An Open Source Toolkit for Medical Imaging De-Identification

D. Rodríguez González[1,2,3], T. Carpenter[1,3], J.I. van Hemert[1,2], J. Wardlaw[1,3]

[1] SINAPSE collaboration www.sinapse.ac.uk

[2] National e-Science Centre, School of Informatics, University of Edinburgh, UK

[3] SFC Brain Imaging Research Centre, Division of Clinical Neuroscience, University of Edinburgh, UK

Phone: +44 (0)1315372663

Fax: +44 (0)131537 2661

Email: david.rodriguez@ed.ac.uk

Project URL: http://www.sinapse.ac.uk

Abstract

Objective: Medical imaging acquired for clinical purposes can have several legitimate secondary uses in research projects and teaching libraries. No commonly accepted solution for anonymising these images exists because the amount of personal data that should be preserved varies case by case. Our objective is to provide a flexible mechanism for anonymising DICOM data that meets the requirements for deployment in multicentre trials.

Methods: We reviewed our current de-identification practices and defined the relevant use cases to extract the requirements for the de-identification process. We then used these requirements in the design and implementation of the toolkit. Finally, we tested the toolkit taking as a reference those requirements, including a multicentre deployment.

Results: The toolkit sucesfully anonymised DICOM data from various sources. Furthermore, it was shown that it could forward anonymous data to remote destinations, remove burned-in annotations, and add tracking information to the header. The toolkit also implements the DICOM standard confidentiality mechanism.

Conclusion: A DICOM de-identification toolkit that facilitates the enforcement of privacy policies was developed. It is highly extensible and provides the necessary flexibility to account for different de-identification requirements, but at the same time, it has a low adoption barrier to new users.

Keywords: Digital Imaging and Communications in Medicine (DICOM), Privacy Policies, Data Protection Act (DPA), De-Identification, Anonymisation, Toolkit, Pseudonymisation

Introduction

Medical imaging acquired for clinical purposes not only constitutes a valuable resource for patient diagnosis, but is also of great value for secondary uses in research and education. These secondary uses must respect patient privacy and should abide by the corresponding legal framework, which, in the case of UK legislation, is principally[1] the Data Protection Act (1998) [1].

The Data Protection Act does not affect anonymous data, i.e. data where all personal identifiers have been permanently removed, but some personal data is usually required in the aforementioned secondary uses.

[1] There are other pieces of legislation that apply, like the common law duty of confidentiality or the Human Rights Act.

Even when patients give informed consent to the processing of their personal data outside the clinical environment or the point of acquisition in the research environment, it is desirable and good practice that the data are rendered “anonymous” before transferring it. Nevertheless, in some cases, there is a legitimate need to keep some kind of link between the de-identified data and the personal information. Such a link can be either unidirectional, or bidirectional. Unidirectional links facilitate the addition of new pieces of information about the same subject using the same pseudonymous identification, but they do not allow recovering the personal data from the anonymous data. Whereas with a bidirectional link there is a coded identifier that links the anonymous data back to the personal data making it possible to identify the person if necessary (such as might be desirable in the case of an important incidental finding).

To clarify what are and what are not anonymous data, we can refer to the definitions provided by the UK Medical Research Council (MRC) in their “Personal Information in Medical Research Guide” [2]:

- “*Coded information* contains information which could readily identify people, but their identity is concealed by coding, the key to which is held by members of the research team …” Coded information is not anonymous.
- *Anonymised data* where personal information is concealed in such a way that the researchers receiving it cannot identify the person. The MRC further distinguishes:
    - “*Linked anonymised data* are anonymous to the research team that holds it, but contains coded information which could be used to identify people”.
    - “*Unlinked anonymous data* contain nothing that has reasonable potential to be used by anyone to identify individuals: the link to individuals has been irreversibly broken”.

Nevertheless, even with unlinked anonymous data there is still a potential risk of subject identification through combinations of data held by the research team or other people with access to the information.

Another difficulty is that it is not possible to build general rules that will suit all cases. For instance, in the case of rare diseases, the condition itself is a factor that increases the potential for identification and greater precautions have to be taken, while using the same criteria in other cases can lead to unnecessary destruction of useful information. In the case of imaging data there is the additional risk of identification from the pixel information, for instance a 3D reconstruction of the head can produce a potentially recognisable face [3].

