p < db.sql

6. Install ***Java Runtime Environment***. The project was tested under JRE version 8.

7. Run the project: go to *Project_Folder/absoa16/scenarios/webcrypto* and run 'start' script:
  $ ./start

8. Open the browser, go to *http://localhost:3000/* (default port for prototype is 3000). Internet connection is required. You should see the page from Fig.1 below.

*To run prototype on a remote machine (not on localhost) modify 2 lines in the following source code files: *Project_Folder/absoa16/scenarios/webcrypto/hospital/public/index.html*
and
Project_Folder/absoa16/scenarios/webcrypto/authenticator/public/index.html
-replace localhost with real IP address where prototype (start script) will be running

The server can be run either on a remote machine or on localhost. In our scenario server is running at waxedprune.com, port 3000. When you open the browser and type the following URL:
http://waxedprune.com:3000/
you will see the following page:

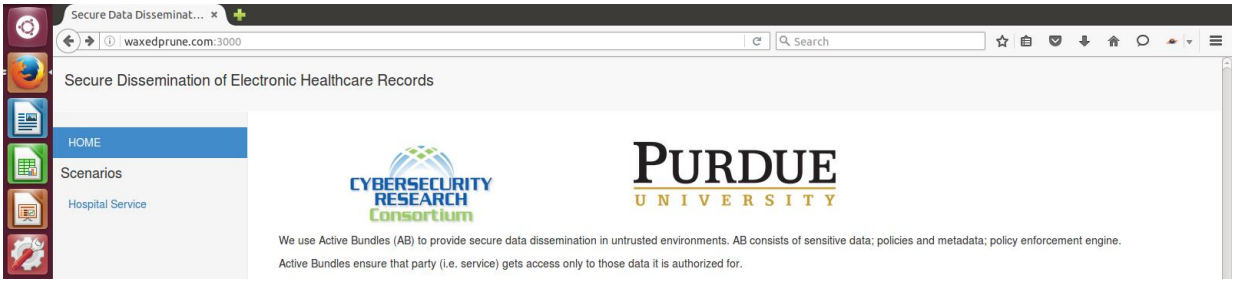

Fig.1. Secure Dissemination of EHR Main Page

Click "Hospital Service" button. Then you will see the following page:

