A Security Analysis of the Dutch Electronic Patient Record System

Guido van 't Noordende

ABSTRACT

In this article, we analyze the security architecture of the Dutch Electronic Patient Dossier (EPD) system. Intended as a national infrastructure for exchanging medical patient records among authorized parties (particularly, physicians), the EPD has to address a number of requirements, ranging from scalability and performance to security and privacy – as well as usability in (clinical) practice. The EPD is partially centralized. Patient records are stored decentrally, while a central component takes care of authentication and authorization of health professionals and of the mechanics required for exchanging patient records.

The requirements for the EPD, as well as descriptions of solutions and protocols, are described in a set of documents that are publicly available. This paper describes the security and privacy implications of the EPD design, argues where it falls short, and briefly discusses some improvements that may alleviate some of the risks that exist in the current design.

1.1 INTRODUCTION

The Dutch EPD is mandated by government as the infrastructure to use for exchanging patient information in the Netherlands, and is planned for introduction in 2009-2010. The EPD is designed by the Dutch National IT Institute for Healthcare (NICTIZ). The overall architectural design of the EPD

System and Network Engineering Group Technical Report UVA-SNE-2010-01
University of Amsterdam, the Netherlands
Science Park 107, 1098XG Amsterdam
guido [at] science.uva.nl

system is published under the name AORTA [1]. One of the most notable and widely referred-to features of the EPD design, is that it is developed as a partially decentralized system, in contast to, for example, the SPINE electronic patient record system developed in the U.K [2], which is fully centralized. Here, all patient records are stored in a central database managed by the National Health Service.

Dutch regulations do not favor storage of patient information in a central infrastructure [3, 4, 5]. This is due to legal, security, and privacy concerns raised by a centralized approach. The legal argument that favors decentralization over centralization, is that, in the Netherlands, the physician (and healthcare organization) who has a treatment relationship with a patient is responsible for managing the patient's dossiers [6]. Handing over control of management over patient records to a third party is in conflict with these regulations [3, 5]. The approach taken by the EPD is that patient records are not stored centrally, but instead remain stored in the information system of the hospital, GP, or other party responsible for managing the patient record(s) of given patients. To allow for finding and retrieving patient information using the EPD, a central *reference index (verwijsindex, VWI)* maintains a set of pointers to the patient records of each patient, using which the records can be retrieved.

Despite decentralized storage of patient records, authentication and authorization (to control access to patient records) in the EPD are fully centralized in the current design. Furthermore, some patient related information *has* to be stored in a central part of the system (such as the VWI), for the EPD to function. Because the EPD contains -in principle- information about all patients in the Netherlands, the privacy risks related to a potential security breach of the central components of the EPD are quite significant.

This paper discusses the architectural design and the mechanisms of the EPD, and evaluates some of the risks associated with the chosen approach. We also briefly discuss some ways to alleviate some of these risks using improvements to the architectural design of the EPD.

1.2 APPROACH AND ARCHITECTURE

The primary function of the EPD is to couple the decentrally stored patient records such that health professionals throughout the Netherlands can find and fetch patient records that are relevant, provided that they are authorized to see these records. Patient records are stored *decentrally*, i.e., only in the information systems of the care providers (such as hospitals and GPs) that have a treatment relation with the patients. The (central) EPD infrastructure provides the mechanisms for retrieving the decentrally stored patient records.

References to all patient records that are accessible through the EPD are registered in the VWI. The VWI references (index lines) describe the available

patient records to health professionals, and allow them to locate relevant patient records for retrieval. Alternatively, physicians may specify a *query* to let the EPD find and retrieve a set of records that match the query based on information stored in the VWI. Patients are identified using a unique number (the 'burgerservicenummer' or BSN, formerly known as the Dutch social security number), which can be looked up by means of a separate *BSN verification service* [7, 8]. VWI index lines and patient records are only visible to health professionals which are authorized to access the patient record in question.

1.2.1 Connecting to the EPD

Decentral information systems located at the care providers (e.g., hospitals, GPs, and pharmacies) are connected to a central part of the EPD infrastructure, called *Landelijk Schakel Punt (LSP, literal translation National Switching Point)*. All interactions required for finding and accessing patient records in the EPD go through the LSP. The LSP authorizes and logs all attempts to access information in the EPD.

Information systems must meet some general (security) requirements before they can obtain the credentials required to connect to the LSP [9]. These requirements are, to a large extent, organizational in nature and emphazise aspects such as management and maintenance procedures. Systems that meet these requirements are termed *GBZ*, which, translated from Dutch, stands for *well-managed care system*. The GBZ requirements are an (important) first step towards improving the security of systems that are part of the EPD. However, it is not possible to guarantee correctness of all systems hardware, operating systems, application programs, and usage of all systems that are part of the EPD; therefore, these requirements should not be viewed as a complete answer regarding the security of GBZ systems.

The connections between (decentral) GBZ systems and the LSP are cryptographically protected to avoid that outside attackers can listen in on the communication channels between GBZ and LSP. However, information passing through the GBZ or LSP systems is unprotected while *inside* these systems. Some of the components inside GBZ and LSP are shown in Fig 1.1.

