The e-document case study: functional analysis and access control requirements

Maarten Decat
Jasper Bogaerts
Bert Lagaisse
Wouter Joosen

Report CW 654, February 2014

Katholieke Universiteit Leuven
Department of Computer Science
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

The e-document case study: functional analysis and access control requirements

Maarten Decat
Jasper Bogaerts
Bert Lagaisse
Wouter Joosen

Report CW 654, February 2014

Department of Computer Science, K.U.Leuven

Abstract

Software-as-a-Service (SaaS) is a maturing model for offering online applications which is drawing a growing interest from industry. However, SaaS is still facing many challenges which hinder its widespread adoption. One of these challenges is manageable and effective access control in the presence of the multiple organizations involved. The first step to address this challenge is clarifying the requirements for access control for SaaS and the challenges that result from them. To achieve this, we analyzed a case study of a SaaS application in the domain of electronic document processing. The analysis was performed with the cooperation of the involved company (which is anonymized in this document). This document (i) describes the SaaS application itself, using an illustrative scenario, use cases and textual non-functional requirements and (ii) provides a set of access control policies that apply to this application.

Keywords : C.2.4 [Computer-Communication Networks]: Distributed Systems - Distributed Applications, D.4.6 [Security and Protection]: Access controls.

Contents

1 Introduction **5**

2 Illustrative scenario **6**
2.1 Basic scenario . . . 6
2.2 Sub-organizations . . . 8
2.3 Resellers . . . 10
2.4 Zoomit . . . 11
2.5 The whole picture . . . 12
2.6 Remark: Provider conflict of interest 1 . . . 13
2.7 Remark: Provider conflict of interest 2 . . . 13

3 Functional requirements **14**
3.1 Actors . . . 15
3.2 Use Cases . . . 16
3.2.1 UC1: Log in . . . 17
3.2.2 UC2: Log out . . . 18
3.2.3 UC3: Send document . . . 18
3.2.4 UC4: Send document to postal address . . . 19
3.2.5 UC5: Send document to e-mail address . . . 20
3.2.6 UC6: Send document to Registered Receiver . . . 20
3.2.7 UC7: Send invoice using Zoomit . . . 21
3.2.8 UC8: View received digital document (Unregistered Receiver) . . . 22
3.2.9 UC9: View received digital document (Registered Receiver) 22
3.2.10 UC10: Get overview of received documents . . . 23
3.2.11 UC11: View sent document . . . 24
3.2.12 UC12: Get overview of sent documents . . . 24
3.2.13 UC13: Search documents . . . 25
3.2.14 UC14: Register . . . 25
3.2.15 UC15: Register (as Private Receiver) . . . 26
3.2.16 UC16: Register (as Receiving Organization) . . . 26
3.2.17 UC17: Update organization configuration . . . 27
3.2.18 UC18: Add print house . . . 28
3.2.19 UC19: Deliver documents to Print House . . . 28

4 Non-functional requirements **29**

5 Glossary **30**

6 Policies from the scenario **31**
6.1 General for the application . . . 31
6.2 eDocs . . . 31
6.3 Large Bank . . . 33
6.3.1 Large Bank Leasing . . . 35
6.3.2 Local bank offices . . . 35

6.4 Car Leaser . . . 36
6.5 ICTProvider . . . 36
6.6 NewsAgency . . . 37
6.6.1 Europe Region . . . 37
6.6.2 London Office . . . 38
6.7 Reseller . . . 39
6.8 Registered Private Receivers . . . 40

7 Conclusion **41**

List of Figures

1 Overview of the basic scenario 7
2 Overview of the basic scenario (Figure 1) extended with subtenants. 8
3 Example of a multi-level subtenant hierarchy. 9
4 Overview of the basic scenario (Figure 1) extended with subtenants (Figure 2) and resellers. 10
5 Zoomit scenario. Note: the relationship between eDocs and Large Bank as described in previous sections is not regarded any more. 11
6 Complete overview of the scenario in terms of business relationships. 12
7 Illustration of a first case of a conflict of interest for the provider. 13
8 Illustration of a second case of a conflict of interest for the provider. 14
9 Actor and organization hierarchy 14

1 Introduction

Software-as-a-Service or *SaaS* is a maturing model for offering online applications with a growing interest from industry. Software-as-a-Service (SaaS) is a type of cloud computing in which a tenant organization rents access to a shared, typically web-based application hosted by the provider [1]. For the tenant, SaaS promises the benefits of ease of use and low management costs: employing a SaaS application does not require the tenant to have specialized on-premise IT infrastructure, nor skilled (and expensive) IT personnel. For the provider, SaaS also promises lower management costs by allowing the same application to be utilized by multiple tenants, a concept called multi-tenancy. However, SaaS is still facing many challenges which hinder its widespread adoption. One of these challenges is manageable and effective access control for SaaS applications in the presence of the multiple organizations involved.

Access control is an important part of application-level security that, from a high-level point of view, limits the *actions* a *subject* (e.g., a user) can take on an *object* in the system (e.g., a file). On the one hand, the process of access control can be divided into *authentication* and *authorization*. Authentication confirms the stated identity of a subject, for example by checking that the subject knows the combination of a username and password; authorization subsequently confirms the subject is allowed to do the desired action on the desired object. On the other hand, access control can also be divided into the two main concerns of *management* and *enforcement*. Access control management is generally referred to as *Identity and Access Management* or *IAM*. The management of authentication data is generally referred to as *User Management* or *Identity Management* and its primary purpose is to manage user accounts and their properties. The management of authorization data is generally referred to as *Entitlement Management* and its primary purpose is to manage the rights (or entitlements) of users in the applications of the organization. A user's rights are often not defined explicitly but inferred from declarative *access control policies* which define the rules that constrain the actions of users in an application. Since these policies reason about the properties of users, entitlement management is closely related to identity management. Enforcement of these policies is then externalized into security middleware.

In this document, we investigate the requirements for access control for SaaS by analyzing a case study of a SaaS application in the domain of electronic document processing. The analysis was performed with the cooperation of the (existing, but anonymized) company eDocs. eDocs is a company that provides a multi-tenant e-document processing application as a service. In a nutshell, the application allows the tenants to distribute documents to their respective customers, either digitally or physically (by printing them and employing snail mail), optionally after creating the documents out of raw data using predefined templates. The application is a good example of an application that has to cope with complex business relationships, which results into complex requirements for access control. Instead of describing the system as a whole, this document focuses on one specific scenario for illustrating resulting access control require-

ments. More specifically, this document (i) describes the SaaS application itself, using an illustrative scenario, use cases and textual non-functional requirements and (ii) provides a set of access control policies that apply to this document.

The rest of this part is structured as follows: Section 2 informally describes the illustrative scenario, starting from a basic scenario and gradually adding complexity. Section 3 describes the system from the scenario more formally as use cases. Section 4 briefly lists some important non-functional requirements for the system. Section 5 gives a glossary of the most important terms used in this document. Section 6 elaborates on access control policies that apply to the system in the context of the presented scenario. Finally, Section 7 concludes this part.

2 Illustrative scenario

This section illustrates the scope of this case study by describing a specific scenario. We start by describing the basic setup of the application and gradually add more complexity.

2.1 Basic scenario

In the most basic scenario (see Figure 1), the application provider (eDocs) provides the application directly to tenants, such as Large Bank. Therefore eDocs employs application admins managing the application (e.g., creating new tenants after a new customer has been signed) and provides supporting services such as a help desk for tenants (e.g., for retrieving lost documents, undoing incorrect actions or helping with user management).

