Open and Trusted Information Systems/Health Informatics Access Control (OTHIS/HIAC)

Vicky Liu, Luis Franco, William Caelli, Lauren May and Tony Sahama
Faculty of Information Technology and Information Security Institute
Queensland University of Technology
PO Box 2434, Brisbane 4001, Queensland Australia

{v.liu, luis.franco, w.caelli, l.may, t.sahama}@qut.edu.au

Abstract

Information and Communications Technologies globally are moving towards Service Oriented Architectures and Web Services. The healthcare environment is rapidly moving to the use of Service Oriented Architecture/Web Services systems interconnected via this global open Internet. Such moves present major challenges where these structures are not based on highly trusted operating systems. This paper argues the need of a radical re-think of access control in the contemporary healthcare environment in light of modern information system structures, legislative and regulatory requirements, and security operation demands in Health Information Systems. This paper proposes the Open and Trusted Health Information Systems (OTHIS), a viable solution including override capability to the provision of appropriate levels of secure access control for the protection of sensitive health data.

Keywords: access control, architecture of health information systems, security for health information systems, health informatics, information assurance, trusted system, open solutions.

1 Introduction

Social, political and legal imperatives are emerging worldwide for the enhancement of the privacy and security of health information systems (HIS). A high level of "information assurance" is now seen as the necessary baseline for the establishment and maintenance of both future and current HIS. A security violation in HIS, such as an unauthorised disclosure or unauthorised alteration of individual health information, has the potential for disaster among healthcare providers and consumers.

Indeed, such emerging legal obligations as "breach notification", whereby custodians of private data are legally compelled to divulge any real or suspected breach in privacy to possible victims, are gaining international attention. This has been recently referenced by the USA legal firm, Steptoe & Johnson LLP (Steptoe & Johnson LLP 2008) , in the following terms:

> *New data protection requirements are being considered all over, including in Australia, Mexico, Turkey, South Korea, Peru, and Vietnam.*

Although the concept of Electronic Health Records has much potential for improving the processing of health data, electronic health records may also pose new threats for compromising sensitive personal health data if not designed and managed effectively. Indeed malevolent motivations could feasibly disclose confidential personal health information on a more massive scale and at a higher speed than possible with traditional paper-based medical records. There is also the factor of the healthcare service providers' willingness to accept and adopt a new technology that does not always facilitate efficient working practices. To encourage healthcare service consumers and providers to use electronic health records, it is crucial to instil confidence that the electronic health information is well protected and that consumers' privacy is assured. Indeed, unlike other industries and enterprises such as the banking and finance sectors, loss and disclosure of health record data is normally not recoverable. Again unlike the banking sector, a new "account" cannot be created along with all other necessary identification and authentication data and processes. Health data is usually "locked" to an individual.

However, it can be argued that concepts of privacy, with resulting requirements placed on data holders to maintain associated confidentiality, have rapidly changed in part due to the widespread acceptance of Internet based "social networking" and the very low cost of Terabyte level data storage facilities. Indeed Dyson (2008) has proposed that, in an era of "*Facebook*", "*Flickr*" and associated systems and services for "free" data sharing, the concept of individual privacy may be rapidly changing. This change involves a move from closely guarding the confidentiality of personal data records to one of personal control over access to that data.

Dyson states that "…people are learning to exert some control over which of their data others can see...". Dyson continues to point out that such control over access must become more dynamic and even allow for certain levels of ambiguity in just how such access patterns may be defined and managed by individuals, particularly as this relates to health records. Moreover, such access control structures have to be "user friendly", allowing those non-expert in aspects of information and data communications technology to understand and administer associated computer based systems.

1.1 Security Requirements for E-health

Achieving the usual security goals, normally applied through confidentiality, integrity and availability constraints, for HIS is an essential requirement and not just a technology feature. Privacy concerns take on new importance in this environment and may, in some cases, modify aspects of the usual confidentiality-integrity-availability trilogy. At the same time, emergency override requirements may involve more complex definition and implementation of confidentiality schemes. This can involve further parameters of time and location, identity and authentication when accessing healthcare, law enforcement or allied professionals, etc. Security techniques are a critical factor in the successful implementation of e-health initiatives. Several countries such as Australia, the United Kingdom (UK) and the United States of America (USA) are actively involved in the development of national e-health initiatives. These designs rely upon a basic set of security requirements to implement their e-health initiatives.

The USA government intends to reform its national healthcare system with the goal of improving the effectiveness and efficiency of healthcare operations whilst assuring that sensitive health information remains private and secure through their 1996 Health Insurance Portability and Accountability Act (HIPAA). The purpose of HIPAA provisions is to encourage electronic transactions whilst simultaneously requiring appropriate security measures for protection of the individually identifiable health information.

