jamk.fi

Integration of a heterogeneous authentication environment to Active Directory

Saku-Matti Pietilä

YII13K
September 2015

Master´s Degree Programme in Information Technology (YAMK)
Technology and transport

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

Description

Author(s)	Type of publication	Date
Saku-Matti Pietilä	Master's thesis	29.9.2015
		Language of publication: English
	Number of pages: 81	Permission for web publication: X

Title of publication

Integration of a heterogeneous authentication environment to Active Directory

Degree programme

Master´s Degree Programme in Information Technology

Tutor(s)

Mika Rantonen
Jouni Huotari

Assigned by

Qvantel Finland Oy

Abstract

The thesis presents the implementation of a centralized authentication environment that was to integrate into the company infrastructure and the effect on administrative work was studied. The main goal was to integrate the new technology into company Active Directory to provide authentication and configuration management for Linux and Solaris servers. In addition, other goals for the study were the reduction on the work effort allocated to access management, enhancement in security, and improvement with environment expansion.

The original environment was studied to identify the problematic points and a plan was constructed for the implementation. The complexity of the infrastructure was acknowledged during the planning phase with the functional requirements presented by the users. The amount of downtimes caused by the project were minimized during the implementation.

The system proved to have major implications to day-to-day access management operations. In addition, installation of new servers and configuration management were improved. The cost savings on operational expenditure from access management in a short time span justify the investment.

It was concluded that the project was successful and fulfilled all the requirements although challenges were faced in many phases. Many problems occurring with the implementation of a new technology could be avoided with proper training before the actual deployment.

Keywords/tags (subjects)

Authentication, Linux, Solaris, Centrify, active directory, access management

Miscellaneous

JYVÄSKYLÄN AMMATTIKORKEAKOULU
JAMK UNIVERSITY OF APPLIED SCIENCES

Kuvailulehti

Tekijä(t)	Julkaisun laji	Päivämäärä
Saku-Matti Pietilä	Opinnäytetyö	29.9.2015
	Sivumäärä 81	Julkaisun kieli Englanti
		Verkkojulkaisulupa myönnetty: X

Työn nimi
Heterogenisen ympäristön autentikaation integrointi aktiivihakemistoon

Koulutusohjelma
Master´s Degree Programme in Information Technology

Työn ohjaaja(t)
Mika Rantonen
Jouni Huotari

Toimeksiantaja(t)
Qvantel Finland Oy

Tiivistelmä

Tässä opinnäytetyössä tuotetaan keskitetty autentikointiympäristö toimeksiantajan tietojärjestelmään ja tutkitaan sen vaikutusta hallinnoinnin kannalta. Päätavoitteena oli implementoida Active Directory teknologiaan yhdistetty integraatiokerros, johon oli liitettävissä Linux ja Solaris palvelimia sekä jalkauttaa järjestelmä tuotantoympäristöön. Tarkoituksena oli keventää pääsynhallintaan käytettyä työpanosta, parantaa tietoturvallisyytta sekä mahdollistaa ympäristön hallinnointi myös ympäristön kasvun aikana.

Ennen järjestelmän implementointia tutkittiin alkuperäisen ympäristön ongelmakohtia sekä laadittiin suunnitelma käyttöönotosta. Suunnittelussa huomioitiin järjestelmän kompleksinen rakenne sekä tuotantojärjestelmien ja käyttäjien asettamat vaatimukset. Tuotantokatkoksien määrä pyrittiin pitämään mahdollisimman pienenä koko implementaation ajan.

Käyttöönotetulla järjestelmällä todettiin olevan merkittävä vaikutus päivittäiseen pääsynhallintaan operatiivisessa työssä sekä järjestelmien käyttöönotto ja konfiguraationhallinta tehoistuivat. Yksinomaan käyttäjänhallinnan tehostuminen on vähentänyt operointikustannuksia ja investoinnille on saatu selkeä kate jo lyhyellä aikavälillä.

Kokonaisuutena projektia voitiin pitää onnistuneena ja määritellyt tavoitteet saavutettiin, vaikka haasteita kohdattiin useissa eri vaiheissa. Uuden teknologian käyttöönotossa olisi vältytty useilta ongelmilta, jos koulutukseen olisi panostettu enemmän.

Avainsanat (asiasanat)
autentikointi, Linux, Solaris, Centrify, aktiivihakemisto, käyttäjänhallinta

Muut tiedot

ACKNOWLEDGEMENTS

I want to express my gratitude to my tutors Mika Rantonen and Jouni Huotari for their support, guidance and motivation through the creation process. I also want to thank Qvantel for the interesting project that I could participate in. Last but not least, thank you my family, for supporting my studies during my work and outside working hours.