The tenant Large Bank is a large bank that uses the application to send documents both to internal receivers, e.g., documents sent from central offices to local bank offices, and to external receivers, e.g., invoices sent to business customers such as an organization as ICTProvider, or banking notes sent to private customers such as a person called Bart. The invoices are sent manually by the employees of Large Bank. In order to do so, the employee uploads the invoice (or the raw data needed to generate it) and provides the destination. The banking notes on the other hand are sent automatically in bulk, each note also specifying a destination. The destination can be a registered party in the system or, in case the receiver is not registered, a simple e-mail address or postal address. In the former case, the invoice will be added to the receiver account and an update e-mail will be sent. The receiver can visit the application and log in to see and manage its received documents. In the latter case, the invoice will be sent as as an e-mail attachment or an email containing a URL where the document can be downloaded (e.g., with unique identifier), or will be print and sent to the postal address using snail mail. For printing a document, eDocs employs the services of one or multiplePrint Houses, possibly located around the world.

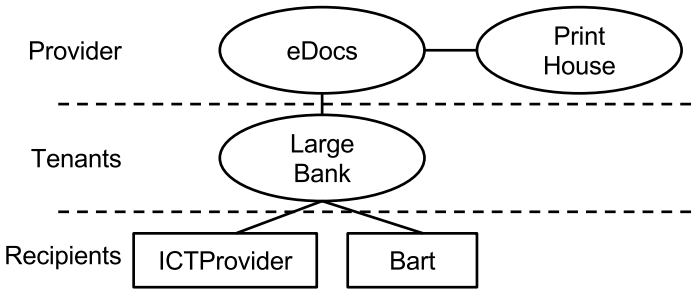

Figure 1: Overview of the basic scenario

A tenant can use the application to send documents to both external as internal receivers, get an overview of the sent documents, search the sent documents and manage its destinations (e.g., create a receiver organization). An unregistered receiver can only view received digital documents as e-mail attachment or unique URL. A registered receiver can use the application to view received documents and search received documents. Tenants can also manage internal users and their rights, for example specifying user accounts, declaring roles and specifying which users/roles can view which documents (e.g., invoices are sent to the financial office, customer complaints to the customer care office). This functionality is also available to registered receiving organizations, allowing them to organize the documents which are sent to them and who will handle which types of documents.

A tenant can register receivers or receivers can register themselves after receiving an email with a URL. Notice that registering a new receiver is not trivial. A receiver will always register after having received documents when not being registered. In that case, the document was sent using snail mail or e-mail. The e-mail will contain information on how to register, after which the document itself can be added to the receiver account (e.g., by using an id in the URL pointing to the registration page). However, after registering as a receiver, it is unclear how the senders can send to the new receiver account. A first option is adding the receiver to the sender's address book. However, this will require a change in operation for the sender. Another option is automatically adding the documents sent to the organization using its e-mail address or postal address to its account. However, this mapping is hard to get right, since the format of a postal address can differ from sender to sender and an organization can have multiple e-mail addresses. Allowing the receiving organizations to manage its e-mail addresses in a usable and secure way is also not trivial. A third option is having the receiver inform the organizations that he/she has a registered account at eDocs. This is similar to adding the receiver to the sender's address book, but it is end-user driven. It requires the receiver to have an advantage in informing the organizations that he/she has a registered account at the application, for example, a central platform on which he/she can configure how to receive documents from the organizations he/she is connected

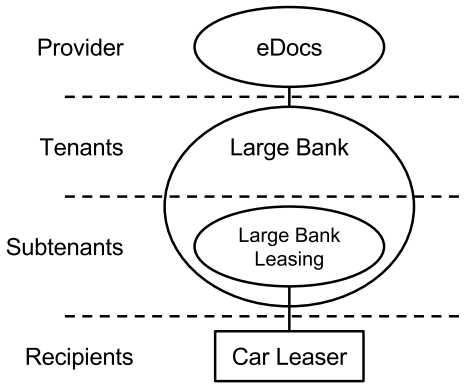

Figure 2: Overview of the basic scenario (Figure 1) extended with subtenants.

to.

Technically, the application is a web application which is located in an organization-specific sub-domain of the provider domain for both senders and registered receivers (e.g., largebank.edocs.com or ictprovider.edocs.com). Possible alternatives are a tenant-specific sub-folder in the provider domain (e.g., www.edocs.com/large-bank), a sub-domain of the organization domain (e.g., documents.large-bank.be) or a general (i.e., not organization-specific) provider domain (e.g., www.edocs.com). A tenant-specific domain has the advantage of inherently identifying the organization to which the user belongs.

2.2 Sub-organizations

In reality, the basic scenario from above is too simplistic and has to be extended on several points. Firstly, tenants can be large and complex organizations consisting of autonomous sub-organizations (see Figure 2). For example, Large Bank consists of Large Bank Leasing, Large Bank Insurances etc. Because of the choice of the parent organization to employ the application and become a tenant of eDocs, each of these sub-organizations also uses the application. However, each of these organizations is independent from each other and requires separate management capabilities. This also counts for the eDocs application and the sub-organization thus becomes a subtenant for the application. For example, in the case of Large Bank, the parent organization declares it consists of multiple sub-organizations, after which each sub-organization can declare its internal structure in terms of users, roles, permissions etc. A subtenant can also be separately billed for the use of the application.

In practice, the hierarchy is not constrained to a single level, i.e., a subtenant can consist of sub-organizations itself (see Figure 3). An example of such a tenant is NewsAgency. NewsAgency is a large, international news agency. The headquarters are mainly concerned with high-level and strategic management, auditing and financing. The rest of the organization is organized by region: Eu-

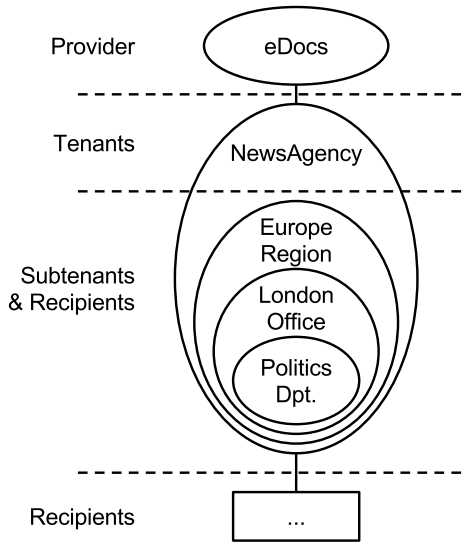

Figure 3: Example of a multi-level subtenant hierarchy.

rope, Asia, North-America, etc. Each region is then organized as departments and offices. Each of these levels should be supported as subtenants in the application. Moreover, each of these organizations is also a receiver, since documents can be sent internally.

A subtenant is added to the hierarchy as follows: an employee of NewsAgency can create a subtenant (if he/she was internally assigned the access rights to do so - usually this implies the employee is the tenant administrator). Doing so, he/she will have to provide a sub-domain name which will be the interface for the subtenant to the application of eDocs. Also, an authentication/authorization interface will have to be given to allow the subtenant to control access to the application. Moreover, the employee provides a contact address for the subtenant administrator, which is an employee of the Europe Region. The eDocs administrator of the Europe Region is then contacted to allow him/her to create the subtenant account with any data that eDocs needs in order to handle requests by the Europe Region employees. In a federated situation, this implies that the local account of the tenant administrator of the Europe Region is linked with a federated account at the eDocs site, where additional user data may be stored. This situation would also require the tenant administrator to provide an Identity Provider to eDocs in order to allow authentication of the Europe Region employees.

When registered, the Europe Region tenant administrator can also add a subtenant using the same process.

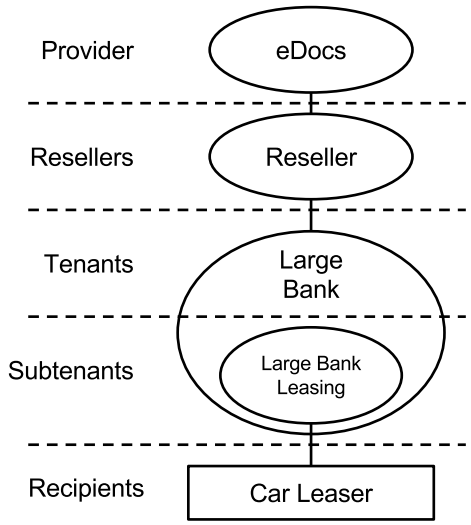

Figure 4: Overview of the basic scenario (Figure 1) extended with subtenants (Figure 2) and resellers.