1.2.2 Authentication

The LSP is a central component of the EPD infrastructure, which by design requires all systems that participate in the EPD to trust it. In particular, the LSP authorizes all requests in the system. Examples of requests are the retrieval of index lines from the VWI and requests for retrieving patient

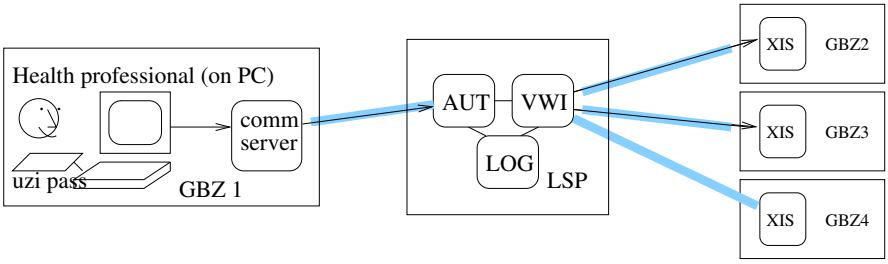

Fig. 1.1 Overview of the EPD, showing how the different components are connected. Several GBZ systems are shown, each connected to the central LSP system by means of encrypted, authenticated SSL connections (thick grey lines). In GBZ 1, a physician is shown who issues a request; this request is sent over a communication server and then forwarded to the LSP (arrows). Authentication and authorization takes place in the LSP (AUT); LOG is the component responsible for logging all requests. Using information from the VWI, the request is forwarded to two information systems (XIS) in different GBZs. Note that the SSL connections cannot prevent the GBZ or LSP components from reading or interfering with traffic that passes through them. The exact internal architecture of the LSP is not described in detail in the public AORTA documentation.

records. Underlying authorization lies an authentication mechanism which is also centralized. An authorization service (AUT, Fig. 1.1) located in the LSP takes care of authentication requests and enforcing (role-based) access control rules. The LSP is responsible for authentication and authorization of all requests sent to the EPD.

Health professionals can access the EPD using a personal smartcard that contains a public/private keypair. This smartcard is protected by a PIN code, and it is called a *Unique Healthcare provider Identification (UZI)* pass. Each smartcard contains a certificate containing information about the (medical) title, specialization, and function of the health professional, issued by a PKI based on Dutch professional registries. This information is used by the EPD for (role-based) access control.

All data in the EPD (messages, requests, patient records) are transfered as part of a *Health Level 7 version 3 (HL7v3)* (request) message [10] . HL7v3 is a standard supported by most existing healthcare related information systems.

Each request that is sent to the LSP is associated with a token. A token is a data structure, separate from the HL7v3 request message, which contains information required by the LSP to verify the authenticity of the request. The LSP compares the content of the token with the HL7v3 message. The token contains the *BSN* of the patient whom a request concerns, the *information category* that the request is concerned with, and some information to prevent replay. A token is signed by the health professional using his or her UZI pass before a request is sent to the LSP. Using the signature over the token, the LSP can authenticate (verify) which health professional made the request. Normally, a physician signs a token, but it is also possible that a token is signed by a mandated employee or co-worker (Section 1.4.2). The

token is removed by the LSP after authentication and *not* forwarded to the information system(s) that contain the patient record(s).

1.2.3 Access Control

The EPD defines two authorization policies to mediate access to patient records. These policies also apply to the VWI index lines for these records: if a health professional is not allowed to access a patient record, he or she cannot obtain the index lines regarding those records either.

First, an *authorization protocol* defines per class of health professional (e.g., GP, gyneacologist, pharmacist) whether that class is authorized to access a specific type of patient record. For example, a GP is allowed to inspect records created by a pharmacist, as well as records created by other GPs for patients that he or she has a treatment relation with. A pharmacist, on the other hand, is never allowed to see a GP patient record. The authorization protocol is agreed upon nationally by physicians and health organizations, and is enforced by the LSP.

Second, patients are able to define a fine-grained *authorisation profile*, which allows them to define *(restrict)* which health professionals or which care providers (hospitals or other organizations) may access their records. Details on the authorization profile are somewhat vague. In later sections, we will derive some limitations of the authorization profile based on what we know of the current AORTA authentication protocols and delegation mechanism.

1.3 SECURITY AND PRIVACY

This section focuses on technical aspects of the EPD. We describe some security weaknesses and risks in the current LSP design, both inherent risks and risk that can be alleviated by an improved protocol design.

1.3.1 Threat Assumptions

The possibility cannot be excluded that a system like the LSP will be successfully attacked. Because of its central role in authentication and authorization and the exchange of patient records, the LSP could be an attractive target for attackers.