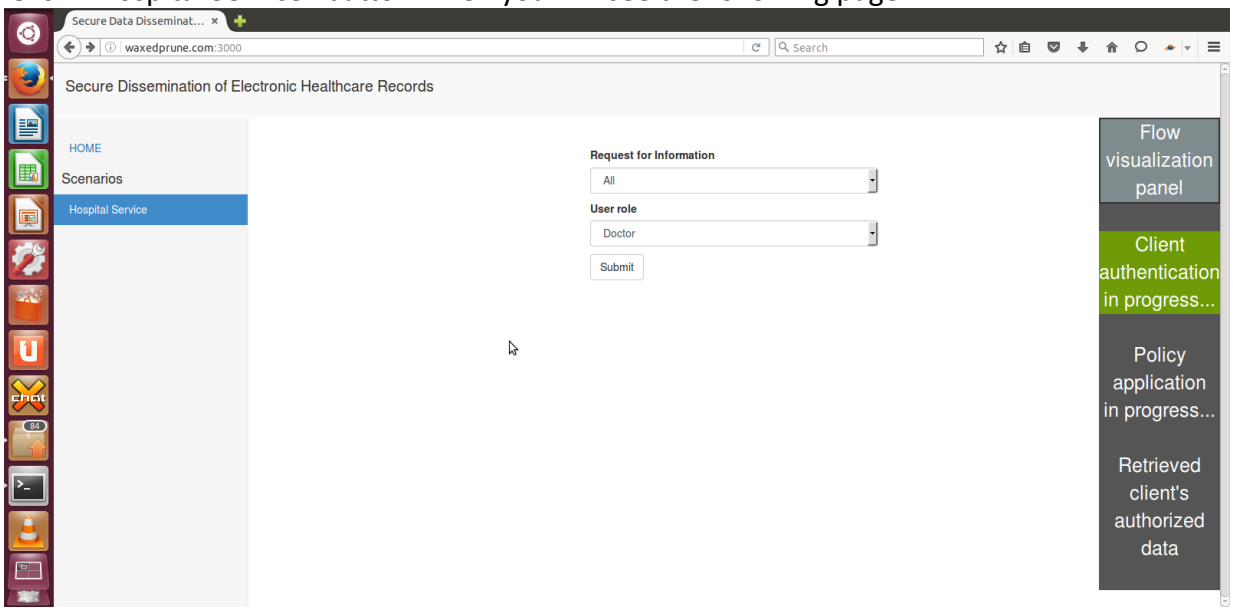

Fig.2. Hospital Service Page

Select the role (Doctor, Insurance company or Researcher) and select information to request: 'All' or specific field, e.g. Contact Info of a Patient. Then click 'Submit' button. After that you will be redirected to authentication server (AS) web page, where you need to enter credentials. Since https protocol is used between client and authentication server, security exception needs to be added in the browser.