2.3 Resellers

As a second extension, we can take into account resellers (see Figure 4). Resellers are tenants in the sense that they rent access to the application from eDocs, but differ from "normal" tenants in that they do not use the application for sending documents, but resell (or rather, re-rent) the application to other organizations. Customers of resellers thus are the actual tenants as described above. Resellers are a useful way for eDocs to provide the application to tenants and they can provide additional services to their customers. For example, the reseller Reseller extends the basic application by adding legally certified storage and functionality aimed specifically at receiving and handling traffic fines for leasing companies, e.g., Large Bank Leasing with customers such as Car Leaser. Resellers can also add additional supporting services such as an extensive helpdesk or outsourced application management. Thus, the reseller acts similarly to an application admin from eDocs and should be able to manage its tenants and their documents (and those of possible subtenants).

The process of adding a new customer tenant as reseller is similar to that of a normal tenant adding a subtenant. In this case, multiple employees of Reseller should be authorized to perform this operation. They will need some information about their new subtenant before starting the add-process, like what type of authentication/authorization they would like (outsourced or federated), who will be responsible for the administration at subtenant side and which tasks they would like the reseller to perform. Because Reseller is a reseller in this hierarchy, it should have the proper access rights to view all documents of Large Bank Leasing. A clear warning that this is implied due to the tenant

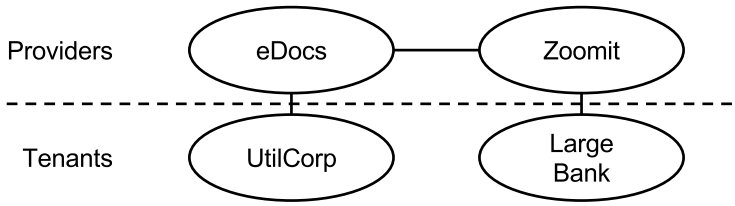

Figure 5: Zoomit scenario. Note: the relationship between eDocs and Large Bank as described in previous sections is not regarded any more.

hierarchy should be given in this process.

2.4 Zoomit

As a third extension, we can take into account Zoomit (see Figure 5). Zoomit is a third party application that allows bank customers to receive and manage invoices digitally (note: we simplify here by only regarding invoices; in reality, Zoomit can also handle other documents such as pay checks). eDocs also supports transferring documents to Zoomit. Zoomit mainly focuses on being a central platform for the large number of private bank customers and operates from the point of view of the bank customer (i.e., it centralizes document management for the customer, not for the sending organizations). Grouping all documents of a customer is done based on its bank account number.

For example, the energy corporation and utility company UtilCorp can send an invoice to Tom, one of its customers. UtilCorp uses the services of eDocs and sends the invoice using snail mail to Tom's postal address or using e-mail to Tom's e-mail address (both are given by Tom when registering with UtilCorp). Tom receives the invoice and pays it, after which UtilCorp knows Tom's bank account number (using the structured comment in the Tom's payment as given by UtilCorp in its invoice). For the next invoice, UtilCorp also provides Tom's bank account number when sending the document to eDocs. eDocs can use this number to transmit the invoice to Zoomit. Zoomit then checks whether Tom can and wants to receive the invoice using Zoomit. Zoomit determines that the bank account number belongs to Large Bank, which uses Zoomit. Zoomit then checks whether Tom has agreed to receive invoices from UtilCorp using Zoomit. For the first invoice, this will never be the case, after which Zoomit replies to eDocs it cannot deliver the invoice and eDocs delivers it to Tom's e-mail or postal address. However, Zoomit does register it can deliver invoices from UtilCorp to Tom and the next time Tom visits the online interface of Large Bank, it will show a message requesting to receive invoices from UtilCorp digitally using Zoomit. Assuming Tom agrees, eDocs will deliver the next invoice from UtilCorp solely to Zoomit. When another company, e.g., ICTProvider, sends an invoice to Tom, eDocs knows Tom has received documents using Zoomit before (e.g., according to Tom's postal address or because he has registered as a local user to eDocs) and can also try to do this for the ICTProvider invoice.

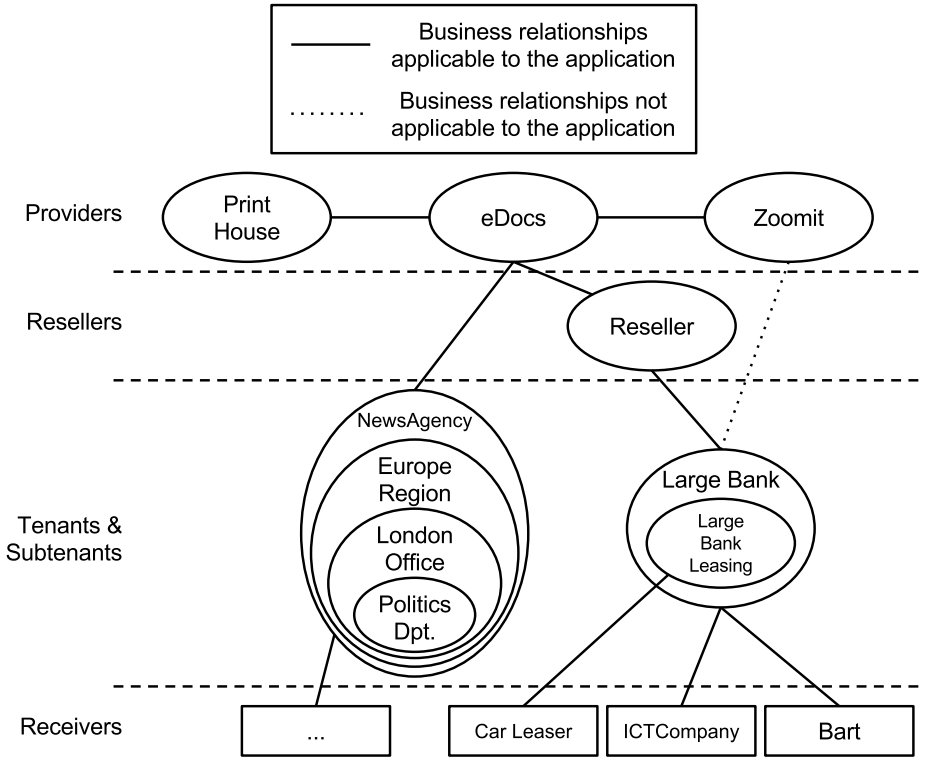

Figure 6: Complete overview of the scenario in terms of business relationships.

2.5 The whole picture

As a summary, we here describe the complete set-up as gradually built up in the previous sections (see Figure 6). eDocs is the provider of the application central to this document. The application allows customers (tenants) to send documents to their respective customers in various ways of transport. In order to print documents and send them using snail mail, eDocs cooperates with one or more Print Houses located around the world, of which this document only takes into account a single Print House. eDocs also cooperates with Zoomit, an application which specializes in delivering digital invoices to private bank account owners.

Examples of tenants of the application are NewsAgency and Large Bank. NewsAgency is a direct customer of eDocs; Large Bank is a customer of the reseller Reseller, which adds functionality to the base application such as certified storage. Notice that Large Bank employs the Zoomit application and therefore is a customer of Zoomit. However, this business relationship is not applicable for this application.

Each tenant can have one or more sub-organizations which also use the application and thus become subtenants, such as Large Bank Leasing for Large Bank.

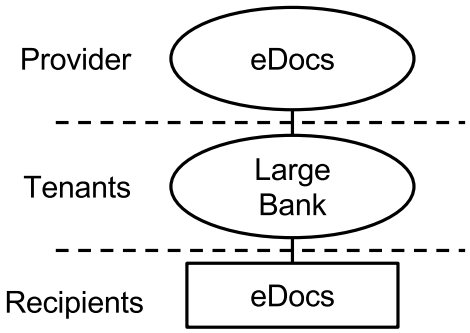

Figure 7: Illustration of a first case of a conflict of interest for the provider.