There are obvious rewards in targeting the EPD to obtain private information from it; (financial) incentives could range from selling information concerning TV personalities to a magazine, to blackmailing high-profile peo-

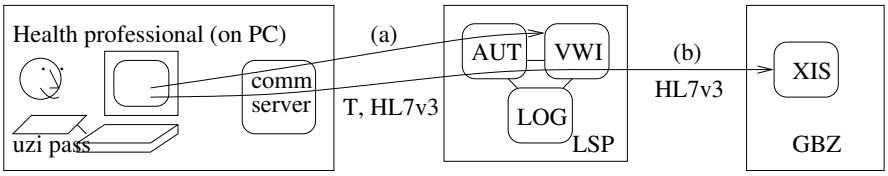

Fig. 1.2 Method invocation and token authentication overview. A request for (a) inspecting VWI index lines, or (b) retrieving a patient record, originates from a physician who signs a token T using his or her UZI pass. The token is sent along with the request *HL7v3*. In some cases, part of the request may be adapted (e.g., XML canonicalization) on a communication server in the GBZ before it is sent to the LSP; this does not invalidate the token. (a) depicts the protocol for requesting index lines from the VWI. In case (b), a patient record is retrieved. The LSP authenticates and authorizes the request (using the authentication service AUT), and forwards the HL7v3 request to the VWI, or to the decentral information system(s) (XIS) where the patient record(s) is or are stored. Replies (containing VWI index lines or patient records) take the same route back.

ple with a sexually transmittable and/or stigmatizing disease such as HIV. In this paper, an important *threat assumption* is that attackers may penetrate central parts of the system, such as the LSP.

Attacks could come from outsiders, but also from insiders who have access to core parts of the (LSP) system, or who have extensive knowledge of its inner workings. Resourceful attackers may be able to compromize core components of the LSP, from which they may be able to bypass regular access control checks. Causes may be viruses, developers, or bribed maintenance personnel who place blackdoors in the system, or personnel which accidentally opens part of the network through which an attack can take place. Of course, personnel in the LSP or in a hospital may also obtain patient information directly in some cases (Section 1.4.2). Assuming such threats is not unrealistic, certainly not for large systems in whose development, implementation, and deployment, many people are involved [11, 12]. Many recent reports emphasize the role that insiders play in real-world attacks [13].

In this paper, we focus on possible attacks that involve components in hospitals or the LSP. Examples are an application or communication server in a hospital, or a router used for handling traffic internal to the LSP. Information pertaining to many patients is exchanged over these systems. Even if the attack is only passive (i.e., listening to and possibly copying, but not changing any messages), a lot of information may be obtained by malicious software running on these components. The risks of these and more active attacks are evaluated in the remainder of this paper.

1.3.2 Inherent Risks

Although the EPD does not store all patient information in a single central system, central components such as the VWI still contain privacy sensitive information. For example, in each VWI index line, information about the hospital or organization that a patient visited is recorded, as well as information concerning the physician who registered the reference and the record type (e.g., GP or psychiatrist record, or lab result). VWI information may be sensitive in some cases, for example, think of a record at a cancer institute, a mental institute, or a rehab clinic. Depending on how the EPD is used, there may even be references to (specific classes of) lab results[1]. Normally, only authorized parties can see VWI index lines, but if an attacker manages to break into the central LSP infrastructure, all index lines of a given individual could be directly obtainable. Because the VWI is required for the functioning of the EPD, this is an *inherent risk* in the current design of the EPD infrastructure.

Patients are allowed to remove records from the EPD. However, removal of information from the EPD is currently not instantaneous, since patients cannot directly remove references to patient records from the VWI: the decentral systems (e.g., hospitals) are responsible for removing information and references from the LSP, possibly involving explicit action from the responsible physician. This may complicate timely removal of information from the EPD, and makes this information at least temporarily vulnerable. Also, time may pass before a patient notices that information was registered in the EPD.

By specification and by proposed law, the LSP is required to keep historical (traffic/access) information, and to allow the VWI to be restored to a previous state, until some *reconstruction horizon* in the past [1][2]. The reconstruction requirement implies that references which were explicitly removed from the EPD, may remain stored in the central LSP infrastructure to allow for reconstruction of the VWI. Traffic information relates to patient records, and may thus (implicitly) contain information about those records. Especially for explicitly removed information, this is a curious situation, as references may have been removed from the EPD precisely because they were considered privacy sensitive by the patient. As a result, complete removal of information (such as references) from the LSP is difficult or impossible, making this information potentially vulnerable to attacks on the infrastructure. Based on the sensitivity of the information, this may be a problem to some patients.

In the discussion section of this paper, we will briefly discuss some options to alleviate this issue.

[1] AORTA provides a mechanism for secure message transport (as a replacement for e-mail), which may be safer and more suitable for exchanging lab results than a (more permanent and more visible) registration in the EPD; however, public examples for using the EPD include registeration of lab results in the EPD [14], so we include the possibility here.

[2] This may be for a period of 15 years, matching the period of time in which physicians are required by law to keep their records [15].

1.3.3 Trusting LSP for Authorization

An important shortcoming of the token based authentication protocol, is that it does not permit for *end-to-end authentication*: the endpoint information systems where patient records are stored are *unable to authenticate incoming requests independently*, and thus cannot establish that this request is legitimate and originates from an actual health professional (i.e., and not some intruder who sends request messages directly from the LSP).

Informations systems cannot distinguish a forged message that originates from malicious software operating in the LSP from a legitimate message. As a consequence, any malicious code strategically positioned in the LSP can obtain any patient record from any information system connected to the EPD, *without* being questioned. The potential impact is high: a succesful attack on the LSP core infrastructure may allow an attacker to actively retrieve *any* patient record stored in any decentral information system connected to the EPD, without being questioned.