Australia's National E-health Transition Authority[1] (NEHTA) clearly defines similar security goals in its mission statements. They emphasise the importance of creating a complete, usable and implementable security architecture for HIS. NEHTA also recognises that privacy perceptions of the Australian community play a major role in ensuring the success of e-health systems.

In the case of the UK, the National Health Service (NHS) also clearly affirms the principles of information security[2] to require that all reasonable safeguards are in place to prevent inappropriate access, unauthorised modification or manipulation of sensitive patient record information.

Section 2 discusses the related work undertaken by the Australian national e-health body, National E-health Transition Authority (NEHTA). The Open and Trusted Health Information Systems (OTHIS) structure is our approach to providing a viable e-health system with the potential for implementing sustainable security measures. OTHIS, outlined in Section 3, has the capacity to protect the privacy and security of health information under an overall trusted health informatics scheme. This paper focuses on one of the OTHIS modules, Health Informatics Access Control (HIAC), in Section 4. An analysis of HIAC is presented in Section 5. Finally, conclusions are drawn and future directions for OTHIS are discussed in Section 6.

2 Related Work

An analysis of common existing approaches to secure health information systems in Australia, UK and the USA is given by Liu, Caelli, May and Croll (2007). In this paper which addresses sustainability of HIS systems, three scenarios related to information privacy violations and weaknesses are identified and discussed. As we are concerned with e-health infrastructures that satisfy the Australian environment, Section 2.1 discusses the Australian direction on this given by NEHTA. Section 2.2 discusses the NEHTA approach from the authors' perspectives.

2.1 National E-health Transition Authority

NEHTA recommends using a Service Oriented Architecture approach to the design of healthcare application systems. "Web Services" technology standards provide the capacity for implementing secure messaging systems (NEHTA 2005). NEHTA argues that the continued development of information systems around Web Services technology is leading the way for the information and communications technology industry into its realisation of this Service Oriented Architecture approach. Web Services are also accepted as best practice for the design of scalable distributed systems

[1] NEHTA was established by Australia's Federal Government in 2005 to oversee the introduction of a system of national electronic health records. Its statement of mission is available at http://www.nehta.gov.au/ accessed 12/07/2008.

[2] The principles of information security from UK NHS are available http://www.connectingforhealth.nhs.uk/systemsandservices/infogov/security accessed 12/07/2008.

today. The Service Oriented Architecture approach is claimed to lead to more reusable, adaptable and extensible systems over other techniques. In particular, NEHTA supports the concept that Web Services technology has gained notable attention within the information and communications technology industry. Its use is extending in both popularity and market penetration.

NEHTA work programs for an e-health interoperability framework include Clinical Information, Medicine Product Directory, Supply Chain Efficiency, e-Health Policy, Clinical Terminologies, Individual Healthcare Identifiers, Healthcare Provider Identifiers, Secure Messaging, User Authentication and Shared Electronic Health Record Specifications.

2.2 Discussion on NEHTA Approach

NEHTA focuses on exchanging clinical information by electronic means securely and reliably. This may be achievable at the data communications link level by using secure messaging technology. The fact is however that the associated and critical health information computer systems will be openly connected to the Internet, and thus be exposed to "cyber-attacks". This exposure has not been prevalent before. In this Internet connectivity environment, the issues of data "at rest" and "under processing" within a specific operating system are far more critical, as is evidenced by any cursory examination of illicit penetration of computer systems connected to the Internet globally. A complete architecture is needed, therefore, and not one that involves just a secure messaging system alone. OTHIS addresses the privacy protection and security for health systems in a holistic and "end-to-end" manner. The OTHIS architecture is designed to complement existing work already evident in related HIS security areas.

3 Our Approach – Open and Trusted Health Information Systems (OTHIS)

Security may be implemented at the level of the health services applications system. Even if security is established within that health service system, however, the overall system can be no more secure than the operating system upon which the applications depend. The operating system itself can be no more secure than the hardware facilities of the computer on which the operating system performs. Likewise, any other software component set at the higher levels is totally dependent upon the security functions provided at the lower levels. Examples of such software include "middleware", database management systems, the network interface structure, and the "stack". The lowest level software is the operating system which provides the foundational security for the higher levels. The operating system also needs a degree of "robustness" against possible attacks at its level.

Necessary healthcare security services such as authentication, authorisation, data privacy and data integrity can only be confidently assured when the operating system is trusted. Thus "trusted operating systems" provide the foundation for any security and privacy schemes. Such strong security platforms may be considered as necessary to ensure the protection of electronic health information from both internal and external threats as well as providing conformance of health information systems to regulatory and legal requirements Loscocco, Smalley, Muckelbauer, Taylor, Turner and Farrell (1998) have stated that the underlying operating system should be responsible for protecting the "application-space" against tampering, bypassing and spoofing attacks. They address the significance of secure operating systems as follows:

> *"The threats posed by the modern computing environment cannot be addressed without support from secure operating systems and any security effort which ignores this fact can only result in a "fortress built upon sand."*

It is an inherently insecure exercise to attempt to build an application requiring high levels of trust in the maintenance of security and privacy when the underlying structure within a computer system is a non-trusted operating system. Simply put, the trusted application relies totally upon the non-trusted operating system to access low level services.