ACRONYMS

AD	Active Directory
ADAC	Active Directory Administrative Center
AD DS	Active Directory Domain Services
ADUC	Active Directory Users and Computers
ACL	Access Control List
BSS	Business Support Solutions
CAPEX	Capital expenditures
DC	Domain Controller
DNS	Domain Name System
GID	Group ID
IAM	Identity and Access Management
KDC	Key Distribution Center
LDAP	Lightweight Directory Access Protocol
MMC	Microsoft Management Console
NIS	Network Information Service
NTP	Network Time Protocol
OPEX	Operating expense
OU	Organizational Unit
PAM	Pluggable Authentication Module
PDCA	Plan Do Check Act, also known as Deming circle
RODC	Read-only Domain Controller
SSH	Secure Shell
SSL	Secure Sockets Layer
UI	User Interface
UID	User ID
ZPA	Zone Provisioning Agent

TABLE OF CONTENTS

FIGURES

TABLES

1 INTRODUCTION

1.1 Qvantel

> *"Qvantel improves business performance with a unique BSS offering allowing Service Providers to offer consumer and business touch points that simply work." (Qvantel, 2015)*

Qvantel is an international 20-year-old IT company that was founded in 1995 as Starnet Systems. Qvantel acquired Starnet Systems in 2008 and the name was changed to its current form. Now Qvantel focuses on providing customized, cloud based, critical business support solutions (BSS) to ICT service providers, e-Invoicing consolidators, utilities and media companies. The company is located in Finland, Sweden, Estonia and India with most of the over 250 people located in Jyväskylä, Finland. (Qvantel, 2015)

Qvantel Corporation constructs from several companies. Qvantel Finland Oy, Qvantel India, Qvantel AB, Qvantel Estonia OÜ and Onesto Services Oy. Qvantel India is located in Hyderabad and focuses on development projects that cover the Internet of things for Qvantel customers. Qvantel acquired Onesto Services in the early 2014 to enable Qvantel to provide a wider solution portfolio to telecom operators. Onesto Services focuses on designing service concepts and BSS solutions e.g. customer service, prepaid integration, credit scoring, and sales force management tools. (Qvantel, 2015) Qvantel Sweden Ab is located in Karlskrona and mainly focuses on developing the web application modules of the Qvantel BSS solutions. (Qvantel, 2015)

Qvantel Finland Oy is the largest unit of Qvantel Corporation and has over 150 employees with most of them located in Jyväskylä. The company provides end-to-end support solutions on business process for billing and enterprise resource management combining online store sales and product catalog solutions to telecom operators' back-end systems. (Qvantel, 2015)

1.2 Scope of Thesis

The thesis focuses on the implementation of the Centrify Suite solution to enhance the capabilities of Active Directory (AD) in an environment that contains Windows, Linux and Solaris servers. AD implementation and configuration was not included in the thesis as a whole; however, there are parts that were critical to the configuration of Centrify and therefore they were included. In addition, initial testing and evaluation of Centrify Suite 2012 is no included in this thesis regardless of the importance of proper product evaluation and testing. This thesis does not cover workstation environment deployment although it was a part of the centralized authentication project.

1.3 Background

Due to the growth of the amount of systems and personnel that need to be managed by the administrative team the pressure for resourcing has grown linearly. In addition, end users have faced a situation where they have a huge amount of usernames and passwords to remember. A recent survey by Finn Partners and Centrify (Finn Partners, 2014) shows that an average 100-person company loses over 35 000€ per year in productivity because of password management alone; not to mention the security risks that are faced when users circulate the same password in many systems

because it cannot be denied, managed or monitored by anyone with reasonable effort.

Qvantel has chosen the Centrify Suite to help securely leverage AD infrastructure to manage centrally authentication, access control, privilege management, policy enforcement and compliance across server platforms (virtual and physical). The solution was deployed to Linux and Solaris servers in the Qvantel production platform and to the heterogeneous workstation environment, which included Windows, Macintosh OS X and Linux workstations.

1.4 Research method

Choosing research methods proved to be somewhat challenging in this case since the thesis describes an implementation of a technology to an information system, and the desired end result for the thesis was to cover the actual technical aspect of the change in the environment. A further aspect of research could have been the selection of the technology or study the impact of the implementation on the business processes. Those areas would have presented more options in choosing actual research methods because in that case qualitative and quantitative methods could have been used to study the results in more depth.

Case study was a logical choice as the main research method; however, as Lukka states in his article (Lukka, 2001), case study method can contain several sub methods such as constructive method, which is used to research real-life problems that need to be solved. Currently the constructive research method is commonly used in information systems science (Lukka, 2001). Unlike conventional research methods, constructive research seeks to strongly influence the subject and the ideal result of the research would be to solve a real-life problem with new construction implementation that offers practical and theoretical contribution.