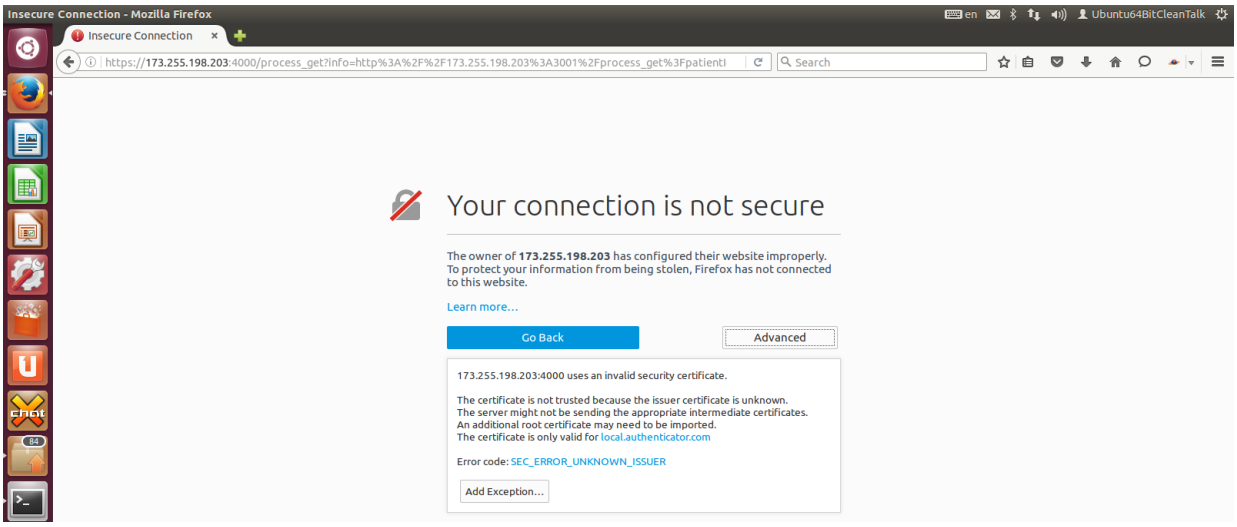

Fig.3. Adding security exception for https connection between client and AS

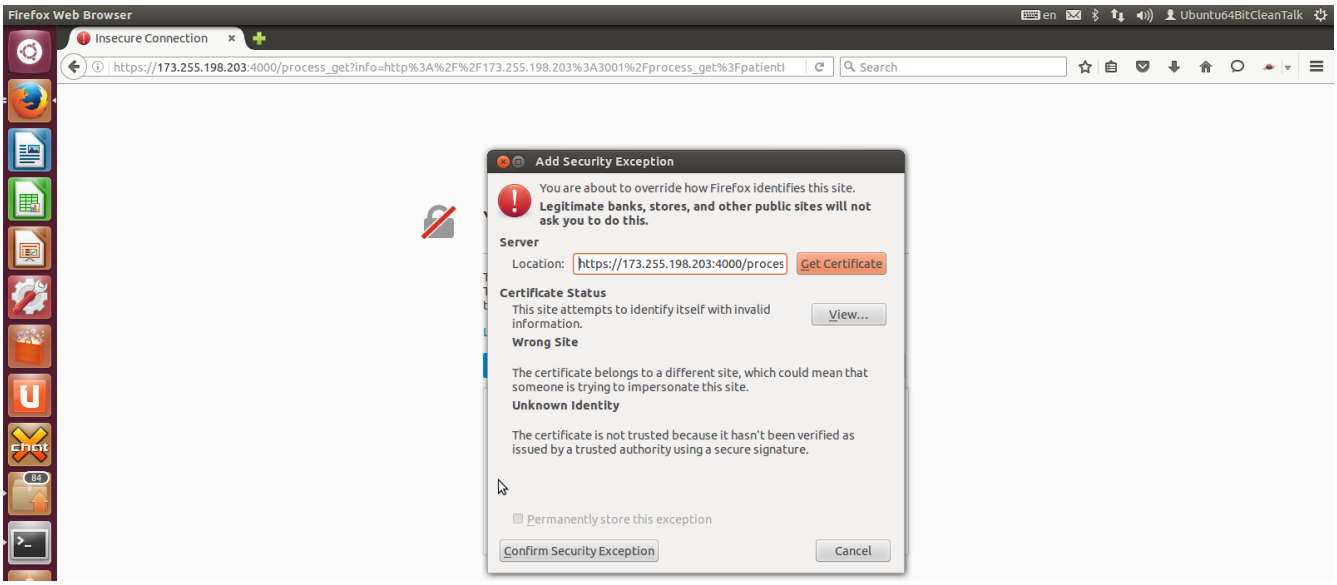

Fig.4. Security exception confirmation for https connection between client and AS

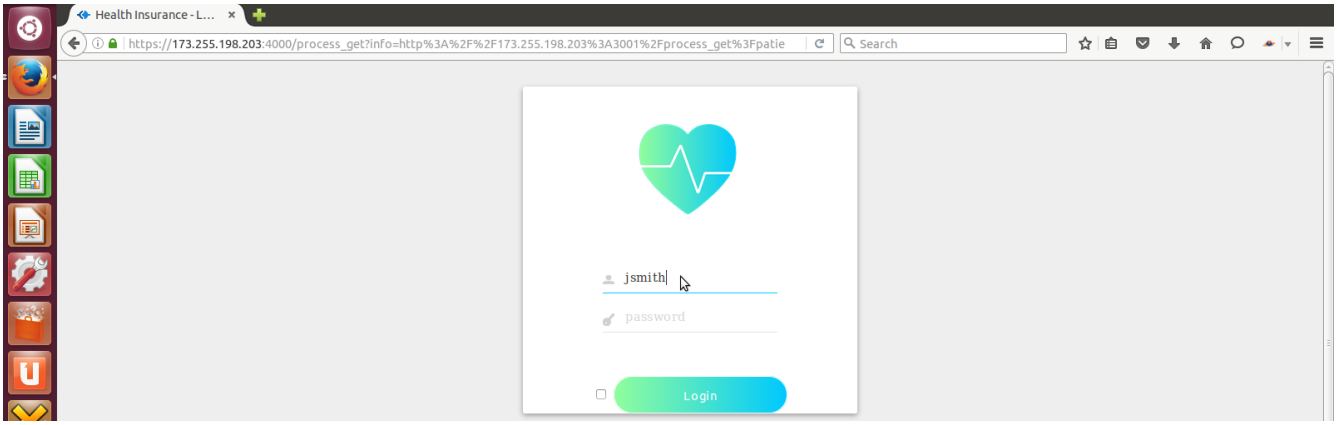

Fig.5. Password-based authentication of a Client at AS

If credentials entered by the Client are incorrect then data request is rejected. If credentials are correct then cryptographic capabilities of a browser are determined. If the level of cryptographic capabilities is high then Doctor will get data on behalf of 3 patients (see Fig.6), whereas if level of cryptographic capabilities is low then Doctor will get data on behalf of only 1 patient (see Fig.7). The idea behind this is that if level of cryptographic capabilities is high then Doctor can get data on behalf of patients assigned to other doctors, in addition to data of a patient (Monica Latte) assigned to the given Doctor.