DICOM Standard and Data De-Identification

Digital Imaging and Communications in Medicine (DICOM) [4], the standard for medical imaging communications and storage, defines an attribute level confidentiality mechanism. In part 3 (PS 3.3-2008), a procedure is defined to encrypt attributes and store them in the DICOM header (in the “Encrypted Attributes Sequence”); while part 15 (PS 3.15-2008) Annex E defines the “Basic Application Level Confidentiality Profile” (the *Basic Profile* in the following), which lists the attributes that must be protected to provide a minimal level of confidentiality. Nevertheless, it seems that this mechanism has not been widely adopted yet.

The de-identification process of the DICOM header must be adapted to the specific legal framework of each country and to the particular circumstances of the use case. For example, depending upon the specific use case data like patient weight or sex may or may not be relevant.

Furthermore, there is no consensus on several points, for instance, whether DICOM Unique Identifiers (UIDs) should be changed or not. Another contentious point is the level of ambiguity necessary for anonymising some data, such as dates or postal codes, since even with relatively small amounts of personal information it might be possible to identify an individual, especially in the case of rare conditions.

Multicentre Clinical Research Projects

Imaging studies conducted in a single centre are typically small, so they can lack power and generality. That is the reason why multicentre clinical imaging research projects are becoming increasingly important; they provide larger cohorts of subjects with greater clinical relevance, more robust research results, generality and statistical power. They also encourage communication and interchange of ideas between researchers and avoid duplication of efforts.

In terms of privacy protection, in multicentre studies it is desirable that uniform anonymisation policies are adopted by the collaboration including a common data de-identification framework. At the same time, it is advisable to de-identify data near the point of acquisition before transfer for centralised analysis and storage.

Existing Tools

There are now a considerable number of tools (both commercial and open source) for de-identifying DICOM data. Such tools employ various de-identification strategies with approaches that range from removing all patient related information in an automated way, which often renders the data useless, to manual editing with absolute freedom, which is impractical and error prone. Some software fails to provide comprehensive policies, e.g., removing the attribute "Patient Name" (0010,0010) and not the attribute "Other Patient Names" (0010,1001) thereby effectively disclosing the patient's identity.

We examined some quite diverse tools as examples and we will present them in the rest of the section; Table 1 summarises their main features.

Table 1 Features comparison of several DICOM anonymisation tools.

	PrivacyGuard	**Sante DICOM Editor**	**Clinical Trial Processor**	**Universal De-Identification Platform**	**Gdcmanon**
License	Open Source	Commercial	Open Source	Commercial	Open Source
Platform	Multiplatform (Java)	Windows XP/Vista	Multiplatform (Java)	Multiplatform (Java)	Windows, Linux, MacOSX
Anonymisation Library	Java	No	No	Java	No
Application types	Standalone process, Windows & Linux service	Desktop application	Standalone process with web interface	Standalone process, J2EE application	Standalone process
Data Input	DICOM, file system	DICOM, file system	DICOM, HTTP	File system, web service, JMS	File system
Data Output	DICOM, file system, SFTP	File system, DICOM	DICOM, HTTP, database	Not specified	File system
Burned-in annotation removal	Configurable removal method	Define up-to 4 rectangles	No	No	No
DICOM ALC[2] support	Full	*Basic Profile* template only	*Basic Profile* template only	No	Full
Policy definition	XML documents	Templates	Scripting (template)	Configuration files	No
New methods	Java classes	No	Limited (scripting)	Java classes & scripts	No

Sante DICOM Editor is a commercial (Windows only) application that provides an anonymiser [5]. This anonymiser can process a single file, a study or all the files contained under a folder recursively. It also provides a mechanism to remove burned-in annotations by defining up to four rectangles that can be set to the background colour. Overall, this tool, clearly designed for desktop usage, is not adequate for batch processing in a production deployment for a clinical trial or similar.

The RSNA (Radiological Society of North America) Clinical Trial Processor (CTP) [6] is a stand-alone application that provides a customisable workflow and some extension possibilities. The

[2] Attribute Level Confidentiality

anonymisation service is configurable via a script language, but this limits the possibilities for new anonymisation methods. CTP does not support the DICOM standard anonymisation mechanism, and does not provide a solution for dealing with burned-in annotations. This package is designed for clinical trials and incorporates a DICOM receiver, but it cannot be installed as a Windows service, so it has to be manually restarted every time the computer is rebooted.