Quantitative methods were used to some extent to interpret the facts and initial theory that the author had regarding the impact of the final implementation. Since quantitative methods involve measuring and statistical analysis (Gillham, 2000, p. 9) they were used to analyze how the implementation affected the Identity and access management (IAM) in teams that had the most workload in processing the access requests. As stated in the previous chapter, the workload was growing rapidly regarding IAM, which had an effect on resolution times used as evidence in the quantitative analysis. The evidence was gathered from the company service request management system where all IAM issues were documented.

1.5 Research objectives

The objectives for the research were to establish a technological capability for scalable IAM in the organization's production platform that would enhance administration work, information security and provide evidence in form of audit logging. In addition, the thesis strives to study what level of impact a managed IAM caused to operating expense (OPEX). How could the initial capital expenditure (CAPEX) be redeemed in operational cost savings? Moreover, what would have been the situation without the initial investment?

2 SECURITY

2.1 Information security

The objective for information security in businesses in general is to protect the organization's information by reducing the risk in losing the integrity, confidentiality and availability of information assets to an acceptable level.

Confidentiality

The term confidentiality means guarding the information from everyone except those with rights to it. The information includes private data of customers, and intellectual property of businesses or customer information. In everyday life, private data is handled and passed on to other parties to store. These transactions are so common that people do not even understand the vast amount of information that is trusted to another party. To establish a trust to a service it is highly important to keep this data confidential. (Solomon & Kim, 2012, p. 12)

Integrity

Information must not change uncontrollably and its validity and accuracy must be maintained. If the integrity of information is compromised, then the information has no use since it cannot be trusted. Integrity deals with validity and accuracy of data (Solomon & Kim, 2012, p. 12).

Availability

In information security, availability is a common term used to express the amount of time users can access a system, application or data. The common measurements used to calculate the availability are : (Solomon & Kim, 2012, p. 11)

- Downtime
- Uptime
- Availability = (Total Uptime) / (Total Uptime – Total Downtime)

2.2 Identity and access management

What is identity? That can be defined in information technology as a set of user attributes or properties that distinguish entities in the environment from one another (Bertino & Takahashi, 2011). Users in this context are not just people but also computing software agent or a hardware device. IAM is a set of technologies and business processes to enable individuals to access the needed resources when they need and have permission to access them for a specified reason. IAM usually enhances the organization's capability to manage changes to access rights and other features in a more secure way.

2.3 Privileged access management

User account compromise is the most probable way for a malicious entity to access organization's infrastructure based on the 2015 Data Breach Investigations Report (DBIR) by Verizon as seen in Figure 1.

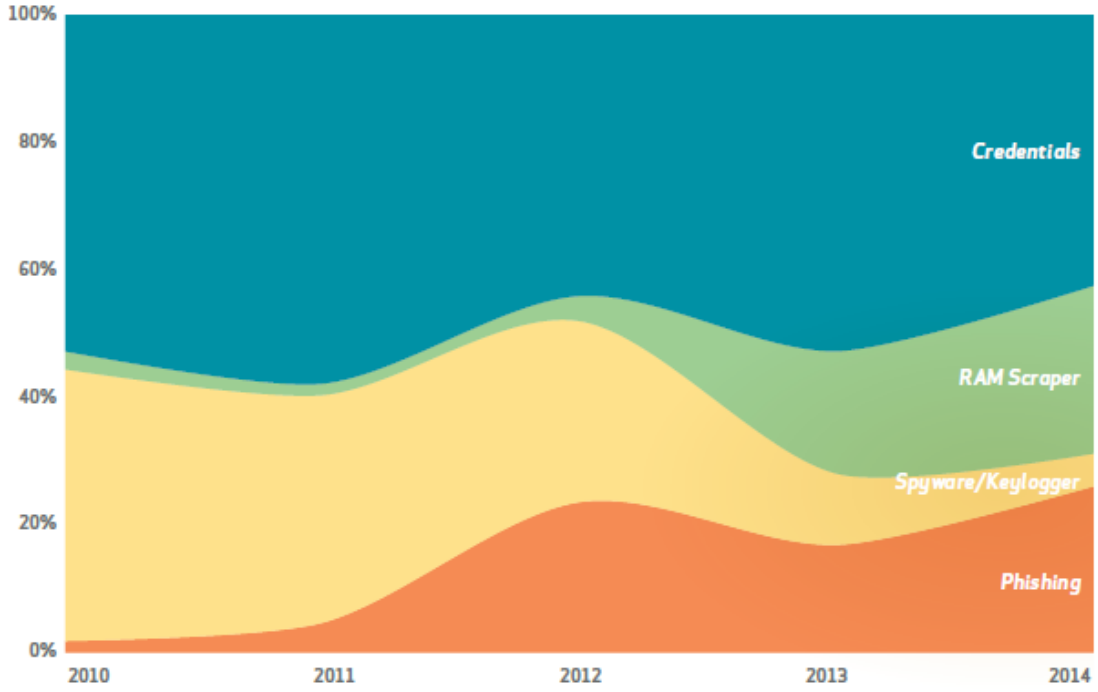

Figure 1. Significant threat actions over time by percent. (Verizon Enterprise Solutions, 2015)