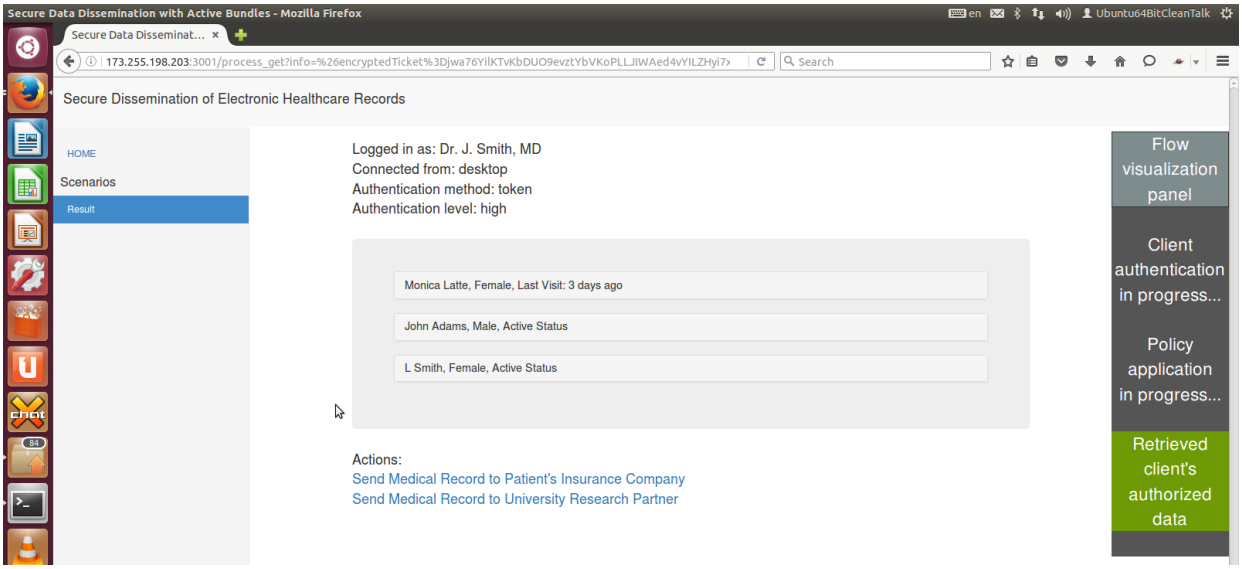

Fig.6. EHRs retrieved for Doctor with High Level of browser's crypto capabilities