The Universal De-identification Platform[3] (UDiP) [7], developed by IBM Haifa Labs, aims to provide a comprehensive solution to data de-identification of different data formats (including DICOM). It is available both as a Java library, and as a ready to use application. However, this commercial solution might be expensive and needs to be adapted to the local legislation.

Recently (April 2009) the open source C++ Grassroots DICOM library (GDCM) [8] has introduced a command line tool for anonymisation implementing the *Basic Profile* and the DICOM standard encryption based mechanism. This simple command line tool provides few features, so it is not suitable to be used “out of the box” for a clinical trial or a teaching library.

Materials and Methods

This work has been developed in the framework of SINAPSE (Scottish Imaging Network: A Platform for Scientific Excellence) [9], a collaboration that seeks to enable multicentre clinical studies in brain imaging. To identify the requirements for DICOM de-identification in SINAPSE, we first reviewed the de-identification practices in the participating centres, and defined use cases to extract relevant information. We reviewed several existing tools and found that none was adequate for our purposes. Therefore, we took those requirements, used them to design a DICOM de-identification tool and implemented it. We then compared the result with those of existing tools. Finally, the requirements were updated using the feedback received from users during the development of *PrivacyGuard*, and were then incorporated into the next development cycle.

De-Identification practices and requirements

Although regulations on de-identification exist, it seems that there is no common well-established procedure to implement them. Instead, a multitude of different tools are in use, each with its own interfaces and features. In the UK, the CHERRI (Common Healthcare Educational Recordings Reusability Infrastructure - Practice, Interoperability and Ethics) project [10] has carried out surveys that “revealed that most practitioners were aware of relevant legislation and national guidance relating to patient consent and confidentiality but did not know how to translate these into processes in the local context”.

Software currently in use does not provide a satisfactory default de-identification of DICOM data. This is dangerous because some users might not be aware of the necessary steps to obtain an acceptable anonymous dataset with such software and unintended personal data disclosures can happen as a consequence. Other software will remove (by default) too much data rendering it useless for the intended purpose. Some anonymisation applications just hide personal information in private attributes (mentioned in chapter 11 “DICOM Security” in [11]). In some other cases, personal information is disclosed in file or folder names and de-identification tools commonly overlook this.

Depending on the use case, further measures might be needed, for instance, to avoid the risk of face recognition in a 3D reconstruction, but this is outside the scope of this work.

Use Cases

Another, important aspect is the various intended uses of data. We have considered the following three scenarios in the context of brain imaging:

1. Sharing data in a multicentre trial: the amount of personal data to be kept depends heavily on the type of study. In long-term studies (e.g. on ageing), it is fundamental to keep the data for many years and to preserve the possibility of linking the data back to the individual for follow-up studies.

[3] Also known as IBM De-identification Framework for Compliance to Privacy Laws.

2. End of project data publication. Data must be fully anonymous, but secondary analysis might require the subsequent linking of new data, and in such cases the use of an irreversibly encrypted identifier would be useful.

3. Education. This is again a case for unlinked pseudonymisation. Some personal data, such as age, should be retained, as they are useful for the purposes of education.

Requirements

The following essential requirements for data de-identification were identified in the review of the use cases described above:

1. The data acquired in a clinical environment must be anonymous before leaving that environment.
2. The tool must function on any modern computer platform; it should not depend on the operating system used.
3. The tool must have either no or at least a very low adoption threshold. Ideally, it should not interfere with daily departmental operations and workflows.
4. The creation and management of privacy policies must be relatively quick and straightforward for policy administrators.
5. The tool must be able to handle images with burned-in annotations.
6. The tool must support the attribute-level confidentiality mechanism defined in the DICOM standard.
7. The tool should allow implementing anonymisation with unidirectional and bidirectional links with the personal data.
8. Traceability: it should be possible to keep a record of the software used in the de-identification process for audit purposes.

We also identified the following desirable features:

1. Open source. Reduces costs and allows institutions to adapt the tool to their own needs.
2. Limited software dependencies: the installation process must be straightforward.
3. Sensible default options to help support inexperienced users.
4. Flexibility: the privacy requirements for the different use cases can be quite diverse, so the tool should be easily configurable while providing enough options to handle most use cases.
5. Extensibility: projects may have unforeseen requirements, so the tool should provide an easy extension mechanism.