Our approach caters for the trusted operating system with the capacity to provide a viable and sustainable solution for the protection of sensitive health data in the healthcare environment. The authors define the characteristic features of OTHIS as:

- OTHIS is an holistic approach to HIS consistent with health legal requirements,
- OTHIS is an open architecture,
- the OTHIS scheme builds on the top of trusted firmware and hardware bases, and
- OTHIS is modularised architecture.

3.1 Holistic Approach to HIS

In achieving a high level of information assurance in HIS, we propose an holistic approach to a more trusted scheme, the Open and Trusted Health Information Systems (OTHIS). The goal of OTHIS is to address privacy and security requirements at each level within a modern HIS architecture to ensure the protection of data from both internal and external threats. OTHIS has the capacity to ensure legal compliance of any HIS to appropriate legislative and regulatory requirements. The primary emphasis in this paper is on the Australian health sector.

3.2 Open Architecture

OTHIS takes an open approach that can provide cost effective, viable and sustainable architecture to security and privacy in HIS. OTHIS embraces emerging open architecture, standard and solution technologies rather than use proprietary technologies. The inclusion of “open” in the OTHIS framework is to allow our proposed architecture to be available for public access and to provide a platform for interoperability. This approach is also supported by Goldstein Groen, Ponkshe and Wine (2007). Open systems allow disparate HIS to communicate and exchange clinical information in an open network environment. Normally HIS are based around open and distributed network systems; therefore, it is entirely appropriate to relate OTHIS to international standards such as Open Systems Interconnection (OSI) security architecture through standards ISO 7498-2 and ISO/IEO7498-4. This research adopts the broad architectural concepts as proposed in those standards and as adopted for some time by national governments via “Government OSI Profiles”.

3.3 Trusted Platform

OTHIS also involves the term “trust”, relating to “trusted system”. Any information system depends, fundamentally, upon a trusted base for safe and reliable operation, commonly referred to as a “trusted computing-base”. Without a trusted computing-base any system is subject to compromise. In particular, data security at the application level can be assured only when the healthcare application is operating on the top of the trusted computing-base platform. Otherwise the adversary can exploit illicit means to perform the actions that bypass or disable the security features of healthcare applications or that grant inappropriate access privileges. Inevitably healthcare applications or databases must be executed atop the trusted platform in order to achieve adequate information assurance. For this reason OTHIS aims at running on the top of trusted firmware and hardware bases. This trusted firmware and hardware base is commonly referred to as a Trusted Platform Module. This research assumes a commodity Trusted Platform Module upon which to deploy OTHIS. Many such modules are available in the marketplace.

3.4 Modularised Architecture

Appropriate data security management involves the protection of such data in storage, during processing and transmission. The proposed OTHIS structure (Figure 1) addresses all these areas and consists of three of distinct modules:

- Health Informatics Access Control (HIAC),
- Health Informatics Application Security (HIAS), and
- Health Informatics Network Security (HINS).

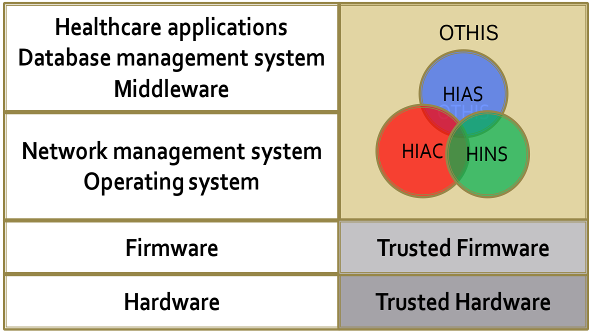

Figure 1: Open and Trusted Health Information Systems

OTHIS is a modularised architecture for HIS. It can be clearly divided into separate and achievable function-based modules. The advantages of the modularisation include the fact that each module is easier to manage and maintain. One module can be changed without affecting the other module. OTHIS is, thus, a broad architecture covering those requirements and parts that may be selected as required to meet particular circumstances. There is some overlap with these three modules, however, each module has a specific focus area. HIAC is data centric dealing with information at rest, HIAS is process centric dealing with information under processing, and HINS is transfer centric dealing with information under transfer. Trust in network operations through HINS rests completely upon trust in HIAS and HIAC; otherwise the security of messaging becomes futile. The focus of this paper is on the HIAC model.

4 Health Informatics Access Control (HIAC)

Access control mechanisms are used to define and then restrict users’ access to resources. Organisations would normally use these controls to grant employees, for example, the authority to access only the information those users need to perform their duties, i.e. the principle of “least privilege”. Access controls can limit the activities that an employee can perform on data at the level of granularity desired. Access control mechanisms are therefore enabled at the operating system level as well as higher levels including data network management and the database management systems for the application.