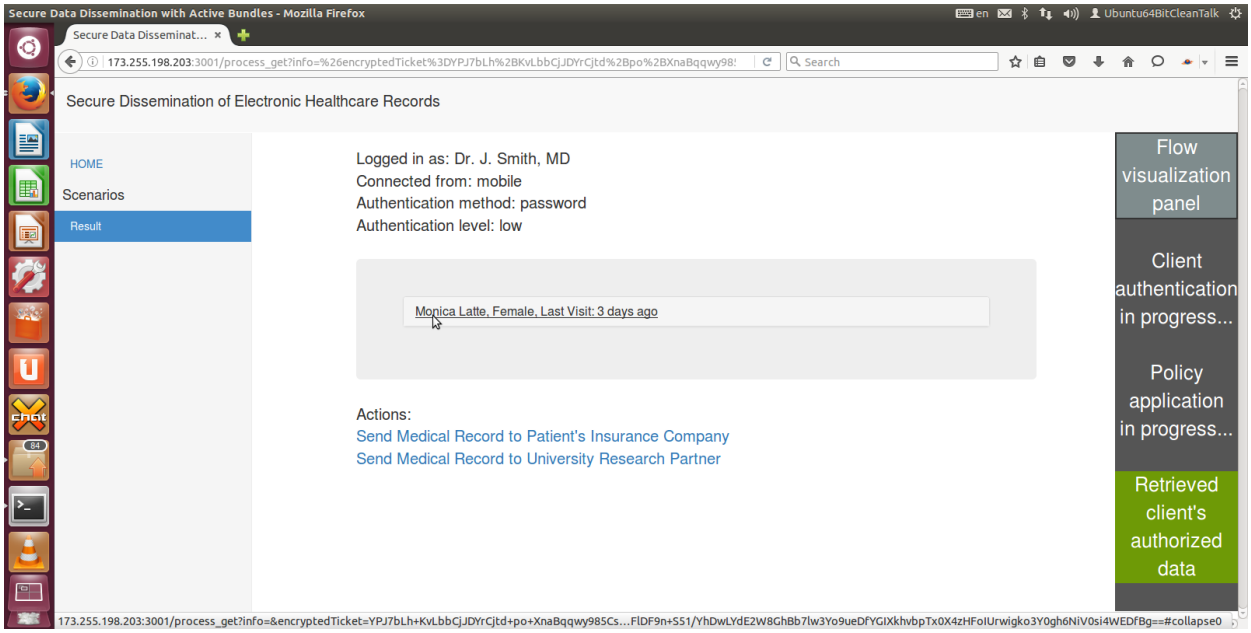

Fig.7. EHR retrieved for Doctor with Low Level of browser's crypto capabilities

Each party (corresponding to the Role) interacts with an Active Bundle (AB), running on a cloud platform, to access only those data from AB for which the party is authorized.

If client (e.g. Doctor) clicks on a patient's record, then detailed information on behalf of that patient is displayed:

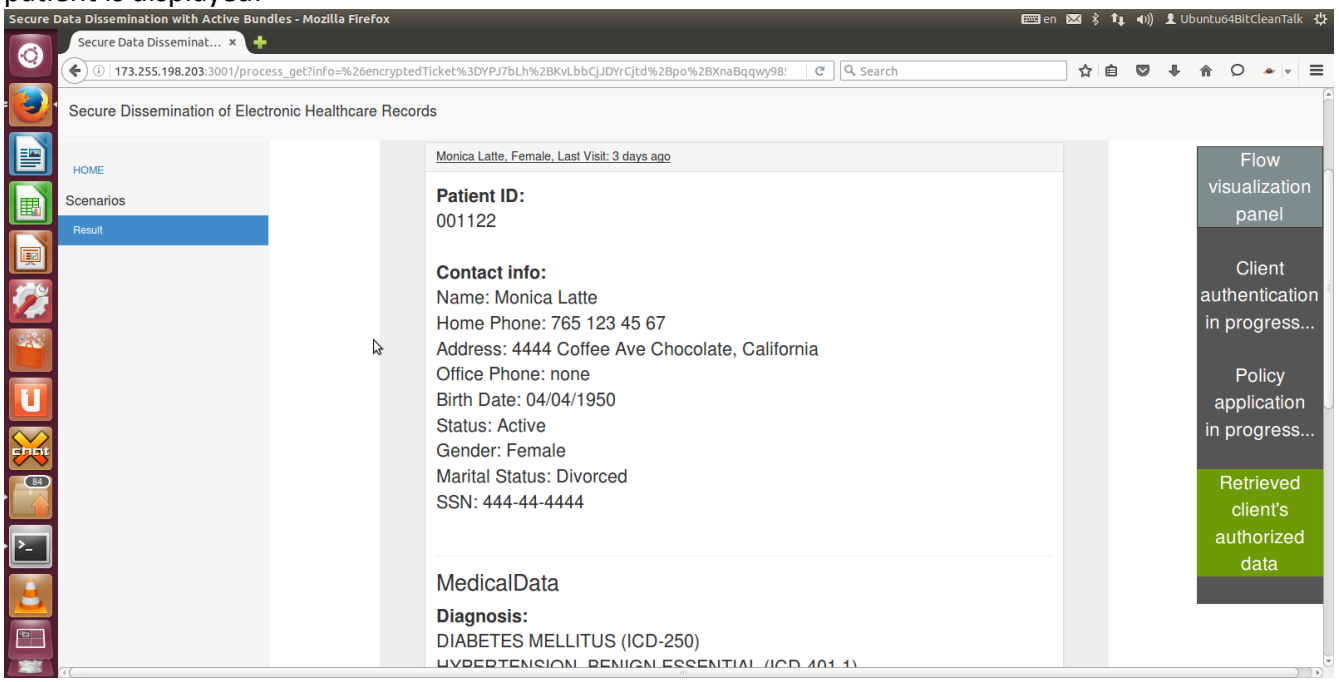

Fig.8. Detailed EHR of a patient retrieved for Doctor

Doctor can also select among 2 actions "Send Medical Record to Patient's Insurance Company" or "Send Medical Record to University Research Partner". Then Active Bundle, containing EHR on behalf of a patient, will be sent to another cloud, i.e. to corresponding service running in another cloud.

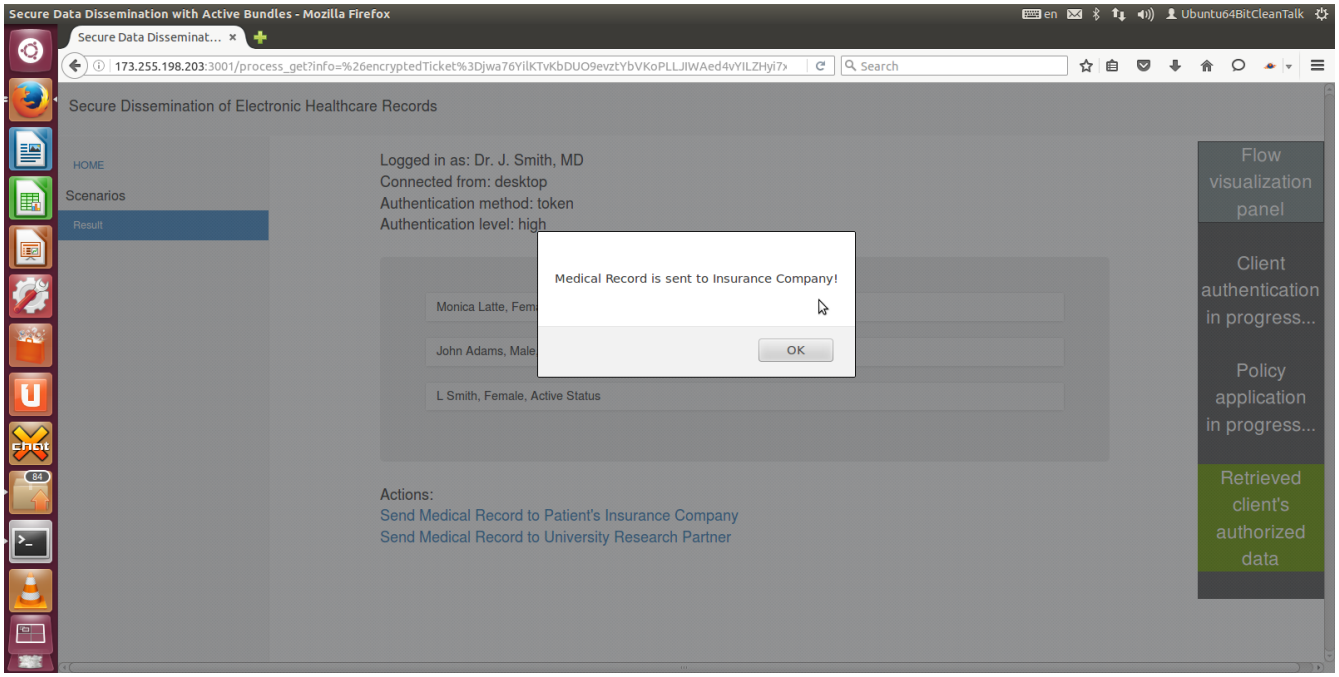

Fig.9. EHR sent to service running in another cloud

According to specified Role-Based Access Control Policies, Doctor can get access to Contact, Medical and Billing Information of a Patient; Insurance company can get access to Contact and Billing information (see Fig. 10 below); Researcher can get access to Medical and Billing Information (see Fig.11 below).