NewsAgency shows an extensive hierarchy of subtenants can arise because of the segregation of the organization in regions, offices per region, departments per office etc. Each tenant uses the application to send documents to its customers, i.e., the Receivers. For example, Large Bank sends documents to companies such as ICTProvider and private customers such as Bart; Large Bank Leasing sends documents to companies such as Car Leaser. We do not explicitly take into account the customers of NewsAgency in this document.

2.6 Remark: Provider conflict of interest 1

A possible conflict of interest exists in case the provider (or a reseller in the scenario above) also is a customer of a tenant, e.g., eDocs also is a customer of Large Bank (see Figure 7). In this case, the documents handled by the application can also contain information relevant to the provider, such as documents describing pricing changes etc. Because of this conflict of interest, a subtenant should be able to revoke access rights to certain documents for which it has ownership and the provider (or a reseller, in general a supertenant) has certain access rights to. This should cover all possible rights that can apply in this relation.

2.7 Remark: Provider conflict of interest 2

Another possible conflict of interest arises when a provider (or a reseller) manages documents of two related organizations, e.g., two organizations that are active in the same branch of industry (see Figure 8). For example, if eDocs (or a reseller such as Reseller) manages the documents of both Large Bank and Other Bank, the combination of sensitive documents of both organizations can provide inside information that can be in conflict with local regulation which might even be illegal. Contracts with customers like Large Bank en Other Bank actually cover these situations in practice. Thus, it should not be possible for an employee of eDocs to access documents of both organizations and different employees should be assigned management rights to the parties to mitigate the

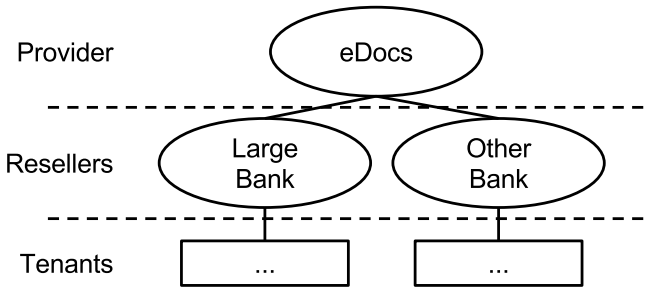

Figure 8: Illustration of a second case of a conflict of interest for the provider.

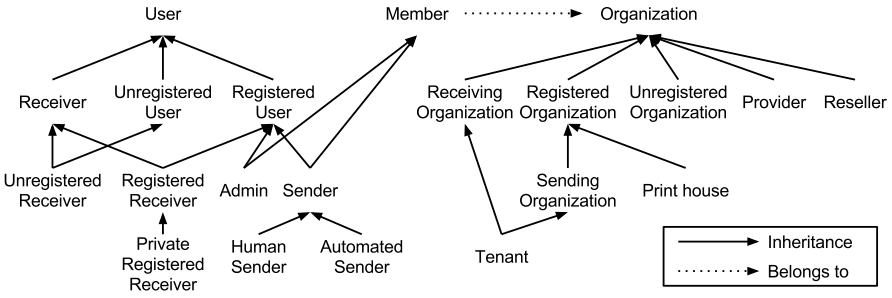

Figure 9: Actor and organization hierarchy

possible conflict. Currently, employees are instructed to adhere to these rules, but they are not enforced using access control. The latter would also allow to limit access to the document more fine-grainedly, for example allowing employees to only see meta-data of the documents and denying access to the contents.

Similar situations arise with specific types of documents. For example, eDocs handles both the pay checks, insurance documents and holiday documents. Employees of eDocs (e.g., the helpdesk) should clearly not be allowed to view all three types of documents of the same person.

3 Functional requirements

Following the illustrative scenario of the previous sections, this section describes the eDocs application more formally in the form of use cases and non-functional requirements. The use cases describe the actions which should be controlled by access control, the non-functional requirements apply to the access control sub-system as well. For clarity, we start by giving an overview of all actors and organizations involved in the eDocs application.

3.1 Actors

Figure 9 shows the Actor Hierarchy used in describing the functional requirements. Since organizations are an important concept in this case study, we explicitly model them and the relationship between the Actors and the Organizations. For readability reasons, we do not graphically show the relationship between the actor hierarchy and the organization hierarchy in the figure, but only describe it in the text below. Note that although an organization technically is not an actor, the term can be used as an abstraction in use cases. For example, the phrase "The Primary Actors sends a document to organization X." means the document will be received by a member of organization X.

Actors

1. **User**: general user of the application.
2. **Registered User**: User that is registered in the application.
3. **Unregistered User**: User that is not registered in the application.
4. **Receiver**: User that receives documents using the application. A Receiver can be a member of an organization (registered or unregistered) and can be private.
5. **Sender**: Registered User that sends documents using the application. Every Sender is a member of a sending organization.
6. **Human Sender**: Sender which uses the application through a GUI.
7. **Automated Sender**: Sender which uses the application using automated interfaces.
8. **Registered Receiver**: a Receiver that is registered in the application, either as Private Registered Receiver or as Member of a Registered Organization.
9. **Private Registered Receiver**: a Registered Receiver that is not a Member of a Registered Organization. In Dutch: "een particulier".
10. **Unregistered Receiver**: Receiver that is not registered in the application.
11. **Admin**: Registered User that manages the application for a certain Organizations.

Organizations

1. **Organization**: group of people (i.e., the Members of the Organization), e.g., a company.
2. **Member**: member of an Organization.

3. **Receiving Organization**: Organization that receives documents using the application. At least one member of a Receiving Organization is a Receiver, either Registered Receiver or Unregistered Receiver.

4. **Registered Organization**: Organization which is registered in the application. Members of a Registered Organization can be Registered Users or Unregistered Users.

5. **Unregistered Organization**: Organization which is not registered in the application. Members of an Unregistered Organization can only be Receivers. If they are Registered Receivers, they act as private Registered Receivers and not as Member of the Organization.

6. **Provider**: the Organization that provides the application, i.e., eDocs. Members of the provider can be Receivers, Senders and Provider Admins.

7. **Reseller**: Organization that resells the application to other organizations (Tenants). Members of the Reseller can be Receivers, Senders and Reseller Admins.

8. **Sending Organization**: Organization that sends documents using the application. Every Sending Organization is a Registered Organization. Members of a Sending Organization can be Senders.

9. **Tenant**: an Organization which rents access to the application from the Provider. The Members of a Tenant (or at least a part of) are Users of the application. A Tenant is a Receiving Organization and a Sending Organization and therefore has to be Registered Organization. Members of a Tenant can be Senders, Receivers and Tenant Admins.

10. **Print House**: a Registered Organization which allows to print digital documents. Documents are sent to or fetched by automated services acting as the organization as a whole, the Members of a Print House are not explicitly taken into account.

3.2 Use Cases

These use cases describe only a part of the whole system of eDocs. For scoping reasons, we ignore:

- Uploading raw data: In case of raw data, the raw data is first transformed into a document by the application by applying the appropriate templates and then sent to the receiver similarly to uploaded documents. This transformation step does not influence access control requirements. Therefore, we abstract from it by only regarding sending documents per se. As a consequence, we also do not incorporate use cases regarding template management.

- E-mail attachment or URL: There are two options for sending a document to a Receiver using e-mail: either attach the document to the e-mail or send a URL (with a unique identifier) where the document can be downloaded. These options do not influence access control: even in case of the URL, the website visit is anonymous and apart from the unique identifier in the URL, access control is not possible. Therefore, we abstract from both by ignoring the type of e-mail.
- Printing, scanning or archiving own documents: We only regard sending a document to a destination.
- Manual actions in the document processing workflow: At some points in the document processing workflow, it can be the case that manual intervention of an employee is required. For example, invoice can automatically be generated at the end of the month, but an employee has to confirm before sending them. In this document, we only regard manually sending documents, which can be seen as an abstraction of the confirmation.