Access control is one of the fundamental security mechanisms used to protect computer resources, in particular in multi-user and resource-sharing computer based environments such as those incorporated into a contemporary HIS. The lack of adequate access control and associated system management in health relevant computer systems has been demonstrated on numerous occasions in recent history, including the privacy invasion situation at Australia's Centrelink (Sharanahan and Karvelas 2006), the lack of adequate safeguards in the UK NHS patient records system (Leigh and Evans 2006), and the significant information technology security weaknesses identified in the US HHS information system (GAO 2006). These types of information privacy violations or weaknesses have the potential for inflicting, and do inflict, major harm on HIS consumers and providers alike. The issue of providing suitable computer operating system access control in such systems is not an insurmountable one. Indeed, appropriate computer-based access control schemes do exist and can be deployed to address these information security issues.

4.1 Access Control Models

Discretionary access control essentially assigns responsibility for all security parameters to the "owners" (users) of such larger entities, usually their creator, who could pass on such parameters to others and perform functions as desired. Role-based Access Control refines the concept to allow users to be grouped into defined functions or "roles" allowing for far easier management of overall system security policy particularly in dynamic business environments. Mandatory access control (MAC), in principle, enforces security policy as set out by the overall enterprise and not set up by definitions provided by file/program "owners". The traditional MAC policy was originally designed for a military environment based on the multi-level security policy hierarchical structure and was quite rigid in its application. More recent research has modernised the traditional MAC approach to a flexible form of MAC (Flexible MAC) that overcomes traditional MAC limitations with the enforcement of a wider range of security requirements including confidentiality, integrity, least privilege and separation of duty.

4.2 HIAC is Flexible MAC-based Architecture

HIAC is a Flexible MAC-based model accompanied by Role-based Access Control properties to simplify authorisation management. This degree of simultaneous control, flexibility and a refined level of granularity is not achievable with Discretionary Access Control, Role-based Access Control or MAC individually. HIAC proposes a viable solution to providing appropriate levels of secure access control for the protection of sensitive health data. Increasingly, HIS are being developed and deployed based upon commercial, commodity-level information and communications technology products and systems. Such general-purpose systems have been created over the last 25 years with often only minimal security functionality and verification. In particular access control, a vital security function in any operating system that forms the basis for application packages, has been founded upon earlier designs based on Discretionary Access Control. Discretionary Access Control systems were defined around an environment where data and program resources were developed and deployed within a single enterprise, assuming implicit trust amongst users. This environmental model is no longer valid for modern HIS. In some commercial systems, for example, even the addition of a simple single printer unit has the capacity to seriously undermine the overall integrity of the information system.

4.3 HIAC Platform

Currently available products that support the MAC principles of operating systems include:

- "Red Hat Enterprise Linux (RHEL) Version 5 and "Fedora Core 9",
- "Sun Microsystems Solaris 10 with Trusted Extensions Software",
- "Novell SUSE Linux Application Armor (AppArmor)", and
- "FreeBSD 5.0".

The HIAC model exploits the security-enhancement features of such trusted operating system in the healthcare environment. The end result is a dedicated trusted HIS which satisfies all security requirements. To determine the practical viability of a HIAC model for HIS a demonstrator was built on the Security Enhanced Linux (SELinux) operating system by Henricksen, Caelli and Croll. (2007) with RHEL Version 4. This was later modernized by Franco Martin (2008) with Fedora Core 9.

4.4 Flask Architecture – Flexible MAC – SELinux

The U.S. National Security Agency designed and engineered SELinux with a security architecture named the Flux Advanced Security Kernel (Flask). It aims to set an example of how Flexible MAC could be added to a mainstream operating system to greatly improve the security of the system. Flexible MAC provides a balance of security needs and flexibility of implementation that allows the security policy to be modified, customised and extended as required in line with normal application and system requirements. SELinux also provides separation of security domains as a fail-safe feature to enable the confinement of damage caused by the probability of malicious or flawed code execution (Loscocco and Smalley 2001). The flexibility of SELinux includes the separation of the security policy logic from the enforcement mechanism. This enables the independent policy module to be modified and extended as required without affecting the rest of the kernel or the need to restart the system.

4.5 Protection and Enforcement Using SELinux Policy and Profile in HIAC

In general, the organisational security policies are defined by CEO/CIO. Access privileges are determined by the data custodians. The system administrator configures and deploys the organisational access policy defined and determined by the CEO/CIO and the data custodian. The following sections describe the procedures of developing a security policy and using SELinux security mechanisms to protect sensitive health information for HIS.