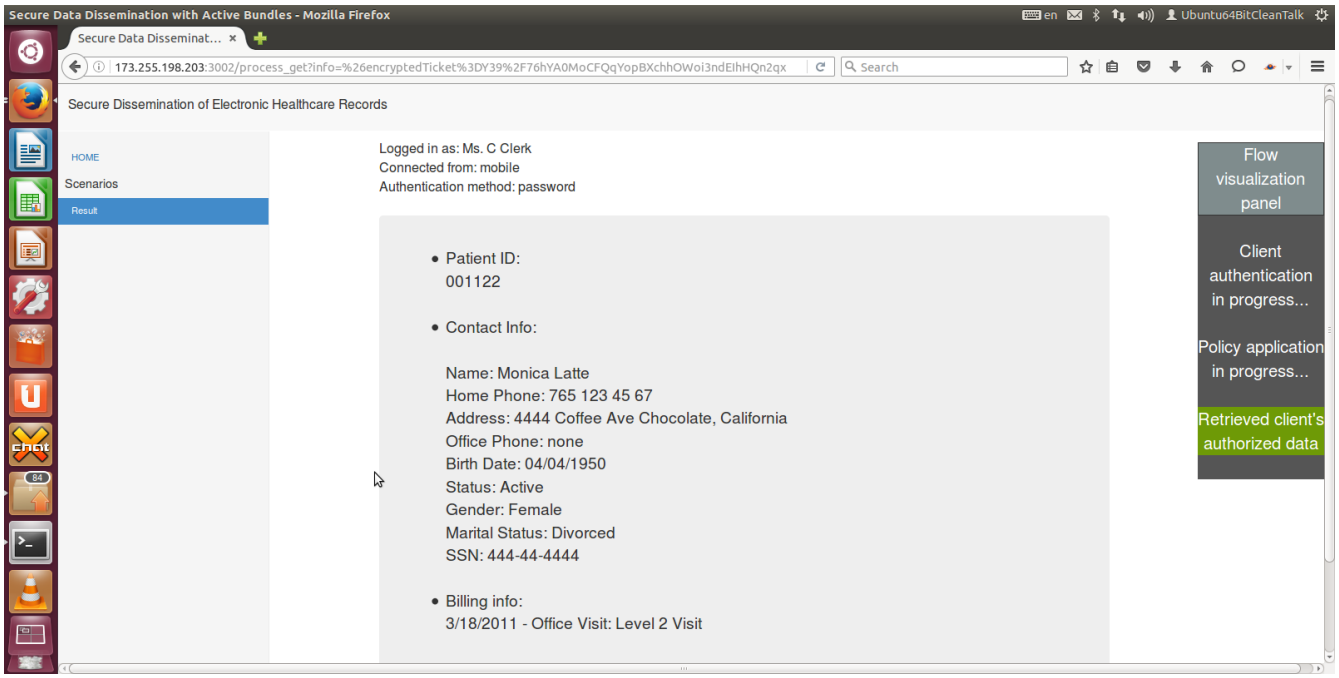

Fig.10. Patient's data accessible by Insurance Company

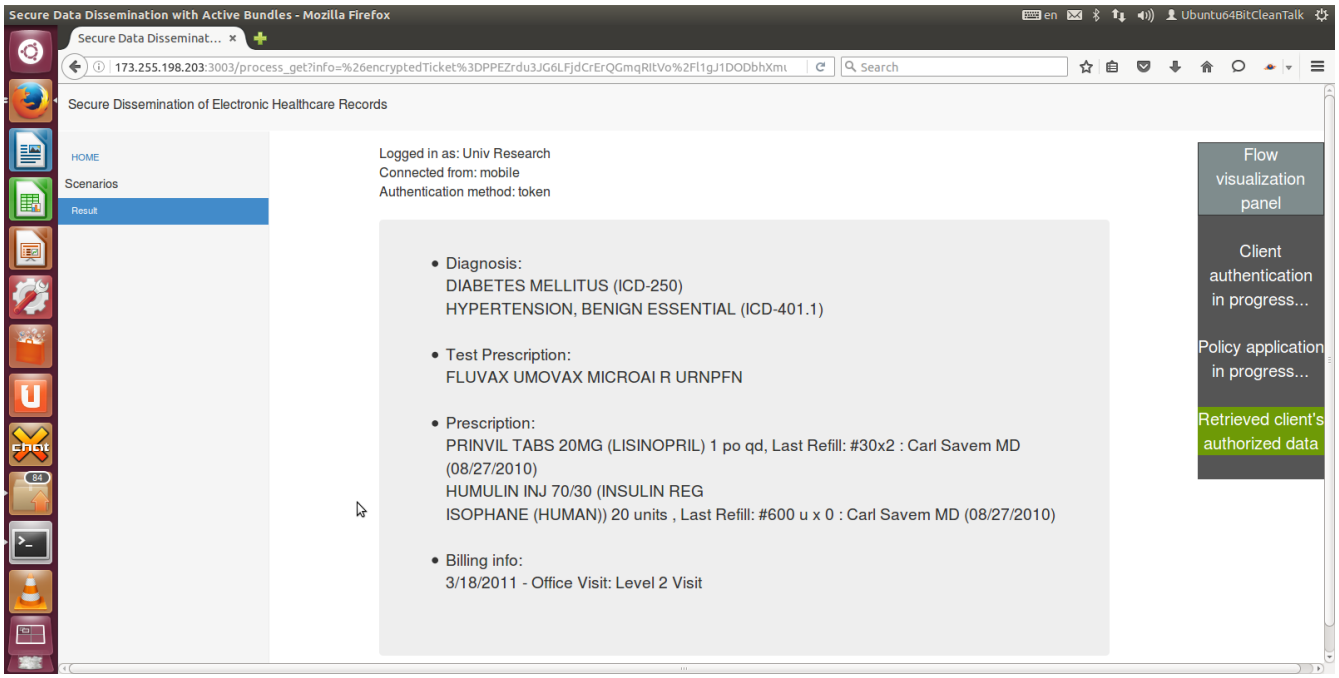

Fig.11. Patient's data accessible by University Researcher

It should be said, that Active Bundle is tamper-resistant: if malicious user tries to modify policies

or AB code or bypass policy check then tamper attack will be detected and data access will be denied. Context-based and trust-based data dissemination is also supported by Active Bundles.