The AORTA specification does describe some XML headers to support end-to-end authentication protocols and (payload) encryption for future use. However, these protocols are not currently used for the AORTA *EPD* application. The lack of end-to-end authentication of patient retrieval requests is an important shortcoming of the current EPD design – a simple forwarding of authentication tokens together with the request messages to the decentralized information systems would suffice for these systems to instantly detect any forged messages originating from the LSP.

Note that healthcare providers such as hospitals are legally responsible for ensuring appropriate protection of data – including access control [6]. Therefore, *autonomy* of information systems to implement access control policies independently from the LSP is a very important property. An additional benefit of implementing end-to-end authentication, is that it allows decentralized systems to independently enforce access control rules (e.g., for blocking access to some patient records[3]) if required. Forwarding tokens to the decentral information systems together with the requests, allows these systems to independently check the integrity of each request, allowing them to detect attacks as well as potential mistakes in the authorization logic of the LSP.

[3] Note that, depending on the implementation of a decentralized information system, an attacker inside the LSP may be able to 'guess' references of patient records that have been blocked or removed, and request them explicitly. If an information system were to rely fully on LSP authorization, without making additional checks, this may even lead to a succesful attempt to retrieve *hidden* patient records from decentralized information systems.

1.3.4 Binding Information to Tokens

Tokens allow the LSP to determine the validity of a request, and to authorize the request given some access control rules. Access control rules consider *only* BSN and information categories, and do not enforce fine-grained policies at the level of individual patient records.

Each token contains a nonce and an expiry date to avoid replay, the category of the requested information, and the BSN number of the patient that the request is concerned with. Tokens do not contain an identifier for a specific patient record; in fact, the EPD implements a protocol where a request or query regarding a specific BSN and information category is forwarded (replicated) to all information systems that contain a patient record of the BSN and type and that matches the query, and have the LSP collect the results before returning them back to the requestor. Thus, a single token can theoretically be used to request *all* records of a given information class and BSN in a single operation. The token does not distinguish a request for a specific patient record from a request to obtain multiple patient records.

Any field in a request message which is not in the token, can be manipulated along the way from the requestor to the LSP without the requestor being aware of it, or the LSP being able to detect it. For example, query parameters are currently not embedded in the token. Suppose a physician wants to obtain all records of a given type up until a year ago, but nothing further back in the past. Malicious software on a communication server can change the query such that *all* available records of this patient and information category are retrieved. On return of the patient records, the malicious software may read the obtained data, but return only the requested data to the requestor to avoid raising suspicion.

The problem explored here is that insufficient information is embedded in the tokens used in the EPD, making it possible to expand requests to obtain a larger set of patient records than a physician intended. This problem is exacerbated by the exceptionally large scale of the EPD compared to other systems; indeed, even a slight change to a query may make a very large number of records available to a potential attacker[4].

In general, it is important that physicians retrieve only information which they require, preferably based on selecting VWI index lines. Recording the precise request (parameters, record identifiers) in the token, avoids that malicious software on the way from physician to LSP obtains more records than the physician requested by modifying a request message. Furthermore, embedding more information in the token would allow for enforcing more fine-grained access control rules in a patient's authorization profile, possibly on a per-record basis.

[4] Officially, patient records must be kept for about 15 years [6], so it may be normal to find references to patient records that far back in the EPD.

Including sufficient information in a token is also required to achieve the property of *non-repudiation* [11]: only for information signed by a requestor can it be shown (in court) that the requestor made this request - i.e., that no intermediate party or software on an intermediate system could have changed the request without the signer's knowledge. Not including all relevant information in the token provides attackers with an opportunity to deny (repudiate) claims of involvement with an attack in court. Therefore, all relevant information, including query parameters and/or explicitly requested patient record identifiers, should be encoded in the authentication tokens of the EPD.

1.4 AUTHORIZATION

The authorization model of the EPD is based on legal constraints. First, existing regulations concerning patient treatment and patient treatment teams provide a guideline on who may access medical data in the course of medical treatment [6]. Second, patients have a right to decide who may access which information, as defined in (European and Dutch) data protection regulations [16, 5]. This section describes some aspects of the EPD that relate to these constraints.

1.4.1 Patient Treatment Relation

When a physician becomes involved in the treatment of a patient, he or she must declare to have a treatment relationship with the patient [17]. The treatment relation is registered locally, in the physician's information system. When a physician accesses a patient record for the first time, the LSP takes this as an (implicit) declaration of a treatment relation, without any further verification: the LSP simply assumes that the treatment relation exists. Similarly, the LSP assumes that a treatment relation exists when a physician registers a record in the LSP[5]. Patients can use the access logs of the EPD to verify who accessed which data, and take (legal) action when detecting that a physician outside a treatment relation accessed their patient record(s).

Because treatment relations are currently not explicitly confirmed by patients, it is not possible to *automatically* verify the validity of a claimed treatment relation in the LSP. Because the LSP cannot verify at the time of invoking an operation whether a treatment relation actually exists, the validity of a treatment (team) relation can only be verified after the fact.