To use SELinux Policy to implement the organisational access policy, one must understand the SELinux Policy mechanisms. SELinux Policy is a collection of rules that determine allowed access for a system created in accordance with the corporate security policy. An SELinux Policy consists of a set of SELinux Profiles (policy modules) that define the associated security properties controlling the security behaviour of the system. The following procedure steps show the development of an SELinux policy (Figure 2):

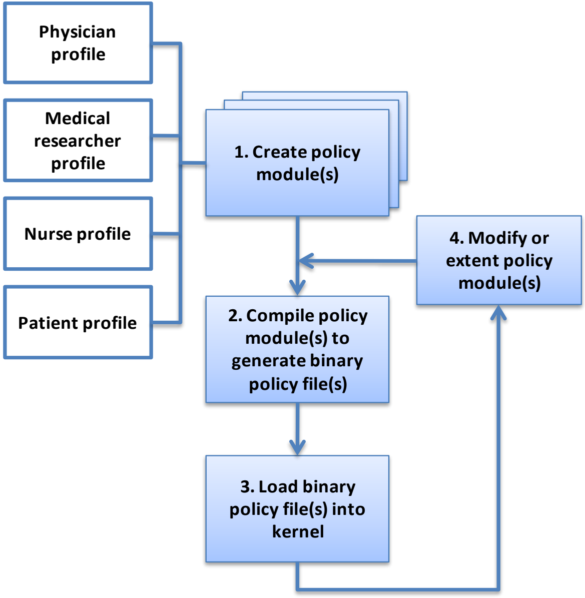
Physician profile
Medical researcher profile
Nurse profile
Patient profile
1. Create policy module(s)
2. Compile policy module(s) to generate binary policy file(s)
3. Load binary policy file(s) into kernel
4. Modify or extent policy module(s)

Figure 2: SELinux Profile Development Cycle

1. Create policy module(s), such as for physicians, medical researchers, nurses and patients policy modules.
2. Compile the policy module(s) to generate the binary policy file(s) as a loadable kernel module(s).
3. Load the binary policy file(s) into the running kernel for access enforcement.
4. If policy module(s) require(s) changes, the modified policy module(s) is (are) recompiled and then reloaded into the running kernel.

4.6 SELinux Concepts – User Identifier, Role and Type Identifier

The SELinux Policy is now configured and loaded into the kernel ready for operation. Figure 3 shows the authorisation process flow in SELinux.

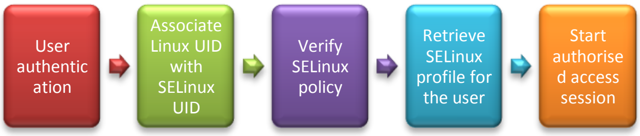
User authentication
Associate Linux UID with SELinux UID
Verify SELinux policy
Retrieve SELinux profile for the user
Start authorised access session

Figure 3: Authorisation Process Flow in SELinux

After a user is authenticated to the SELinux system, the user logs into the system with his/her username which is associated with a Linux unique user identifier (UID). A Linux UID is generated when a user account is created (Table 1). A user may have more than one user account. In SELinux, the system administrator maps the Linux UID(s) of the user to an SELinux UID, so that any action performed within the system by the same user can be traced for accountability. In addition, having different user identifiers helps to keep Linux Discretionary Access Control mechanisms separated from the SELinux MAC mechanisms.

The user access privileges, which are user, role, domains and types associated with SELinux UID, are defined in the SELinux Policy. The system verifies the SELinux Policy to retrieve access privileges which define the SELinux Profile of the user. The authorised access can now begin from this point.

SELinux Profile	Linux UID	SELinux UID	Role	Authorised Domain
Physician	drpaul (501)	hc_doc_u	hc_doc_r	hc_doc_diag_t
Medical Researcher	resjohn (502)	hc_res_u	hc_res_r	hc_res_diag_t
Nurse	nuralice (503)	hc_nur_u	hc_nur_r	hc_nur_diag_t
Patient	patluis (504)	hc_pat_u	hc_pat_r	hc_pat_diag_t

Table 1: Linux UID, SELinux UID, Role and Type

4.7 SELinux Security Mechanisms to Protect Sensitive Health Data

In SELinux, type enforcement is the primary access control feature where access control is based on a security context. All subjects and objects have a type identifier associated with them. To access an object, the subject's type must be authorised for the object's type. Namely, TE makes access decisions based on the security context to determine access. A security context consists of three elements: user, role and type identifier.

With SELinux, users are assigned a set of roles which determine a set of processes authorised for the user's identity. Domains are used to specify how roles can interact with subjects and objects in the system. Different sets of domains are authorised for each of the user roles based on the TE rules defined in the SELinux policy.

SELinux allows dividing the system space into a set of "sandboxes" determined by the authorised user domains. An application running on behalf of a user is allowed to access certain resources in the system. To prevent unauthorised access, a medical related sandbox can be used to isolate a space in which a medical application is permitted to access medical records. The following clinical scenario is used to explain this concept.