3.2.1 UC1: Log in

- Summary: A Registered User wants to use the application and logs in by providing authentication credentials.
- Primary actor: Registered User
- Preconditions:
- Postconditions:
  - The Primary Actor is authenticated and can use the application.
- Main scenario:
  1. The Primary Actor indicates he wants to log in to the application.
  2. The Primary Actor provides authentication credentials, e.g., username and password.
  3. The system checks the provided authentication credentials.
  4. The system confirms successful authentication to the Primary Actor and logs him in, e.g., by setting a cookie.
- Alternative scenario:

  4b. The provided credentials are incorrect. The system notifies the Primary Actor of this. Restart from step 2.

- Remark: This use case does not take into account which method of authentication is used (cfr. step 2 and 3). For example, authentication can be done locally with eDocs or with a third party using a technique such as username-password, security tokes, IP-based etc.

3.2.2 UC2: Log out

- Summary: A logged in Registered User signs out of the application
- Primary actor: Registered User
- Preconditions:
  - The Primary Actor is logged in.
- Postconditions:
  - The Primary Actor is logged out and cannot use the application any more without logging in again.
- Main scenario:
  1. The Primary Actor indicates he wants to log out of the application.
  2. The system logs the primary user out.
  3. The system indicates success to the primary user.
- Remark: The use case does not take into account which method of authentication is used (cfr. step 2 and 3). This can be done to eDocs or to a federated party, according to what is supported by the system.

3.2.3 UC3: Send document

- Summary: A Sender provides a document, a destination and a means of transport, after which the system delivers the document to the destination using the selected means of transport.
- Primary actor: Sender
- Abstract use case: implemented by UC4: Send document to postal address, UC5: Send document to e-mail address, UC6: Send document to registered receiver, UC7: Send document to Zoomit
- Remarks:
  - The way of providing the document, the preferred means of transport and the destination depends on the type of Sender:
    - A Human Sender provides these items using a GUI.
    - An Automated Sender provides these items using an automated interface, e.g., a web service.
  - The way of authenticating depends on the type of Sender:
    - A Human Sender authenticates using human authentication mechanisms, e.g., username-password, client certificate, token, multi-factor authentication etc.

    - An Automated Sender authentications using automated authentication mechanisms, e.g., certificates.

- Preconditions:
  - The Sender is authenticated.
- Postconditions:
  - The provided document is sent to the destination using the selected means of transport.
- Main scenario:
  1. The Primary Actor provides a document (note: we abstract from raw data, see "we ignore" at the start of this section), selects a means of transport (e.g., snail mail, e-mail, Zoomit or account) and provides a destination (depends on the means of transport, e.g., postal address, e-mail address, Zoomit id or account name).
  2. The system delivers the provided document to the provided destination using the selected means of transport.

3.2.4 UC4: Send document to postal address

- Summary: A Sender provides a document and a postal address, the document is printed and the document is delivered to the postal address by snail mail.
- Extends: UC3: Send document
- Primary actor: Sender
- Preconditions:
  - The Sender is authenticated.
- Postconditions:
  - The provided document is printed and delivered to the provided postal address by snail mail.
- Main scenario:
  1. The Primary Actor indicates he wants to send a document using snail mail and provides a document, preferred Print House and a postal address.
  2. The system prints the document (can be done by sending the document to the specified Print House, Include UC19: Deliver documents to Print House) and sends it to the provided address.

- Alternative scenario:

  1b. The Primary Actor does not provide a preferred Print House.

  2b. The system sends the document to the Print House (selected optimally depending on the destination address), which prints the document (Include UC19: Deliver documents to Print House) and sends it to the provided address.

3.2.5 UC5: Send document to e-mail address

- Summary: A Sender provides a document and an e-mail address and the document is delivered to the e-mail address by e-mail.
- Extends: UC3: Send document
- Primary actor: Sender
- Preconditions:
  - The Primary Actor is authenticated.
- Postconditions:
  - The Receiver has received the document.
- Main scenario:

  1. The system sends the document to the provided e-mail address (note: we abstract from the type of e-mail, see above).
  2. The Primary Actor indicates he wants to send a document using e-mail and provides a document and an e-mail address.

- Alternative path:

  2b. The system notices that the e-mail address belongs to a registered Receiver and adds the document to the Receiver account. Include UC6: Send document to registered receiver (last steps).

3.2.6 UC6: Send document to Registered Receiver

- Summary: A Sender provides a document and an account name and the document is delivered to the account.
- Extends: UC3: Send document
- Primary actor: Sender
- Preconditions:
  - The Primary Actor is authenticated.

- Postconditions:
  - The Receiver has received the document.
- Main scenario:
  1. The Primary Actor indicates he wants to send a document to a Registered Receiver, provides the destination account name and a document.
  2. The system adds the document to the Receiver account.
  3. The system sends an e-mail notification to the owner of the account stating a document has been received.
- Alternative scenario:

  1b. In case the Sender is a Human Sender, he can select the destination account name from his/her address book in the application.

  3b. In case the receiving account is an organization, the e-mail notification is sent to the designated Receiver as configured by the organization.

3.2.7 UC7: Send invoice using Zoomit

- Summary: A Sender provides an invoice and a Zoomit id and the document is delivered to the corresponding Zoomit account.
- Extends: UC3: Send document
- Primary actor: Sender
- Preconditions:
  - The Primary Actor is authenticated.
- Postconditions:
  - The Receiver has received the document.
- Main scenario:
  1. The Primary Actor indicates he wants to send an invoice to a Zoomit account, provides the destination account id and another address (postal address or e-mail address) and an invoice.
  2. The system sends the document to Zoomit.
  3. The Receiver corresponding to the Zoomit account belongs to a bank that uses Zoomit and has agreed to receive invoices from this Sender via Zoomit. Zoomit adds the document to the destination account and responds that the invoice can and has been delivered.

- Alternative scenario:

  3b. The Receiver corresponding to the Zoomit account does not belong to a bank that uses Zoomit or has not agreed to receive invoices from this Sender via Zoomit. Zoomit responds that it cannot deliver the invoice.

  4b. The system delivers the invoice to the other provided address of the Receiver. Depending on the provided address, Include UC4: Send document to postal address, UC5: Send document to registered receiver or UC6: Send document to e-mail address.

3.2.8 UC8: View received digital document (Unregistered Receiver)

- Summary: An Unregistered Receiver has received an e-mail containing a document and downloads it.
- Remark: There are two options for sending a document to a Receiver using e-mail: either attach the document to the e-mail or send a URL (with a unique identifier) where the document can be downloaded. As stated above, we abstract from this here.
- Primary actor: Unregistered Receiver
- Preconditions:
  - The system has sent a document to the Primary Actor by e-mail.
- Postconditions:
  - The Primary Actor has downloaded the document.
- Main scenario:
  1. The Primary Actor receives the e-mail in his e-mail client.
  2. The Primary Actor downloads the document in the appropriate way.
  3. The Primary Actor optionally chooses to register as described in the e-mail. Include UC14: Register

3.2.9 UC9: View received digital document (Registered Receiver)

- Summary: A Registered Receiver has received an e-mail containing a notification that a new document has been received and downloads it.
- Primary actor: Registered Receiver
- Preconditions:
  - The system has sent an e-mail to the Primary Actor containing a notification that a new document has been received.

- The Primary Actor is logged in.

- Postconditions:
  - The Primary Actor has downloaded the document.

- Main scenario:
  1. The Primary Actor receives the e-mail in his e-mail client.
  2. The Primary Actor visits the application to read the received document, e.g., following the URL given in the e-mail or visiting the application directly.
  3. The Primary Actor downloads the received document.

3.2.10 UC10: Get overview of received documents

- Summary: A Registered Receiver requests an overview of all received documents.

- Primary actor: Registered Receiver

- Preconditions:
  - The Primary Actor is logged in.

- Postconditions:
  - The Primary Actor has been shown an overview of all received documents.