[5] It may be straightfoward for an employee to register information regarding a patient to claim a treatment relationship in prepation of an attack. Also, registering information in the EPD may be critical not just to security, but also to integrity of a patient's records. For these reasons, we believe that these operations should be reserved for physicians alone.

Patients will in the future be able to access their patient records, adapt their authorization profile, and inspect logging information about who accessed which information of their EPD [18]. Patients can thus verify whether access to their EPD by (or on behalf of) a particular health professional was legitimate, that is, that those health professionals indeed had a treatment relationship with the patient at the time of obtaining a patient record [6, 18].

AORTA provides a model that allows authorized health professionals to *delegate* authority to access the EPD to (unauthorized) employees or co-workers. This is a valid operation: multiple people including co-workers and employees may be part of a patient's *treatment team*, and are in that role authorized to access the patient's patient record(s) - at least as far as the physician is permitted to access these records [6]. However, it is not clear if the notion of a treatment team should extend to EPD authorization (delegation) per-se (see Sections 1.4.2, 1.5). Note that it may often not be clear to patients who is a valid member of a treatment team. This makes it very difficult for patients to assess whether access to the EPD by a particular employee on behalf of a given physician was legitimate or not, even when detailed access logs are available[6]. Also, not all patients may be willing or able to inspect the access logs of their EPD, or do so in a timely manner.

1.4.2 Delegation

AORTA's delegation model (called *"mandatering"* in the AORTA specification) is decentralized. Every care provider that makes use of delegation, must maintain a *delegation table*. The delegation table describes which employees are allowed to access the EPD on behalf of which health professionals.

Employees of care organizations can have a personal UZI pass similar to those of health professionals, except that the certificate on this pass contains only the employee's name, and not a medical title. Normally, this pass may only be used for low-security tasks, such as verification of a patient's BSN number [7][7], except when a physician delegates authority to access the EPD to this employee. Physicians can delegate authority to access a patient's records to any employee that has a UZI pass. Employee passes can in principle be used for obtaining any patient record of any information class on behalf of any physician, as long as the mandating physician is authorized to access this record. The LSP assigns exactly the same rights to a mandated employee, as to the mandating physician.

[6] It is unclear if the names / details of mandated employees are included in the access logs visible to patients, or whether the authorization profile will contain functionality to deny access to (specific) employees.

[7] Note that patient treatment relationships may be derived from the BSN verification logs, by inferring information about who requested BSN verification for which patients. Care should be taken to protect the access logs of the BSN verification service accordingly.

The delegation mechanism is implemented as follows. A mandated employee signs a token for the request message using his or her UZI pass. The health professional on behalf of whom the EPD interaction takes place, is noted as the *overseer* in a field of the HL7v3 message (this field is not present in the token). Based on this field, the LSP can tell who the overseer is, and based this information assign permissions and decide what access is allowed.

1.4.3 Issues related to Delegation

Currently, there is no way for the LSP to verify *at invocation time* whether an employee truly acted on behalf of a given physician or not: there exists no way for either the employee or the physician to assert or prove to the LSP that an employee is indeed mandated by the physician; AORTA fully relies on security of the GBZ systems delegation tables, and, if required, on inspection of LSP audit logs after the fact.

The delegation table is used as an auditing tool that allows maintainers of the EPD to verify whether an interaction of an employee could have been made on behalf of the physician specified in the overseer field. Working schedules or agenda entries are used to establish whether a particular employee could have accessed an EPD patient record legitimately on behalf of this physician or not [1]. Note that it may well be easy to tamper with delegation tables to cover up mistakes - for example in case of a hospital or physician that wants to avoid getting a bad reputation.

The delegation system is vulnerable to an attack that combines malicious code within a GBZ (that allows for generation of tokens and HL7v3 messages) with a (stolen) employee pass together with a PIN code. Employee UZI passes can be used to sign tokens on behalf of arbitrary overseers. (In fact, the overseer field is not even included in the token, so it could be tampered with by anyone without the signer's knowledge – a fact which could allow an employee who conspired with an attacker to deny involvement with the attack in court). By constructing a HL7v3 message with a suitable overseer field and signing the accompanying token (whose information category should match the overseer's specialization) with any (stolen) UZI pass, patient records can be retrieved on behalf of effectively any physician in the hospital. Similar attacks can be mounted by malicious software on a (set of) desktop PC(s), which lets the UZI pass of an employee sign a token with a different information category than this user intended[8].

Misuse of the delegation mechanism can currently only be prevented by blocking a complete GBZ in an authorization profile; otherwise, there are

[8] We assume for the moment that an employee's desktop PC may be more easily compromized than a physician's PC. When a physician's PC is compromized, similar attacks are possible - in addition to more direct misuse of the physician's authorization. For the purpose of this paper, we do not further address this attack possibility.

few limits to this attack. The attack may be particularly powerful because delegation can also be used to claim a treatment relationship, as far as the LSP is concerned [15]. This is a direct consequence of the fact that the EPD relies on security of the decentralized GBZ systems to handle delegation (and patient administration) correctly.