Data Sources

We used DICOM data produced by CT and MRI scanners from several manufacturers (General Electric, Phillips and Siemens) that included secondary captures as well as data that had been processed using PACS workstations to produce multi-planar reconstructions and perfusion parametric maps (Siemens, Carestream and Merge). Two of the CT and one of the MRI scanners were contained in a clinical setting and so no direct network connection to the central repository could be obtained, data from these sources was anonymised on a workstation prior to being transferred to the central repository on physical media. To test the networking features we deployed our solution in two university departments, in Aberdeen and Edinburgh. These sites are interconnected by SuperJANET5 [12], the United Kingdom high-speed education and research network.

For obvious reasons these data cannot be made public, so a test dataset (built with data obtained from the Biomedical Informatics Research Network (BIRN) Data Repository [13]) is available on PrivacyGuard’s project web site (http://forge.nesc.ac.uk/projects/privacyguard/) along with the corresponding outputs for different sample Privacy Policies.

PrivacyGuard development

The programming language selected for the development was Java because of its multiplatform nature and the availability of several mature Open Source DICOM libraries. However, to avoid a strong coupling with any of them PrivacyGuard includes an adapter interface to access and manipulate the DICOM objects. So far, we have implemented this interface for two popular libraries: pixelmed [14] and dcm4che2 [15].

One of the main design objectives was to make the toolkit flexible to enable the enforcement of different anonymisation strategies tailored to specific cases. To ensure that PrivacyGuard can be applied to many different situations the actual anonymisation procedure is specified in a Privacy Policy described in the next section. Using this mechanism it is possible to implement full anonymisation, linked and unlinked pseudonymisation as well as the DICOM standard anonymisation mechanism.

Finally, to ensure PrivacyGuard can be deployed in a number of different scenarios it can read DICOM data from either a filesystem or a DICOM receiver and the anonymous output can be written into a directory structure or transferred to a remote computer using SFTP or a DICOM push. Fig. 1 depicts the different input and output channels available.

Privacy Policies

A Privacy Policy (for PrivacyGuard) is an XML (eXtensible Markup Language) document that defines de-identification rules. These rules are implemented by Java classes that, to ensure authenticity, must be distributed in signed jar files (libraries). Several sample policies are included in the toolkit including one implementing the *Basic Profile*.

To avoid manual editing of the Privacy Policies, which is a tedious and error prone process, we developed a graphical application we called *Policy Editor* (Fig. 2 shows a screenshot). It also allows users to sign Privacy Policies using XML Signature [16], by doing this the policies recipients can verify the integrity of the policy contents.

DICOM header de-identification classes

In order to de-identify DICOM header attributes that contain personal information, they must be manipulated by either removing them if not required, emptying their content, substituting the information for dummy values or ambiguating their values. As part of the toolkit, we have implemented some sample de-identification classes for these operations. However, users can implement their own methods by providing classes that are loaded at run time. PrivacyGuard also will by default remove all private tags in the DICOM header, although this can be configured.

Burned-in annotations

Personal information may exist in the pixel data contained in DICOM objects. *PrivacyGuard* default action is to skip objects with the "burned-in" flag or identified as secondary captures. Alternatively, it is possible to clean the pixels by executing a user-defined class. The toolkit includes sample classes that implement straightforward methods like blacking out predefined zones of the image or pixels with values about certain threshold. Again, users can implement their own methods providing the potential to use more sophisticated approaches, for instance the method presented in [17] for detecting the textual information present in the pixels using wavelets.

Traceability

As stated before the classes used for the anonymisation operations must be provided in signed jar files. This not only allows their authenticity to be checked, but also allows the user to keep track of the anonymisation operations. *PrivacyGuard* can store traceability information in two non-exclusive ways:

- Along with the data by adding the information to the DICOM header. When anonymising a DICOM object information about the anonymisation process is added to the De-Identification Method (0012,0063) attribute.
- Externally: currently PrivacyGuard supports the storage of this information in log files. Different levels of details can be selected.

Applications

PrivacyGuard includes two applications: one for anonymising DICOM files contained in a folder, and a DICOM receiver-anonymiser (that can run as both a Windows service and a Linux daemon). Both are highly configurable and allow the execution of other operations in addition to the anonymisation such as archiving DICOM objects in the local file system or transporting them to remote computers using SFTP or a DICOM 'push'.

Multisite Deployment Support

A simple way of using a common Privacy Policy between centres is to publish it on a web server and configure *PrivacyGuard* on all sites to read the Privacy Policy from that web server. Furthermore, *PrivacyGuard* can also use directly signed jar files deployed on a web server. This centralised deployment not only helps to maintain coherence, but also eases the update procedures for policies and software. Fig. 3 shows a sketch of the lifecycle of a Privacy Policy from its creation using *Policy Editor* to its use by *PrivacyGuard*.