It is assumed that a doctor "Paul" is associated with a physician role, which is allowed to run the Diagnostic Application within a specified domain "hc_doc_diag_t" and is allowed to access the files with type "hc_diag_file_t". In fact, when Paul activates the Diagnostic Application, the system process labelled with the domain "hc_doc_diag_t", is acting on behalf of Paul. It enters the domain "hc_diag_doc_t" with specified access permissions to the those objects and subject types associated with this domain only. Assume that user "John" is associated with a medical researcher role. Even if John is allowed to access the Diagnostic Application, John is accessing this application through the domain "hc_res_diag_t". This domain is authorised to access different resources than the physician. Therefore, even if they use the same application, in the same system, they cannot access the same resources.

SELinux Sandboxes can be constructed to protect medical data from a compromised application (Figure 4).

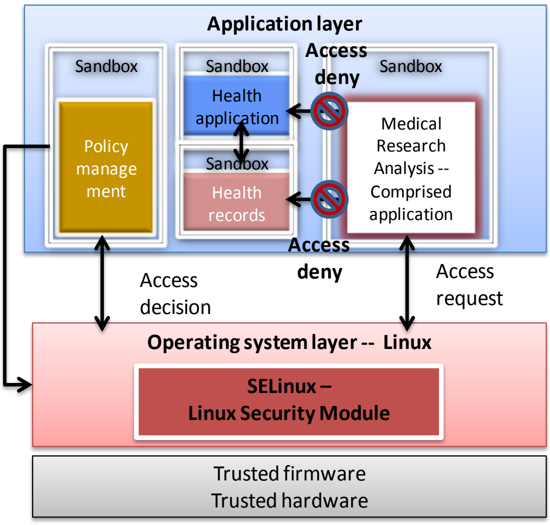

Figure 4: Protect Sensitive Health Data with SELinux

Medical researchers are authorised to access the medical information for secondary usage within another sandbox. If a "backdoor" is open for a hacker to gain access privileges over unauthorised resources, the hacker only has access to the medical information for secondary usage within that sandbox. The damage from this compromised application therefore can be contained within a single domain, not the entire system. In contrast, in a Discretionary Access Control based system, the damage from compromised applications cannot be restricted within a space. In particular, if the hacker gains the access privileges of a system administrator, the entire system is compromised including the sensitive individual health information.

4.8 Example of an SELinux Policy Module

This section provides an example of coding for SELinux Profile in relation to how a physician and a medical researcher can run the Diagnostic Application with different accesses to different types of health data files. The two users can be defined in SELinux: "hc_doc_u" and "hc_res_u". The code specifies the sandboxes to be accessed when the authorised physicians and medical researchers run the Diagnostic Application, that is "hc_doc_diag_t" for the authorised physicians and "hc_res_diag_t" for the authorised medical researchers. The domain "hc_res_diag_t" is defined to allowed access to the data files with type "hc_res_dbfile_t" (i.e. data files for the authorised researchers accesses). The domain "hc_doc_diag_t" is specified to access the data files with type "hc_pnt_dbfile_t" (i.e. sensitive health data files). In such a way, the medical researcher is not able to access sensitive data files with the unauthorised role. That is, the medical researcher can access only the domain "hc_diag_res_t" and data files associated with type "hc_res_dbfile_t".

```
# Type for the Diagnostic Application executable file
type hc_diag_sys_exec_t;
files_type(hc_diag_sys_exec_t)

# Type for the DB files which can be accessed by researchers.
Type hc_res_dbfile_t;
files_type(hc_res_dbfile_t)

# Type for the DB files which can be accessed by physicians.
# This type can be assigned to files containing sensitive information.
Type hc_pnt_dbfile_di_t;
files_type(hc_pnt_dbfile_di_t)

# This interface creates types, roles and domains to be assigned to physicians.
Healthcare_create_users(hc_doc)
# This interface creates types, roles and domains to be assigned to
#  researchers.
Healthcare_create_users(hc_res)

# These interfaces assign only the necessary privileges for the user to access
# their home directory.

# This interface authorises physicians to access the delegated sandbox while
# executing the Diagnostic Application.
Diag_general_domain(hc_doc)

# This authorises researchers to access the delegated sandbox during
# executing the Diagnostic Application.
Diag_general_domain(hc_res)

# The "diag_general_domain" is described in more detail further in this document.
# These references to this interface create two domains which constitute the
# sandboxes for physicians and researchers: hc_doc_diag_t and hc_res_diag_t.

# This line of code authorises physicians to create, write and read DB files
# with the type hc_pnt_dbfile_di_t while operating within the boundaries of the
# sandbox. The boundary of the sandbox is defined with the domain
# hc_doc_diag_t.
allow hc_doc_diag_t hc_pnt_dbfile_di_t:file { create_file_perms \
                              write_file_perms read_file_perms };

# This line of code authorises researchers to create DB files with the type
# hc_res_dbfile_di_t while operating within the boundary of the delegated
#sandbox.
# The boundaries of the sandbox are defined with the domain hc_res_diag_t.
allow hc_res_diag_t hc_res_dbfile_di_t:file { read_file_perms };

# The following 2 statements authorise the roles corresponding to physicians
# and researchers to access their corresponding domains.
role hc_doc_r types { hc_doc_diag_t };
role hc_res_r types { hc_res_diag_t };
```