The above assumes a *software attack*. However, depending on the implementation of the local information system, easier ways to abuse the delegation mechanism are conceivable. For example, it may be possible to simply choose an overseering physician using some drop-box of the local information system, to interact with the EPD on the chosen physician's behalf. When such illegitimate use of delegation does not take place too often, it may well go undetected, because a patient who checks the access logs will often not be able to tell who was part of his or her treatment team at a particular time.

At a high level of abstraction, the problem is that the authorization model is *reversed*: instead of a patient actively authorizing physicians, who then actively authorize co-workers or employees directly involved in treating a patient, employees not known to the LSP (or the patient) can claim to work for any given physician by simply having the system fill in an arbitrary physician in the overseer field. Worse, arbitrary employees can obtain practically any patient record from the EPD system, because the LSP assumes that a physician who invokes an operation on the EPD, has (locally) declared a treatment relationship with this patient, and because the LSP assigns all rights of the physician to any employee who claims to work for this physician. Mandated employees can thus even invoke the request to obtain a record of a new patient, implicitly declaring a treatment relation – an operation that we would expect to be reserved for physicians alone.

Allowing physicians to claim a patient treatment relationship and then allowing them to access a patient's records based on this claim may be defendable and relatively safe, even when depending on verification after the fact: registered physicians are held to professional ethics, have a reputation to uphold, and they can be held accountable for their actions by means of professional sanctions. However, it seems unacceptably risky to assign all the rights -including the implicit right to claim a treatment relation- to any employee who claims to work on behalf of a physician, without any possibility to verify this claim in the LSP.

As far as we know, there exists no (standard) way for the LSP to obtain delegation table information and/or working schedules automatically from the decentral information systems. Were this possible, there would at least be a way to *probabilistically* or *on-the-fly* verify delegation information, for example each first time that a given employee invokes an operation using a new overseer field – although this approach is vulnerable to attacks that manipulate working schedules or delegation tables. Other ways to implement delegation confirmation are discussed in Section 1.5.

In conclusion, a mechanism for *confirmation* of a delegation relation by the overseeing physician is required in the LSP. Verification of the validity

of a specified overseer field should take place *prior to or at the time of an operation* on the LSP, and access should be denied when an unconfirmed delegation relation is found. Only physicians should be allowed to (implicitly or explicitly) claim a patient treatment relationship, and this should be securely recorded in the LSP. Note that if at some time an employee cannot access a patient's record, a physician is always capable of accessing it in person; it thus seems unnecessary to take any chances.

1.4.4 Auditing and Repudiation

The proposed law regarding the EPD emphasizes extensive logging and auditing of these logs as a cornerstone of EPD security [18]. However, the lack of verifiability of delegation and patient treatment relations complicates auditing, and limits the probability that abuse is detected in time.

Key questions raised in this section are:

- *How can the LSP (or an auditor) establish whether an overseer field is valid, when the overseeing physician cannot explicitly confirm that the employee or co-worker is mandated, or at least part of the patient's treatment team?*
- *How can the LSP (or an auditor) distinguish a genuine claim of a treatment relation from an illegitimate one when a patient cannot confirm this relation in the EPD?*

Because of the lack of automatic verifiability of the above relations, the EPD must depend on heuristics, 'intelligence', or manual procedures to distinguish valid treatment and working relations of physicians or employees from invalid ones. It is easy to envision how malicious software can evade detection by letting the misuse exhibit behaviour which is difficult to distinguish from normal usage behaviour.

An important assumption of the *trust model* that underlies authorization in the EPD, is that one can always address the responsible physician (overseer) when something goes wrong. However, our analysis of the internal protocols shows that this assumption does not hold, because any overseer can be filled in in a HL7v3 message *without involving the physician*. Essentially, it will be difficult to hold a physician accountable when the overseer field –which points to this physician as the party responsible for a given employee's actions– cannot be verified as being valid, and when the physician is not involved in constructing or validating the message or the claimed delegation relation.

The basic problem is that the 'chain of involvement' with a patient's treatment - from treating physician to delegated employee - is not clear. Because of the inherent lack of verifyability of the basic relations that underly all authorization decisions in the EPD, the LSP loses out on the possibility to filter

out confirmed (completely authorized) operations. This would allow auditing to focus attention on suspicious operations, rather than having to (also) discover potentially false delegation or patient treatment relation claims.

1.5 SOLUTIONS AND DISCUSSION

This paper highlighted some design, deployment and organizational issues of the EPD which may have an impact on the security it provides.

On the one hand, we found a number of implementation aspects, such as a lack of end-to-end authentication, and inclusion of insufficient information in authentication tokens, which may increase the potential impact of an attack unnecessarily. On the other hand, we found that the authorization policies are not supported by sufficient (verifiable) confirmation of the patient treatment and delegation relations that underlie authorization in the EPD. This makes validation of, in particular, delegation information which is used for access control decisions, very difficult in practice.

Combined, these issues undermine the effectiveness of the authorization policies embedded in the EPD, and limit auditability in general.

Effectively, we distinguish three overall problems in the current design:

- Technically: a lack of end-to-end authentication combined with incomplete and insufficient information embedded in tokens.
- Policy/organizatorial and implemention-wise: there exists no mechanism in the LSP for immediate confirmation of delegation relations and (eventual) confirmation of patient treatment relations.
- An inherent risk of information leakage: attacks on VWI and historical information stored in the LSP may allow attackers to obtain this information directly. In particular for historical information, this may be an important risk.