Results

Features Tests

PrivacyGuard has been able to anonymise DICOM objects produced by different manufacturers' equipment successfully. It correctly de-identified DICOM data produced by General Electric, Phillips and Siemens CT and MRI scanners, as well as DICOM objects produced by post processing. The tests also included using the two DICOM libraries (pixelmed and dcm4che2) for which we have implemented our interface. Other features tested included:

1. Testing of *PrivacyGuard* in Windows XP and several Linux distributions (RedHat, Scientific Linux and Ubuntu).
2. Testing the ability to detect secondary captures and black out the predefined zones of the images.
3. Finding and anonymising DICOM files contained under a given folder.
4. Receiving and anonymising DICOM objects sent using DICOM 'push' by dcm4che2 dcmsnd utility.
5. Transferring anonymised data using SFTP and DICOM to a remote computer.
6. Testing of the DICOM encryption mechanism. It was possible to anonymise the personal data attributes and store them encrypted inside the DICOM object, and recover later the information. A dataset from Grassroots DICOM library was used for this test.
7. We checked the correct inclusion of tracking information in the DICOM header and the generation of audit logs.

All these tests were successful, and we checked that the deployment of *PrivacyGuard* provides a convenient and intuitive method for anonymising data that is easily integrated into existing workflows.

Multicentre Tests

PrivacyGuard has been successfully deployed in a distributed environment: data were collected at Aberdeen and Edinburgh and stored locally. Then using *PrivacyGuard* the data were anonymised and transferred (using SFTP) to the processing site located in Edinburgh. Fig. 4 shows the data flow in the multicentre test. The flexibility of *PrivacyGuard* allowed a different range of subject pseudo-identifiers for each of the sites so that the contributing centres could be easily identified.

Discussion

We reviewed several existing DICOM de-identification tools and found out that none of them satisfied the criteria we had established. Only one of them, the otherwise very limited anonymisation utility of the Grassroot DICOM library, implements the DICOM standard attribute level confidentiality mechanism.

PrivacyGuard is a DICOM de-identification toolkit that facilitates the enforcement of privacy policies with no restrictions on the model chosen. For instance, it supports the DICOM standard confidentiality mechanism, but it does not impose its usage. It can also deal with burned-in annotations and its extension mechanism allows users to adapt it to their needs easily without modifying the library. At the same time, it comes with classes that support the most common use cases; this provides a low adoption barrier to new users.

PrivacyGuard is implemented in Java so it can run on most modern platforms. It is available under an Open Source licence and can be downloaded from http://forge.nesc.ac.uk/projects/privacyguard/. One of the advantages of *PrivacyGuard* is the possibility of making decisions to change the de-identification strategy in the configuration and Privacy Policy files. In this way, changes in regulations and de-identification rules can be enforced without any need to change the code. This along with its ability to read Privacy Policies and jar files from a centralised site, eases the deployment of the solution in multicentre projects.

Its flexibility along with the support for the encryption based DICOM anonymisation mechanism allows, for instance, to provide an alternative to the total removal of private attributes. These attributes can potentially contain personal data so they must not appear in the header. However, some manufacturers' software makes use of them; to support that use case *PrivacyGuard* can encrypt their content into the Encrypted Attributes Sequence so they can be recovered when necessary.

Finally, we have incorporated traceability features; in particular, *PrivacyGuard* can insert into the DICOM header information about the processing done and generates logs with information on the data processed and the policies used for that processing.

Acknowledgements

This work was funded by the SINAPSE collaboration (www.sinapse.ac.uk), a Pooling Initiative funded by the Scottish Funding Council and the Chief Scientist Office of the Scottish Executive. It was undertaken at the SFC Brain Imaging Research Centre, Division of Clinical Neurosciences, University of Edinburgh (www.sbirc.ed.ac.uk) and the National e-Science Centre, School of Informatics, University of Edinburgh (research.nesc.ac.uk), both members of SINAPSE. Two of the authors, D. Rodríguez González and J. Wardlaw, are funded by the SINAPSE collaboration, while J.I. van Hemert and T. Carpenter are also members of the collaboration. T. Carpenter is funded by the Cohen Charitable Trust and the Wellcome Trust IT grant 077393/Z/05/Z.