An SELinux Policy is comprised of different components. These components can be placed in three different files: type enforcement, context file and interface files. In the above code, researchers and physicians are authorised to access their delegated specific sandboxes while running the Diagnostic Application. These privileges are granted through the use of the "diag_general_domain" interface which is shown below.

```
interface('diag_general_domain',`
  type $1_diag_t;
  domain_type($1_diag_t)

  domain_auto_trans($1_t, hc_diag_sys_exec_t, $1_diag_t)
  domain_entry_file($1_diag_t, hc_diag_sys_exec_t)

  allow $1_diag_t $1_t:process sigchld;
  allow $1_diag_t $1_tty_device_t:chr_file { rw_term_perms append };
  allow $1_diag_t $1_devpts_t:chr_file { rw_term_perms append };
}
```

5 Analysis

To meet real-world application security demands that are understandable, implementable and usable, our OTHIS research embraces reasonable security strategies against economic realities using open solution technologies such as SELinux rather than using proprietary technologies. In general, open source technologies are free to use, modify, and redistribute. Developers of open source software distribute their software freely and make profits from support contracts and customised development. It is an expensive exercise to use proprietary software in particular for large enterprises to upgrade software and increase its number of software licenses. The costs of using proprietary software involve the procurement of a software license and software upgrades. Open source technologies have gained significant attention in the marketplace. A Gartner report [3] predicts that more than 90 percent of enterprises will use open source in direct embedded ways by 2012. In particular, open source software is essential for large enterprises who seek to reduce their total cost of ownership and increase returns on investment. A common complaint related to open source is the lack of a reliable source of assistance when organisations encounter problems in open source software. One can resolve this through the subscription of service support from the open source developer.

It is essential to integrate security profiling structures in relation to other enterprise systems such as overall human resource management systems and the like. This allows for definition and deployment of security policies that represent legal, regulatory, policy and enterprise level requirements for reliable and consistent enforcement at the computer system level. The primary and well-known strengthen of SELinux is security, yet the level of complexity in policy configuration could be considered beyond the expertise level of many CIOs in health related organisations. Simplifying the level of complexity in SELinux configuration can be managed through the current distribution containing the SELinux Reference Policy. This is an example of a general purpose security policy configuration which can meet a number of security objectives and can be used as the basis for creating other policies. Additionally, there is a number of SELinux Policy generation and

[3] A ZDNet new article "A Gartner: Open source will quietly take over" is available at http://news.zdnet.co.uk/software/0,1000000121,39379900,00.htm accessed 29/08/2008.

management tools [4] available to simplify the development of SELinux Policy.

Currently, the Flask architecture with the Flexible MAC enforcement is a rapidly growing area gaining global attention since its introduction in SELinux. A recent press release[5] issued in 2008 announced that Flask will also be implemented in Sun Microsystems OpenSolar operating system to advance MAC. Thus, tools and techniques are constantly developed from the open source community to address the complex configuration challenges of SELinux. In fact, our HIAC demonstrator for HIS was built on RHEL version 4", which was carried out at the primitive stage of SELinux project development (Henricksen, Caelli and Croll 2007). It was argued that the previous SELinux policy facilities were too inflexible to handle a large scale of HIS which may involve dynamic and frequent changes to the security policies such as adding/deleting users and applications. With the earlier SELinux distribution, any changes and extensions made to the SELinux Policy would have needed the policy to be recompiled and the system to be restarted. As SELinux continues to advance and evolve, any changes to the security policies can be recompiled with available tools and techniques and then updated security polices reloaded into the system kernel without the need to restart the system. To date our HIAC demonstrator has been updated with Fedora Core 9 to confirm the flexibility of the current release of SELinux.

6 Conclusion and Future Work

Current trends are towards using Web Services as the technology to develop and implement healthcare application systems. Their focus is on security aspects in exchanging clinical information electronically at the application level. This is endorsed by NEHTA (2005). The moves towards Service Oriented Architecture/Web Services global systems present major challenges where such structures are not based on highly trusted operating systems. All applications and supporting software which necessarily reside atop the untrusted operating systems are also considered untrusted. Health information is highly sensitive by its nature. It is therefore critical to protect such information from security hazards and privacy threats.