On the right side of web pages (see Fig.12) you can see flow visualization panel: it shows the phase of a data access process. Initially, client authentication phase is in progress. Then, after client enters correct credentials, data request is transferred to the Active Bundle and 'Policy application' phase starts. Then role-based access control policies specified in Active Bundle are applied abd finally, data for which the party (according to the Role) is authorized are retrieved and sent to the Client.

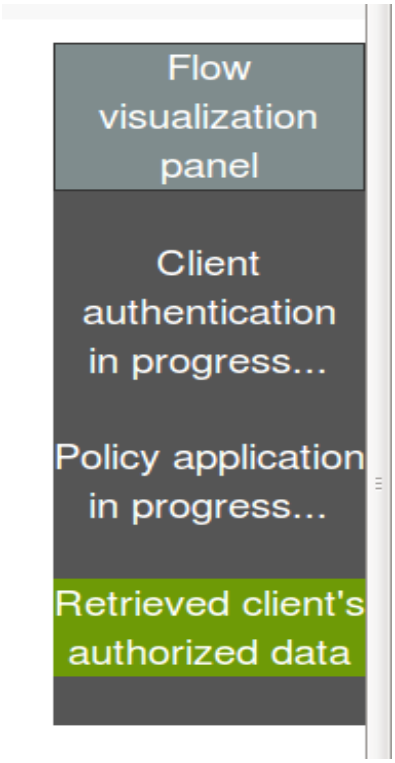

Fig.12. Flow visualization panel