Data used for this study were downloaded from the Biomedical Informatics Research Network (BIRN) Data Repository (http://www.nbirn.net/bdr), supported by grants to the BIRN Coordinating Center (U24-RR019701), Function BIRN (U24-RR021992), Morphometry BIRN (U24-RR021382), and Mouse BIRN (U24-RR021760) Testbeds funded by the National Center for Research Resources at the National Institutes of Health, U.S.A.

References

[1] Data Protection Act 1998. The Stationery Office Limited London 1998, http://www.opsi.gov.uk/acts/acts1998/19980029.htm (accessed 20 May 2009).
[2] Medical Research Council 2000. Personal Information in Medical Research. http://www.mrc.ac.uk/Utilities/Documentrecord/index.htm?d=MRC002452 (accessed 20 May 2009)
[3] Chen J et al (2007) Observer success rates for identification of 3D surface reconstructed facial images and implications for patient privacy and security. Proc SPIE Int Soc Opt Eng 8:65161B-1–65161B-8
[4] The Digital Imaging and Communications in Medicine (DICOM) Standard. NEMA 20008. ftp://medical.nema.org/medical/dicom/2008/ (accessed 20 May 2009)

[5] Santesof Sante DICOM Editor Integrated Anonymizer: http://www.santesoft.com/howto/anonymize.html (accessed 25 May 2009)
[6] CTP-The RSNA Clinical Trial Processor in the MIRC Wiki: http://mircwiki.rsna.org/index.php? title=CTP-The_RSNA_Clinical_Trial_Processor (accessed 25 May 2009)
[7] IBM Haifa Labs. Universal De-Identification Platform. http://www.haifa.ibm.com/projects/software/udip/index.html (accessed 7 Apr 2009).
[8] Grassroots DICOM library. http://gdcm.sourceforge.net/ (accessed 30 June 2009).
[9] The SINAPSE project: http://www.sinapse.ac.uk (accessed 25 May 2009)
[10] Ellaway R et al. Clinical Recordings for Academic Non-Clinical Settings--Cherri Project Report. 1 March 2006. http://www.jisc.ac.uk/media/documents/programmes/digitalrepositories/clinicalrecordingreport.pdf (accessed 14 September 2009)
[11] Pianykh OS (2008) Digital Imaging and Communications in Medicine (DICOM): A Practical Introduction and Survival Guide. Springer, Heidelberg Berlin New York.
[12] Introduction to SuperJANET5: http://www.webarchive.ja.net/sj5/introductionsj5.html (accessed 5 August 2009)
[13] Biomedical Informatics Research Network (BIRN) Data Repository (http://www.nbirn.net/bdr) (accessed 16 September 2009).
[14] Pixelmed Publishing http://www.pixelmed.com/ (accessed 15 May 2009).
[15] dcm4che2 DICOM Toolkit http://www.dcm4che.org/confluence/display/d2/dcm4che2+DICOM+Toolkit (accessed 15 May 2009).
[16] W3C. XML Signature Syntax and Processing (Second Edition). W3c Recommendation 10 June 2008. http://www.w3.org/TR/xmldsig-core/ (accessed 31 July 2009).
[17] Wang JZ, Bilello M, Wiederhold G A Textual Information Detection and Elimination System for Secure Medical Image Distribution. J Am Med Inf Assoc 1997 [symposium suppl]:896

Fig. 1 High-level overview of the data flow, including different data input and output options