Solving the technical problems is relatively straightforward. End-to-end authentication can be achieved by forwarding the signed tokens to the endpoints such that these endpoints can independently authenticate (and possibly authorize) incoming messages; embedding additional information regarding the requestor's original request in the authentication tokens can restrict the scope of attacks that involve tampering with a request. It can also enable fine-grained policies on a per-record basis in the authorization profile. End-to-end encryption is a logical next step -based on end-to-end authentication- to prevent information leakage through compromised systems between requestor and the system where a patient record is stored.

The problem of storing historical (traffic, logging, reconstruction) information in the LSP can be solved by allowing complete, unconditional, and undelayed deletion of all information related to a specific patient record or

treatment relation from the LSP, including logging information. A less rigorous approach would be to encrypt all identifiable historical traffic and VWI information using a public key associated with the patient. Such a public key could be available if patients have access to some type of patient identification smartcard similar to an UZI pass, or possibly an electronic National Identity Card [19]. Anonymized traffic information may still be made available for traffic analysis in this case. Introducing a patient identification pass with cryptographic capabilities can also alleviate some risks related to patient access to the EPD - where a (centralized) attack is currently possible due to -again- a lack of end-to-end authentication of, in this case, patients [19].

To improve security of the delegation mechanism, delegation should only be allowed after *explicit, verifiable confirmation* of the delegation relation by the overseeing physician is registered in the LSP - either before or at the time an operation is invoked. In other words, a care organisation has to *prove* to the LSP that a particular mandate is valid. The reason is simple: authorization should flow 'down' – from patients to physicians to employees. Allowing access to some record simply because an employee says (by means of a HL7v3 field) that he or she is mandated by a physician (who is automatically assumed by the LSP to have a treatment relation with the patient when he or she invokes an EPD operation; a dangerous assumption when combined with delegation), simply places too much trust in an employee who does not have a medical title, and who is not known to and has no direct professional relationship with the patient.

A possible way to implement delegation confirmation, is to create a *delegation certificate* for each possible mandatee, which has to be shipped with each message and token and can be checked by the LSP. Delegation certificates must be signed by a physician, or by someone authorized for this task by the physician or the hospital. Preparation of delegation certificates may be automated using delegation tables, possibly using working schedules. Delegation certificates may be constrained, for example by making them valid only for a limited time interval or a limited number of operations. Although the approach is not infallible – for example, the process of creating and signing delegation certificates may be manipulated, as well as delegation tables and schedules – at least attacks are made more difficult and less effective. An additional approach to implement constraints on delegation, is to constrain employee UZI passes such that they can only obtain records of an information category that matches the specialization of the physician(s) that the employee works for. In cases where (timely) creation of a delegation certificate for a given employee is not possible, a health professional can always interact with the EPD personally without delegation.

Employees within a treatment team may abuse their given mandate directly. This is an inescapable risk, and falls under the legal responsibility of the health professional. Overall, health professionals should be careful with mandating employees. In terms of enforcement, it is imperative that *only* physicians should be allowed to claim a treatment relation in the LSP – ei-

ther explicitly or implicitly by invoking the first request to obtain a record of a patient or by registering a record in the LSP. These rights should not be delegatable; only this way can a physician be held accountable for a false patient treatment claim.

To facilitate auditing, patients could sign an explicit *treatment confirmation*, which is verifyable by the LSP. Using treatment confirmation, it becomes visible which treatment relations are legitimate, allowing auditing to focus on unconfirmed cases. Treatment relations may be confirmed *eventually*, after the fact. Also, depending on the information type, patient confirmation may be required *before* allowing access to patient records. In some cases, related (logging) information in the LSP can be encrypted or removed after treatment confirmation. If some time after access to the EPD, a patient has not confirmed a treatment relation, the system could send a letter to a patient requesting confirmation, or inform with the patient's family or the hospital where an (emergency) exception took place[9]. Such procedures ensure that (written or electronic) confirmation eventually takes place.

As a final remark, we observe that in the current EPD design much emphasis lies on facilitating efficient information exchange between physicians, particularly on efficiently obtaining sets of patient records, which is indeed the primary objective of the EPD [15]. However, for people who do not fully trust the current infrastructure, or who fear an intrusion of their privacy in case of (accidental) leakage of some information from the EPD, there appear to be few mechanisms available to straightforwardly *prevent*, on a per-case basis, that particular medical information gets stored in their EPD.

Patients have a right to *opt-out* of usage of the EPD for storing and exchanging their personal information altogether [18]. Patients also have the right to indicate to their physician and/or care organization, that they do not want their information to be registered in the EPD. However, there may be situations where patients to not have the time to indicate such a *GBZ-specific* opt-out, for example when they have to be treated urgently, or are in no state to inform their physician or care organization that they want some particular information to remain out of their EPD. It currently appears impossible to completely (or instantly) remove information (such as references to patient records) from the EPD's central infrastructure once it has been registered (Section 1.3.2). Although historical information is not visible to normal users, it may be visible to attackers in certain scenario's. Therefore, it could be important to ensure that patients are able to *prevent* registration of specific information in the EPD.