- Main scenario:
  1. The system shows the Primary Actor an overview of all the documents he/she received, e.g., a table or list with a summary of each document.
  2. The Primary Actor requests an overview of all received documents.

- Alternative scenario:
  3. The Primary Actor selects a document from the overview to view its details, Include UC9: View received digital document (Registered Receiver)

3.2.11 UC11: View sent document

- Summary: A Sender requests a sent document and its details, e.g., status.
- Primary actor: Sender
- Preconditions:
  - The Primary Actor is logged in.
- Postconditions:
  - The Primary Actor has been shown a sent document and its meta-information, e.g., status.
- Main scenario:
  1. The Primary Actor requests a sent documents.
  2. The system shows the Primary Actor the content and meta-information of the document he/she sent, e.g., the addressee, the date of sending, the status (e.g., "Sent to Zoomit", "Printed", "Being printed" etc), etc.

3.2.12 UC12: Get overview of sent documents

- Summary: A Registered Sender requests an overview of all received documents.
- Primary actor: Registered Sender
- Preconditions:
  - The Primary Actor is logged in.
- Postconditions:
  - The Primary Actor has been shown an overview of all received documents.
- Main scenario:
  1. The Primary Actor requests an overview of all received documents.
  2. The system shows the Primary Actor an overview of all the documents he/she received, e.g., a table or list with a summary of each document.
- Alternative scenario:
  3. The Primary Actor selects a document from the overview to view its details, Include UC11: View sent document

3.2.13 UC13: Search documents

- Summary: A User searches amongst its document.
- Primary actor: Sender, Reseller employee or Receiver
- Preconditions:
  - The Primary Actor is authenticated.
- Postconditions:
  - The documents that match the specified search are listed. All documents of the Primary Actor are listed if no search term is specified.
- Main scenario:
  1. The Primary Actor indicates he/she wants to search for documents.
  2. The system shows a form to the Primary Actor in which he/she can specify a search query.
  3. The Primary Actor specifies the search query, e.g., a term that the document should contain or a date, type, category, tag or addressee data that should match with the document.
  4. The system searches all documents of the Primary Actor:
     (a) If the Primary Actor is a Sender, all sent messages are searched.
     (b) If the Primary Actor is a Receiver, all received messages are searched.
     (c) If the Primary Actor is a Reseller employee, all documents of tenants for which the Primary Actor is responsible are searched.
  5. The application shows all documents that match the search query.
- Alternative scenario:

  3b. The Primary Actor does not specify a search term.

  4b. All documents for which the Primary Actor has proper access rights are shown.

3.2.14 UC14: Register

- Summary: an Unregistered Receiver chooses to register in the system and becomes a registered party (either the Receiver becomes a private Registered Receiver or the Organization to which the Receiver belongs becomes a Registered Organization).
- Primary actor: Unregistered Receiver
- Abstract use case: implemented by UC14: Register (as Private Receiver) and UC15: Register (as Receiving Organization)

3.2.15 UC15: Register (as Private Receiver)

- Summary: a Private Unregistered Receiver chooses to register in the system and becomes a private Registered Receiver.
- Implements: UC14: Register
- Primary actor: private Unregistered Receiver
- Preconditions:
  - The Primary Actor has received a document from the system by e-mail, containing information on how to register.
- Postconditions:
  - The Primary Actor is now a Registered Receiver.
  - In the future, documents sent to the Primary Actor will be attached to his account.
  - Documents previously sent to the e-mail address of the Primary Actor are attached to its new account.
- Main scenario:
  1. The Primary Actor visits the registration page given in the e-mail. This URL identifies the Primary Actor to the system.
  2. The system asks the actor to provide personal information (e.g., his name) and a password. His e-mail address is already filled in.
  3. The actor provides the necessary information.
  4. The system creates an account for the Primary Actor.
  5. The system confirms success to the Primary Actor.
- Exception scenario:

  4b. The Primary Actor provides information which contains errors.

  5b. The system notifies Primary Actor, go to Step 3.

3.2.16 UC16: Register (as Receiving Organization)

- Summary: an Unregistered Receiver chooses to register its Organization in the system and becomes Member of a Registered Organization.
- Specifies: UC14: Register
- Primary actor: Member of Unregistered Receiving Organization
- Preconditions:

  - The organization has received a document from the system by e-mail. The Primary Actor is responsible for handling this document and receives it for the organization.

- Postconditions:
  - The organization is now a Registered Organization and Receiver.
  - The Primary Actor is assigned as Organization Admin of the Organization.
  - The Primary Actor can now manage the Organization's internal users and document routing preferences.
  - In the future, documents sent to thee-mail address of the organization will be attached to the account instead.
  - Documents previously sent to the e-mail address of the organization are attached to its new account.

- Main scenario:
  1. The Primary Actor visits the registration page given in the e-mail. This URL identifies the organization to the system.
  2. The system asks the actor to provide some information about the organization (e.g., its name).
  3. The system registers the new organization, links the e-mail address to it and assigns the Primary Actor as organization administrator.
  4. The system acknowledges success to the Primary Actor.

3.2.17 UC17: Update organization configuration

- Summary: an Organization Admin updates the configuration of its Organization, e.g., which e-mail addresses are valid, how these addresses are routed, which users are in the organization, which privileges they have etc.

- Remark: this is a high-level use case that bundles multiple administrative use cases for brevity.

- Primary actor: Organization Admin

- Preconditions:
  - The Primary Actor is logged in.

- Postconditions:
  - The organization configuration is updated.

- Main scenario: generic: choose option, input new value.

3.2.18 UC18: Add print house

- Summary: The eDocs admin or tenant admin provides a new print house to send print jobs to.
- Primary actor: eDocs Admin
- Preconditions:
  - The Primary Actor is authenticated.
  - A new print house has made a collaboration agreement with eDocs.
- Postconditions:
  - The new print house will be available to send print jobs to.
- Main scenario:
  1. The eDocs superuser specifies a new print house that can be used to print the documents processed by the application. This includes address, contact information and the preferred method to send the documents to them.
  2. The print house is added to the possible agents to transfer documents to for printing.
- Technology & Data Variations List:
  - If the print house accesses the documents by means of a pull-operation, an authentication of the print house will be required to access the documents. An asymmetric encryption scheme can be used to retain confidentiality of the sent documents to the print house. The public key will have to be given when registering the print house and the application should store it locally.

3.2.19 UC19: Deliver documents to Print House

- Summary: Documents to be printed are delivered to the corresponding Print House.
- Primary actor: Print House
- Preconditions:
  - There are documents for the Print House to be printed.
- Postconditions:
  - The documents are delivered to the Print House for printing.

- Main scenario:

    1. The system determines that one or more documents should be printed by the Print House.
    2. The system connects to the Print House.
    3. The system and the Print House authenticate mutually.
    4. The system sends the documents to the Print House.
    5. The Print House acknowledges the receipt of the documents.
    6. The system closes the connection to the Print House.

- Alternative scenario: The Print House wants to poll for new document and pull them. Similar to the above.

4 Non-functional requirements

This section describes important non-functional requirements for the system. Again, we focus on the parts of the system related to access control.

- Scalability: The application should be able to handle a large number of documents (exceeding hundreds of millions) which are sent by a large number of tenants and their respective subtenants, leading to complex and large tenant hierarchies. Moreover, each of the partner organizations can have a lot of employees using the application. The access control system should therefore scale to large numbers of documents, tenants, users and requests.

- Performance: The latency of a request to the application should be minimized while maximizing the throughput of requests, especially at peak moments (e.g. the end of the month, Christmas shopping period, etc.). Since the access control system is involved in every request to the application, these requirements also hold for the access control system.

- Availability: Because the involvement of the access control subsystem in almost every action performed on the application, it is of paramount importance that the subsystem remains highly available in all circumstances. Failure of the subsystem would imply that the whole application is unusable for almost every user.