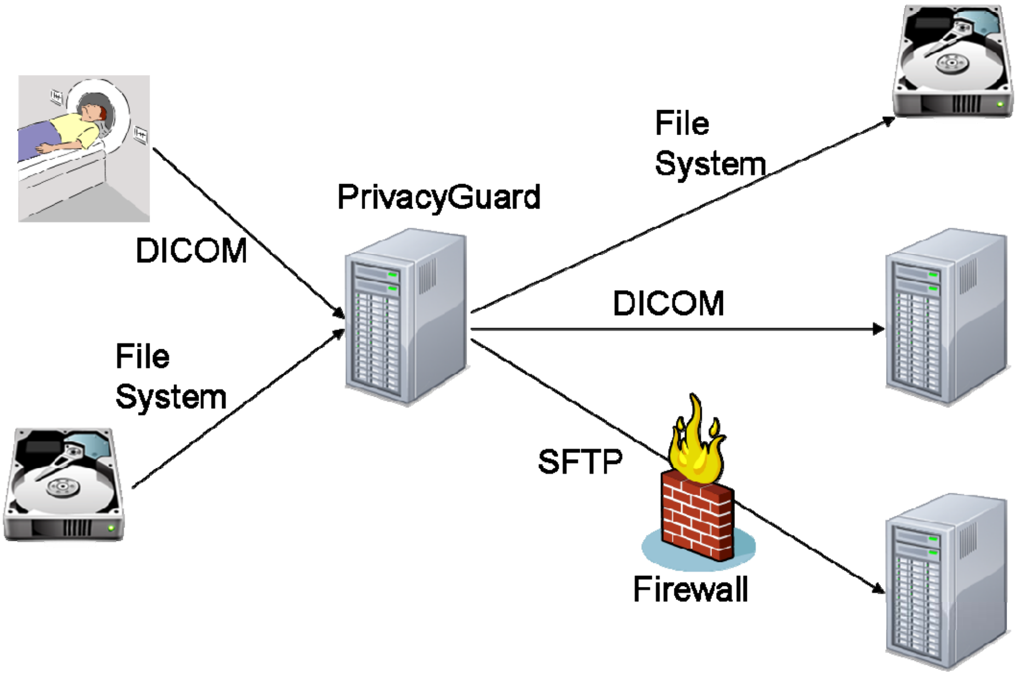
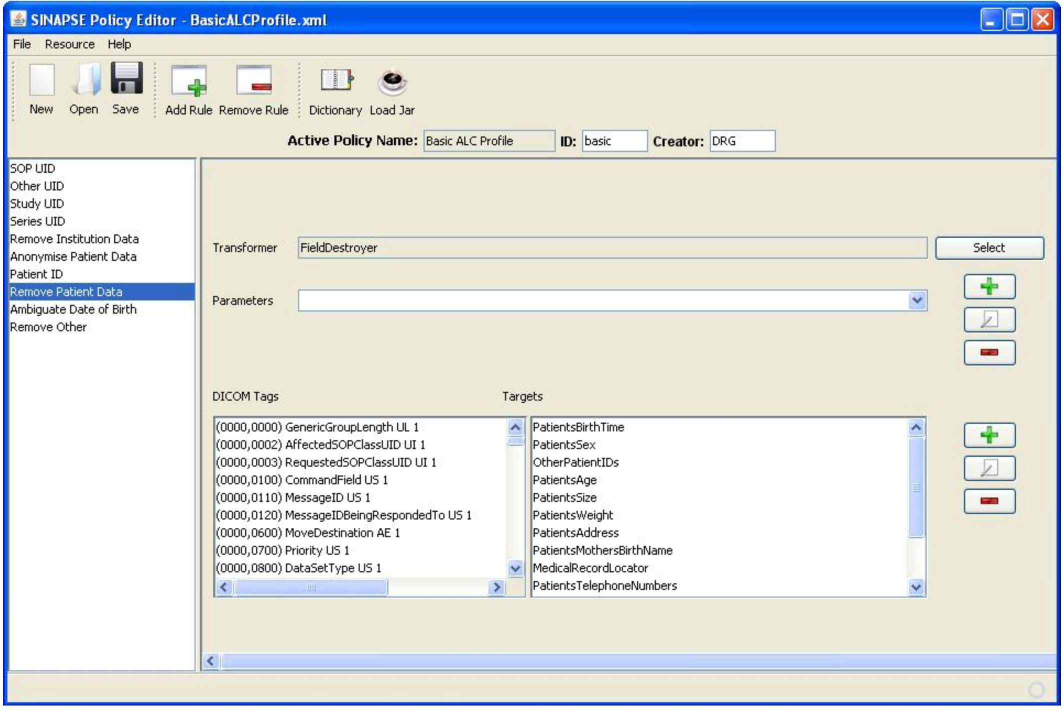


Fig. 2 Screenshot of the Policy Editor application



Active Policy Name: Basic ALC Profile **ID:** basic **Creator:** DRG

SOP UID
Other UID
Study UID
Series UID
Remove Institution Data
Anonymise Patient Data
Patient ID
Remove Patient Data
Ambiguate Date of Birth
Remove Other