In the long term, physicians and care organizations may become so accustomed to using the EPD for exchanging medical information, that a (full or even per care organization) opt-out may become impractical when patients want effective, affordable, or efficient treatment. Meanwhile, the EPD may become so well-integrated with local information systems, that registration

[9] Currently, patients only get a letter for the *very first registration* of information in the EPD, to ensure that citizens are aware of the possibility of opting out of the EPD.

happens quickly and physicians may barely notice when they register information in the EPD.

To address this issue, we believe it would be useful when patients could indicate in their authorization profile -in the EPD- that they wish to be informed (or *asked for consent*) *before* any of their information is registered. Such a consent preference may, for example, pop up whenever a treatment relationship is newly established to remind the physician that, for this patient, registering information in the EPD may have privacy consequences. Note that patients who do not care about this option, do not have to enable it; however, it may be important in some cases. A more flexible consent model may also increase the confidence of people who might otherwise object to using the EPD[10] – in particular in view of attacks that -invariably- continue to threaten the system's security.

Acknowledgements

Thanks for Niels Sijm for taking the first steps into this research [21]. Thanks to the (anonymous) reviewers for valuable comments which improved this paper. Special thanks go to Marie-José Bonthuis and Matthijs Koot for reviewing early drafts of this paper.

References

1. Nictiz. AORTA Documentation Release. *http://www.infoepd.nl/informatiepunt_com/ aorta-documentatierelease_2008_totaal.php*, October 2008.
2. U.K. National Health Service (NHS). SPINE - NHS Connecting for Health. *http://www.connectingforhealth.nhs.uk/ systemsandservices/spine*.
3. J.C.J. Dute et al. Rapport Evalutatie WGBo. 2003.
4. Groep Gegevensbescherming Article 29. Werkdocument inzake de verwerking van persoonsgegevens betreffende gezondheid in electronische medische dossiers (EMD). *http://europa.eu.int/comm/justice_home/ fsj/privacy/index_en.htm*, February 2007.
5. Marie-José Bonthuis. Privacy en het Landelijk Electronisch Patientendossier (EPD), Universiteit Groningen. 2007.
6. Dutch ministry of health, welfare and sports (VWS). Wet op de Geneeskundige BehandelingsOvereenkomst (WGBO). *http://www.hulpgids.nl/wetten/wgbo.htm*, 1994.
7. Sectorale Berichten Voorziening in de Zorg (SBV-Z). *http://www.sbv-z.nl/*, 2009.
8. Wijziging van de Wet gebruik burgerservicenummer in de zorg in verband met de electronische informatieuitwisseling in de zorg (31-466). *Eerste Kamer 31 466, A, SDU publishers*, 2008-2009.
9. Nictiz. Programma van Eisen aan een GBZ (PvE GBZ). *http://www.aortarelease.nl/content/inf/ Programma_van_Eisen_GBZ.html*, 2007.

[10] Recently, it was reported that almost half a million Dutch citizens have opted out of using the EPD to date [20].

10. Health Level 7 standards for interoperability of health information technology. *http://www.hl7.org/*.
11. Ross Anderson. Security Engineering, 2nd edition. *Wiley*, 2008.
12. Ken Thompson. Reflections on Trusting Trust (Turing award Lecture). *Communications of the ACM Vol.27(8)*, August 1984.
13. D. Cappelli, A. Moore, R. Trzeciak, T. Shimeall. Common Sense Guide to Prevention and Detection of Insider Threats, 3d edition - v3.1. *CERT Report, Carnegie-Mellon University*, January 2009.
14. Ministerie van VWS. Informatie en bezwaarschrift EPD. 2008.
15. Nictiz. Personal communication. 2010.
16. European Parliament and the Council. Directive 95/46/EC on the protection of individuals with regard to the processing of personal data and on the free movement of such data. *http://ec.europa.eu/justice_home/fsj/ privacy/docs/ 95-46-ce/dir1995-46_part1_en.pdf*, October 1995.
17. Nictiz. Memo behandelrelatie en additionele GBZ-eisen. april 2009.
18. Minister A. Klink. Memorie van antwoord bij de Wijziging van de Wet gebruik burgerservicenummer in de zorg in verband met de electronische informatieuitwisseling in de zorg (31-466). *Eerste kamer der staten-Generaal*, September 2009.
19. B. Jacobs, S. Nouwt, A. de Bruijn, O. Vermeulen, R. van der Knaap, C. de Bie. Beveiligingseisen ten aanzien van identificatie en authenticatie voor toegang zorgconsument tot het Electronisch Patientendossier (EPD). *Report by PriceWaterhouseCoopers, Radboud Universiteit Nijmegen, and Universiteit van Tilburg*, december 2008.
20. M. Katzenbauer. Te vroeg voor landelijk EPD. *medisch contact 64 nr.20*, pages 880–883, May 2009.
21. Niels Sijm. Onderzoeksrapport LSP. *https://www.os3.nl/_media/2007-2008/courses/rp2/ns-report.pdf*, 2008.