- Maintainability: The access control system should be able to cope with large numbers of documents shared amongst large numbers of senders and receivers over many tenants and subtenants. In the first place, the access control system should allow self-management for every tenant and subtenant. In the second place, the access control model should provide scalable primitives for expressing these complex requirements.

- Security: The system should be secure. We mainly focus on confidentiality and integrity for all sensitive information of any party in the system, leading to access control requirements. The main example of such sensitive information are the documents handled by the application, but the access control policies and the data they use can be confidential as well.

5 Glossary

This section describes some of the terminology that is used in this document to refer to functionality of the application and when describing different scenarios.

- **Conflicting documents**: Two documents are conflicting if they cannot be read by the same person, e.g., an insurance document, a holiday document and a paycheck of the same person or two documents of organizations in the same branch of industry, such as Large Bank and Other Bank in the banking branch.
- **Destination**: a Receiver or a Receiving Organization.
- **Direct tenant**: a direct tenant uses the application by directly renting access to it from the provider.
- **Document**: a document consists of raw data styled in a certain template. The raw data can be converted to a document by eDocs or the document can be uploaded to eDocs as a whole. A document can be sent to a certain person or can be solely stored for future use.
- **eDocs**: the company that provides the application.
- **Indirect tenant**: an indirect tenants uses the application by having a business relationship with another (direct or indirect) tenant. E.g., an organization can rent access to the application from a reseller, an organization can be the customer of a tenant and thereby receive its invoices through the application.
- **Invoice**: a special type of document that contains a bill to be paid.
- **Meta-information of a document**: all details of the document except for its content. Example meta-information is the date the document was sent or received, the sender or receiver, its status (e.g., "Being printed") etc.
- **Output subsystem**: the subsystem of the application of eDocs which handles sending produced documents.
- **Organization**: an organization is a group of users, possibly involved in the application. Special types of organizations are the provider, the tenants, subtenants etc. The expression "an organization uses the application" means the members of the organization use the application and thereby become users of the application.

- **Party**: general term used for an organization or end-user involved in the application.
- **Provider**: the organization that provides the application and manages it. There is only one provider in the application, i.e., eDocs.
- **Raw data**: raw data can be uploaded by a user and will be converted to a document by the application.
- **Sender group**: all users of the application within the same tenant organization which practice the same role as a specified sender.
- **Sending a document**: sending a document involves a sender providing the document and the destination and uploading it, after which the application delivers it to the destination.
- **Tenant**: a organization that rents access to the application from the provider. There are multiple tenants in the application.
- **Zoomit**: Zoomit is a third party application that allows bank customers to receive and manage invoices digitally. eDocs also supports transferring documents to Zoomit. In thid document, Zoomit is only used to handle invoices; in reality, Zoomit can also handle other documents such as pay checks.

6 Policies from the scenario

Following the description of the SaaS application, this section zooms in on the resulting access control requirements. More specifically, this section describes example policies for each organization in the scenario described above. As mentioned before, we are interested in applications which have to cope with complex business relationships amongst the parties that use it. The application should allow each organization to express these access control policies, thereby allowing self-management. For each party we briefly summarize its role in the scenario and list applicable policies.

6.1 General for the application

- An Unregistered Sender can only view documents sent to him/herself.
- Note: Each document is identified by a unique URL which can be used as an access token.

6.2 eDocs

eDocs is the provider of the application. eDocs should therefore limit the tenant access to the application, for example based on their credit. Moreover, some employees of eDocs also use the application themselves. Although eDocs can

use the application itself to send or receive documents, we here focus on the helpdesk and application admins.

Applicable policies

- Provider policies:
  - A Member of a Tenant can only send a document if the credit of that tenant is sufficient.
  - Every action done by a Member of a Tenant is billed to that Tenant.
  - Every action done by a User is logged.
- Helpdesk:
  - Members of the helpdesk can search every document in the application.
  - Members of the helpdesk can see meta-information of every document in the application.
  - Members of the helpdesk can only read the content of documents belonging to tenants for which they are assigned responsible.
  - Members of the helpdesk can only manage tenant configurations (e.g., create users or groups, update document meta-data etc.) for tenants for which they are assigned responsible[1].
  - Delegation:
    - Any member of the helpdesk assigned to a certain tenant can delegate the right to read documents of that tenant and manage the tenant's configuration to another member of the helpdesk for a maximum time of 4 weeks.
    - A member of the helpdesk assigned to a certain tenant can only delegate the right to read documents of that tenant and manage the tenant's configuration to another member of the helpdesk for a maximum time of 12 weeks in a year.
  - Conflicts of interest:
    - A member of the helpdesk can only read the meta-data (i.e., not the contents) of a document if he or she has read the contents of a conflicting document.
  - Logging:
    - Each action done by a member of the helpdesk is logged.

[1] Notice that this responsible list can be very specific per employee. For example, although a responsible for CustomerA switches teams, it can be preferred to keep that employee responsible for CustomerA because of past experience. As a result, the responsibilities are often not clearly structured (e.g., as a tree) in practice.

- Application admins:
  - Access rights to documents can never be revoked by subtenants. Access to documents should be strictly monitored and the monitoring data should be accessible by the tenants.
  - Application admins can create new tenants, manage tenant configurations etc.
  - Conflicts of interest:
    - An application admin can only read the meta-data (i.e., not the contents) of a document if he or she has read the contents of a conflicting document.
- Confidential documents:
  - No one can read the content of documents labeled confidential by a member of the owning tenant.

6.3 Large Bank

Large Bank is a large company, consisting of thousands of Senders and hundreds of thousands of Receivers. Large Bank uses the application to send documents both internally and to external customers, both to organizations and private customers, both manually or automatically. Large Bank itself is structured in sub-organizations (e.g., Large Bank Leasing) and departments (e.g. Sales, ICT, Audit, etc), which in turn consists of offices (e.g. secretary). Next to the main organization, Large Bank also has a large number of regional (local) bank offices.

Applicable policies

- General:
  - Every authenticated user can send documents and receive.
  - Supervisees:
    - A supervisor can read documents sent by its supervisees.
    - A supervisee cannot read documents sent by its supervisors.
  - Projects:
    - A project member can read all sent documents regarding the project.
  - Subtenants:
    - Members of a subtenant can only send documents to customers of that subtenant.
- Invoices:

  - Only members of the sales department can send invoices.
  - Only members of the sales department can view or search invoices.
  - Every member of the sales department can view or search invoices sent by a member of the sales department.

- Banking notes:
  - Only automated services of the ICT department can send banking notes.
  - Only members of the ICT department responsible for banking notes can view the status of a sent banking note.
  - Every member of the ICT department responsible for banking notes can view the status of a sent banking note.

- Paychecks:
  - Only employees which are responsible for payrolling can send paychecks and only to members of their department.
  - Employees which are responsible for payrolling can only send paychecks between the 20th and 25th of each month.
  - Only employees which are responsible for payrolling and receivers of the paychecks can read paycheck documents.
  - Every access to a paycheck should be logged.

- Sales offers:
  - Only members of the sales department can send sales offers.
  - Only senior members of the sales department can send sales offers to all customers at once.
  - Sales offers regarding insurances can only be sent to insurance customers.
  - Sales offers regarding savings accounts can only be sent to customers who own a savings account.
  - Sales offers can only be sent on weekdays between 07h00 and 19h00.

- Internal communication with local bank offices:
  - Only the bank office manager responsible for a certain bank office can send documents to that office.
  - Only senior bank office managers can send documents to all bank offices at once.

- Organization management:

  - Only members of the HR department can create users.
  - Only members of the IT departments which are part of the Senior Management can create subtenants in the application.
  - Users can only be created on weekdays between 08h00 and 02h00.

- Audits:

  - The Large Bank Audit department can read any document sent by any member of Large Bank, except for the paychecks and banking notes, or any other document marked in its meta-data to contain personal information of the customer. Every read action is logged.