Transformer FieldDestroyer

DICOM Tags

(0000,0000) GenericGroupLength UL 1
(0000,0002) AffectedSOPClassUID UI 1
(0000,0003) RequestedSOPClassUID UI 1
(0000,0100) CommandField US 1
(0000,0110) MessageID US 1
(0000,0120) MessageIDBeingRespondedTo US 1
(0000,0600) MoveDestination AE 1
(0000,0700) Priority US 1
(0000,0800) DataSetType US 1

Targets

PatientsBirthTime
PatientsSex
OtherPatientIDs
PatientsAge
PatientsSize
PatientsWeight
PatientsAddress
PatientsMothersBirthName
MedicalRecordLocator
PatientsTelephoneNumbers

Fig. 3 Lifecycyle of a Privacy Policy: 1) creation using PolicyEditor, 2) upload to a Web Server for sharing, 3) PrivacyGuard reads the policy from the web server and 4) applies it to the DICOM data

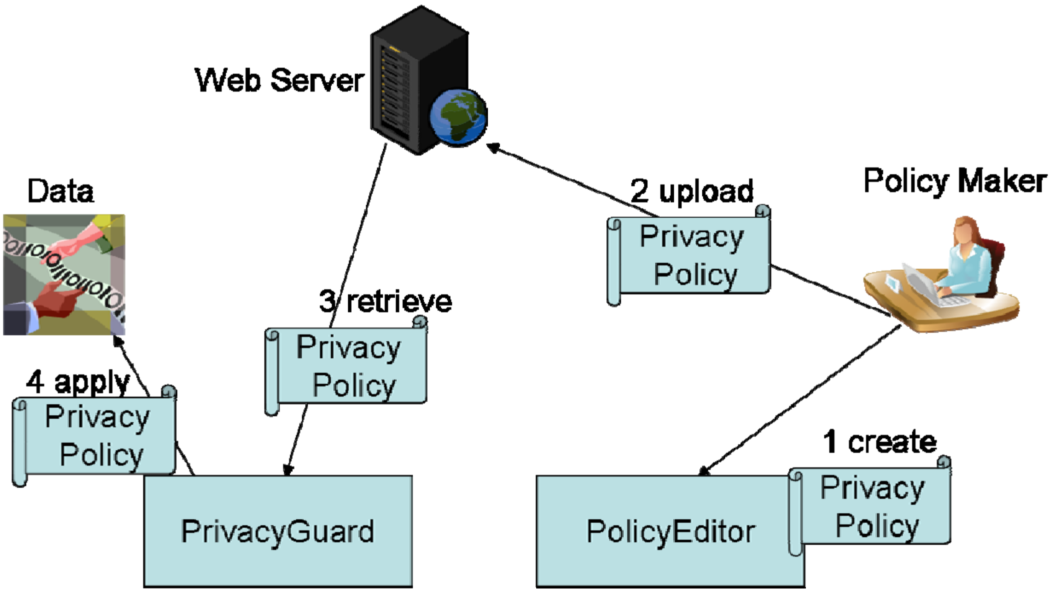
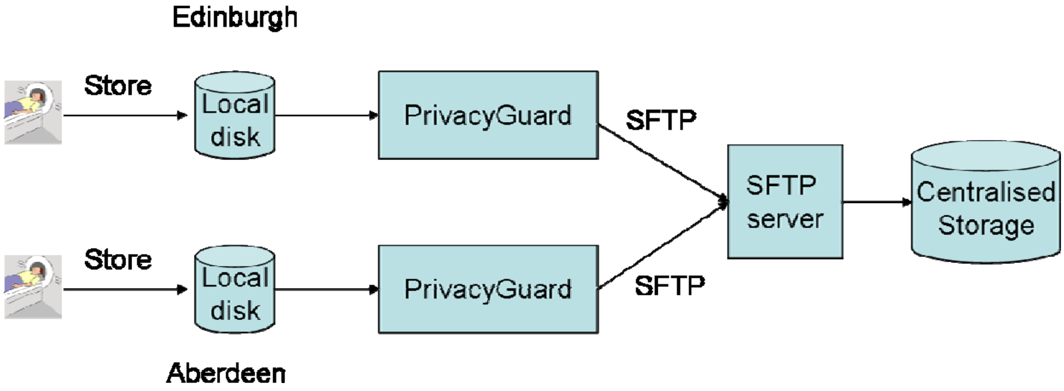


Fig. 4 Data flow in the multicentre test


Edinburgh
Store
Local disk
PrivacyGuard
SFTP
SFTP server
Centralised Storage
Store
Local disk
PrivacyGuard
SFTP
Aberdeen