The authors argue that using the non-MAC-based system to protect personal privacy and confidentiality of electronic health records is not sustainable. This is evidenced by a number of scenarios related to health information privacy violations or weaknesses which have recently been found in Australia, the UK and the USA (Liu, Caelli, May and Croll 2007). Our OTHIS/HIAC research argues that the need of a radical re-think is absolutely crucial in the understanding of access control in light of modern information system structures, legislative and regulatory requirements and security operational demands in HIS. This is affirmed by the Australian Government[6] calls for robust legislation to protect individual electronic health record systems. This security focus enhances the quality of healthcare service delivery with respect to privacy assurance and is a key element of the overall success of such a system.

Information and communications technologies are now sufficiently advanced that a MAC-based electronic healthcare management system is feasible. Our approach overcomes many of the security issues which have plagued previous attempts at electronic health management systems. The authors argue that adoption of appropriate security technologies, including in particular Flexible MAC-oriented operating system bases, can satisfy the requirements for the protection of sensitive health data.

Preliminary results of this research indicate that the broad philosophy of Flexible MAC appears ideally suited to the protection of the healthcare information systems environment. This study, therefore, contends that the approach to "hardening" electronic HIS is essential to build privacy- and security-aware applications that reside atop Flexible MAC-based operating systems. Such systems have the potential to meet all stakeholder requirements including modern information structures, organisational security policies, legislative and regulatory requirements for both healthcare providers' and healthcare consumers' expectations and demands in HIS.

To provide sustainable and trusted health information systems, one must take an holistic approach to address security requirements at all levels in HIS. The overall HIS architecture must evolve into a set of complementary security architectures which, at least, incorporates those

[4] Links to SELinux Policy generation tools are available at http://fedoraproject.org/wiki/SELinux/PolicyGenTools accessed 27/08/2008.

[5] A media release has been issue announcing the joint venture between the NSA and Sun Microsystems to advance MAC named "National Security Agency And Sun Microsystems Lead OpenSolaris Community Project To Advance Mandatory Access Controls" is available at http://www.sun.com/aboutsun/pr/2008-03/sunflash.20080313.1.xml accessed 27/08/2008.

[6] A press release has been issued entitled "E-health privacy blueprint - robust legislation is needed says Privacy Commissioner" is available at http://www.privacy.gov.au/news/media/2008_15.html accessed 27/08/2008

proposed under the OTHIS scheme consisting of HIAC, HIAS and HINS. This paper focuses on OTHIS/HIAC which proposes a viable solution to provide appropriate levels of secure access control for the protection of sensitive health data. Future research under OTHIS will continue to develop and test through experimental structures created on a Flexible MAC-based operating system. Key research questions to be answered include those issues of data "at rest" and "under processing" aspects of the proposed architecture OTHIS. This research will also elucidate the relationships between HIAS which relies completely upon trust in HIAC and HINS.

7 References

Dyson, E. (2008): Reflections on Privacy 2.0. Scientific American **299** (No.3 September 2008): 55-60.

GAO (2006): Information Security: Department of Health and Human Services Needs to Fully Implement Its Program. http://www.gao.gov/new.items/d06267.pdf. Accessed 12/05/2008.

Goldstein, D., Groen, P., Ponkshe, S. and Wine, M., Eds. (2007): Medical Informatics 20/20: Quality and Electronic Health Records through Collaboration, Open Solutions, and Innovation, Jones and Battlett Publishers, Inc.

Henricksen, M., Caelli, W. and Croll, P. (2007): Securing Grid Data Using Mandatory Access Controls. 5th Australian Symposium on Grid Computing and e-Research (AusGrid), Ballarat Australia.

Leigh, D. and Evans, R. (2006): Warning over privacy of 50m patient files. Guardian News and Media Limited.

Liu, V., Caelli, W., May, L. and Croll, P. (2007): A Sustainable Approach to Security and Privacy in Health Information Systems. 18th Australasian Conference on Information Systems (ACIS) Toowoomba, Australia.

Loscocco, P. and Smalley, S. (2001): Meeting Critical Security Objectives with Security-Enhanced Linux. Proceedings of the 2001 Ottawa Linux Symposium.

Loscocco, P., Smalley, S., Muckelbauer, P. A., Taylor, R. C., Turner, S. J. and Farrell, J. F. (1998): The Inevitability of Failure: the Flawed Assumption of Security in Modern Computing Environments. Proceedings of the 21st National Information Systems Security Conference, 303-314.

Martin Franco, L. (2008): SELinux Policy Management Framework for HIS (under examination) Masters Thesis Queensland University of Technology, Brisbane, Australian.

NEHTA (2005): Towards an Interoperability Framework, National E-health Transition Authority.

Sharanahan, D. and Karvelas, P. (2006): Welfare workers axed for spying. The Australian.

Steptoe & Johnson LLP (2008): E-Commerce Law Week, Issue 321. http://www.steptoe.com/publications-3133.html. Accessed 2/09/2008.