6.3.1 Large Bank Leasing

Large Bank Leasing is the part of Large Bank that focuses on leasing contracts. For business and management reasons, Large Bank Leasing is not a department but a sub-organization of Large Bank and therefore a subtenant of eDocs. Large Bank Leasing itself is organized as offices and uses the application to send documents to its customers, such as Car Leaser.

Applicable policies

- General:

  - Members of Large Bank Leasing can only send documents to customers of Large Bank Leasing.

- Traffic fines:

  - Only members of the customer care office can view traffic fines.

- Invoices:

  - Only users of the sales office can send invoices.
  - Only members of the Customer Care Office can manually bill a customer, e.g., billing a traffic fine.

6.3.2 Local bank offices

Next to the main office, Large Bank also has a large number of regional (local) bank offices. These offices can send documents to the customers of that office. Bank offices both provide typical banking products such as banking accounts as insurances.

Applicable policies

- Only the secretary and the office director of a bank office can read documents sent to the bank office.
- Members of a bank office can only send documents to external customers whose main office is that bank office.
- Insurance agents of a bank office can only send documents to insurance customers of that bank office.

6.4 Car Leaser

Applicable policies

- Invoices:
  - Any member of the Accounting department can receive and read invoices.
  - Only members of the Accounting department can receive and read invoices, unless every member of the Accounting department is set to be unavailable (e.g., on Holiday, Leave, ...). In that case, the members of the Secretary department are assigned to receive and read these invoices.

6.5 ICTProvider

ICTProvider is a smaller company with respect to Large Bank and is organized as multiple offices. ICTProvider is not a tenant of eDocs: it is only involved in the application as registered Receiver and only uses the application to receive documents and manage its account. While ICTProvider can receive documents from multiple Senders, we only take into account Large Bank in this document.

Applicable policies

- Offices, e.g., sales office, secretary office, ICT support office etc.
  - Only members of a certain office can read documents sent to that office.
  - Any member of a certain office can read documents sent to that office.
  - Any member of a certain office can read all documents sent by any member of that office.
  - Note: Each office is identified by a separate e-mail address, e.g., sales@ictprovider.com, secretary@ictprovider.com, ict-helpdesk@ictprovider.com.
- Invoices:

  - Only members of the secretary can read invoices.

- User management:
  - Only members of the ICT group can create, delete or manage users or groups in the application.

6.6 NewsAgency

NewsAgency is a large, international news agency. The headquarters are mainly concerned with high-level and strategic management, auditing and financing. The rest of the organization is the organized by region: Europe, Asia, North-America, etc. Each region is then organized as departments and offices. Each of these levels should be supported as subtenants in the application. Moreover, each of these organizations is also a receiver, since documents can be sent internally.

Applicable policies

- Members of the Audit department can read all invoices, offers and contracts.
- All document read attempts by a member of the Audit department should be logged.
- Members of the Audit department can read all paychecks of all employees which are responsible for payrolling and their (indirect) supervisors.
- Only members of the IT department which are part of the Senior Management group and responsible for the software partner management can create subtenants.
- Members of the IT department which are responsible for user management can change the access rights to documents. This should be logged. Logs should be viewable by every member of the IT department and by members of the Audit department which belong to the IT Audit group.

6.6.1 Europe Region

The Europe Region is a sub-organization of NewsAgency. The organization focuses on news events and reporting that occur in Europe. The organization is responsible for allocating resources to make sure that profit as well as the news reporting capability is maximized. It uses the application of eDocs to send contracts to employees. To eDocs, it is a subtenant to NewsAgency.

Applicable policies for the Europe Region

- Subtenant creation and delegation:
  - Any Member of the IT department which is part of the Senior Management group and responsible for the software partner management can create subtenants.
  - Only Members of the IT department which are part of the Senior Management group and responsible for the software partner management can delegate the right to create new subtenants to its subtenants.
- Contracts:
  - Any member of the Human Resource department can send contracts to potential future employees.
  - Only members of the Human Resource department can send contracts to potential future employees.

6.6.2 London Office

The London Office is a sub-organization of the Europe Region which focuses on news reporting in the United Kingdom. Because it also makes use of the eDocs application, it is a subtenant to eDocs. The London Office uses the application to send contracts to freelancers and employees, and invoices to customers.

Applicable policies for the London Office

- Contracts:
  - Members of the Human Resources department can send contracts to freelance members which collaborate with a project X. They can only send these documents after two weeks before the start of X and not after one month after the end of X.
  - Only members of the Audit department, members of the Human Resources department and members which collaborate with a project X and are part of the Senior Management group can read the contracts send out for that project X (besides the Receiver).
- Invoices:
  - Any member of the sales department can send invoices.
  - Only members of the sales department can send invoices.
  - Any members of the sales department can read all invoices sent by the department.
  - Delegation:

    * Any member of the sales department can delegate the right to send invoices to a member of the Secretary for a duration of maximum 6 weeks. This action should be logged and would implicitly allow the secretary member to read the sent documents for this period.

6.7 Reseller

Resellers are tenants in the sense that they rent access to the application from eDocs, but differ from "normal" tenants in that they do not use the application for sending documents, but resell (or rather, re-rent) the application to other organizations. Resellers therefore require the same access control self-management as normal tenants, but also provide helpdesks and admins similar to eDocs and thus requires to be able to manage its tenants and their documents. We assume that sending out documents cannot be externalized to a reseller/provider.

Applicable policies

- Subtenant creation:
  - Any member of the Sales department can create a new subtenant.
  - Only members of the Sales department can create new subtenants.
  - Creation of a new subtenant is logged and can be monitored by members of the Sales department, Audit department, and Customer department. Also, members of the Management group and of the Helpdesk group can view these logs.
- Subtenant access management:
  - Any member of the Customer department can modify access rights of the documents sent by a subtenant which he/she has been assigned to if he/she is authorized by the reseller to do access management for the subtenant.
  - Any member of the Customer department can add properties to sent documents to help classify them for the subtenant.
  - Only members of the Customer department can modify access rights of the documents sent by a subtenant which it has been assigned to and classify the sent documents.
- Subtenant document sending:
  - Any member of the Customer department can send documents on behalf of the subtenant he/she has been assigned to if he/she is authorized by the reseller to perform the subtenant communication.
  - Only members of the Customer department can send documents on behalf of the subtenants.

- Subtenant document viewing:
  - Only members of the Customer department that is assigned to a subtenant can read the documents of that tenant.
  - Any member of the Customer department that is assigned to a subtenant can read the documents of that tenant, provided that he/she is not involved in a possible conflict of interest situation. This occurs when he/she is assigned to two organizations which are active in the same branch of industry.
- Invoices:
  - Any member of the Accounting department can send invoices to customers (subtenants).
- Conflicts of interest:
  - A member of the reseller can read the meta-data (i.e., not the contents) of a document if he or she has read the contents of a conflicting document.
- Confidential documents:
  - No one can read the content of documents labeled confidential by a member of the owning tenant.

6.8 Registered Private Receivers

Registered Private Receivers have the opportunity to get an overview of all received documents and can delegate access rights to other Private Registered Receivers.

Applicable policies

- Registered Private Receivers can only read documents.
- Registered Private Receivers can only read documents they received.
- Registered Private Receivers can allow another Registered Private Receiver to read a certain document.
- Registered Private Receivers can allow another Registered Private Receiver to read all his/her received documents.

7 Conclusion

Software-as-a-Service or SaaS is a maturing model for offering online applications with a growing interest from industry. While SaaS promises benefits for both users and providers of these applications, the challenge of manageable and effective access control in the presence of the multiple parties involved still hinders its widespread adoption. To address this challenge, the first step is to clarify the requirements for access control for SaaS. Therefore, we analyzed the SaaS application provided by the company eDocs as a realistic case study of such a SaaS application. We started from an illustrative scenario, further specified the application in use cases and non-functional requirements and finally deducted access control policies that apply to this application. Towards the future, this case study can be combined with others in order to identify general requirements for SaaS access control.

References

[1] P. Mell and T. Grance. The NIST definition of cloud computing. *National Institute of Standards and Technology*, 53(6):50, 2009.