Abstract

Organizations have grown over time and so has the number of software applications they use. Not only the number of applications but also the number of users that need access has grown. Suppliers and other partners also want to access resources from within the organization. For a couple of years there were no strict access rules, the people who had access to a computer could access all resources. Over the years the number of applications grew and companies started to realize that they had to protect their resources. That resulted in applications with their own authentication mechanism; an employee needed a username and password (identity) for every application. With the growing number of applications the number of usernames and passwords an employee had to remember also grew. The result was that the maintenance of all those identities became more complex. Users needed to remember all the identities. Administrators had to maintain all identities and the access rights belonging to those identities. Management could not really understand those access rights so they were unable to verify things such as privacy protection and they could not hold employees accountable for their tasks when the employees did something they were not allowed to.

Identity management can help to solve the problem above. The idea behind identity management is to centralize identity and access management. Instead of many applications with their own authentication and authorization mechanism identity management is centralized. The centralization can be constructed with a LDAP server which is a central place where the usernames and passwords are stored. That server can be used to authenticate and to define the access control.

The thesis consists of two parts a managerial part and a technical part. These are combined into one thesis but are mainly treated in separate chapters. In the thesis I have tried to find an answer to the following two managerial questions:

- What are the benefits for organizations when using identity management? Or in other words why should an organization opt for identity management?
- What are the considerations for organizations when using identity management? Or in other words, what should the organization do when introducing identity management?

At this point it seems that the problems that companies have with identities and access control can be easily solved with identity management. There are however two problems: companies do not realize the benefits of identity management and/or they implement identity management in a 'bad' way. The problem is that most companies cannot see direct value of identity management, the costs are spread across the company and it is hard to make them explicit. Reduction of costs should not be the (only) driver of identity management. There are more benefits such as improved security, user convenience and the ability to allow other organizations such as suppliers' access to specific resources of the company. However these benefits are unclear for many organizations and they do not implement identity management, or they implement it because it is required by law or legislation. When the management does not understand the clear benefits of identity management then the support from the top level of the company will be low. That will result in employees who will not be too enthusiastic. In the end that could result in identity management that is not well implemented and cannot realize all the benefits. As identity management becomes more and more important and organizations start to realize that it is not only a technical thing, it was interesting to see what the current developments are.

It seems that the organizations start to realize that identity management should involve management, administrators and users. They should work together to define policies,

processes and the technical implementation. There is no straightforward solution to introducing identity management. As identity management involves many aspects and is closely related to the organization's structure (for the access rights) and the organizations applications (for the authentications) it is very organization specific. But there are some guidelines and best practices that can be used to introduce identity management.

This thesis consists of two chapters that are mainly managerially orientated namely chapter **Error! Reference source not found.**: '**Error! Reference source not found.**'. This chapter explains the main drivers for an organization to spend time on identity management. There are quite some advantages of using identity management which are discussed in this chapter. Then chapter **Error! Reference source not found.** '**Error! Reference source not found.**' shows how it comes that some companies end up with 'bad' identity management. To try and give some guidance to companies to avoid 'bad' identity management the rest of the chapter is dedicated to treating the issues one should keep in mind when introducing identity management.

After the managerial part comes the technical part where I tried to find an answer to the following question:

- Is .NET or Java better suitable for authentication and authorization with an LDAP server?

Some organizations have a policy which describes the language to use; other organizations do not have a strict policy about the programming language. If there is no strict policy then it might be interesting to see if some language is better suited for identity management then another language. In this thesis the differences between Java and .NET are analyzed. The conclusion is that it is possible to implement identity management in both languages. The languages have some differences such as the available documentation, dependency on operating system and the level of abstraction but in the end they are both quite suitable. When choosing between the languages it is best to look at the expertise within the company and the configuration of the network. If there is more expertise in one language then that should be the language of choice. If you have mainly Microsoft products then .NET is probably the best choice and if that is not the case then Java might be the better choice. The question however is if it is practical to implement identity management from scratch or if it is better to use a standard package. That is because identity management can get quite complex and it has to communicate with all applications that you use within the organization. Building something that big might prove more costly in the end then buying a standard package and customizing it to your needs.