According to Micki & Tipton (2012) “Effective privileged user controls need to combine policies, procedures, and technologies that address the particular environment and needs of organizations.”. The way organizations adapt this varies and one could state that identical solutions are not found between different organizations because they always need to match the specific requirements of the company. Every organization that wants to secure access management, prevent or detect incidents or react quickly when user access is compromised, needs to have appropriate privileged user controls in place. (Micki & Tipton, 2012)

2.4 Authentication

Authentication as a process is about verifying the truth of a claim (National Research Council, 2003, p. 33). It is about making sure that users are who they claim to be. It is

also about ensuring that authorized end users initiate all processes and transactions. The authentication process pairs the login information such as user id or a user account with a password to form an identifier for the user. The identifier is used to assign privileges and to track all activity for auditing purposes. Identification is the means by which a user provides a claimed identity to the system. Authentication is the means of establishing the validity of this claim. (National Institute of Standards and Technology, 1997, p. 181)

2.5 Critical security controls

In 2008, the U.S. National Security Agency (NSA) began an effort to approach the threats of cyber security with a list of controls that would have the greatest impact on improving the security against real-world threats. The venture quickly grew outside of U.S. government domain when international agencies and privately held companies joined the effort. As a result, a list of recommendations as critical security controls was published through SANS Institute. Currently, the list of controls is planned and managed by a global independent entity called the Council on Cybersecurity. Below is the list of Critical Security Controls as follows: (SANS Institute, 2014)

Critical Security Controls version 5

1. Inventory of Authorized and Unauthorized Devices
2. Inventory of Authorized and Unauthorized Software
3. Secure Configurations for Hardware and Software on Mobile Devices, Laptops, Workstations, and Servers
4. Continuous Vulnerability Assessment and Remediation
5. Malware Defenses
6. Application Software Security
7. Wireless Access Control
8. Data Recovery Capability

9. Security Skills Assessment and Appropriate Training to Fill Gaps
10. Secure Configurations for Network Devices such as Firewalls, Routers, and Switches
11. Limitation and Control of Network Ports, Protocols, and Services
12. Controlled Use of Administrative Privileges
13. Boundary Defense
14. Maintenance, Monitoring, and Analysis of Audit Logs
15. Controlled Access Based on the Need to Know
16. Account Monitoring and Control
17. Data Protection
18. Incident Response and Management
19. Secure Network Engineering
20. Penetration Tests and Red Team Exercises

The centralized authentication solution can provide controls against several critical threats in the cybersecurity domain. AD with Centrify can be configured to provide capabilities beyond just authentication. The use of Group Policies has been a way to provide configuration management in Windows environments and with Centrify attached to AD that is possible in UNIX as well.

When implementing the centralized authentication with the right tools, several controls can be implemented completely or partially and thus decrease the risks in information security. The controls achieved with the right implementation are:

- Partial inventory of Authorized Devices
- Secure Configurations for Hardware and Software on Mobile Devices, Laptops, Workstations, and Servers
- Partially controlling Vulnerability Assessment and Remediation with patch management
- Controlled Use of Administrative Privileges
- Controlled Access Based on the Need to Know

- Account Monitoring and Control

2.6 Segregation of Duties

The principle of segregation of duties separates an operation into function so that no single person can control the whole process from initial actions to finish. Actions would require multiple persons on multiple roles to complete a specific operation. One person creates code, the second person reviews it and another person deploys it to the system. Segregation of duties is especially critical, for example when handling financial transactions.

3 OVERVIEW OF LINUX AUTHENTICATION

3.1 Conventional authentication

Linux was not originally built with a single authentication solution in mind (Kirkpatrick, 2008) and as a result, Linux application developers generally implemented their own authentication solutions to their applications. The authentication solutions could use the operating system credentials since in almost all Linux distributions user information is stored in */etc/passwd* and */etc/shadow* which are text files that contain account information like username, User ID (UID), Group ID (GID), full name, home directory and shell type (Schneider, 2003). The */etc/shadow* contains the actual passwords in encrypted format and additional properties related to the password. (Schneider, 2003)

A sample /etc/passwd entry

```
roger:x:503:50:Roger Foster:/home/roger:/bin/bash
```

An example of the users /etc/shadow entry

```
roger:$1$wKAP1RyH$JeCAcEGhSGVlD0J7.AMg.0:14396:2:5:7:30::
```

In 1995 Sun Microsystems proposed a mechanism called Pluggable Authentication Modules (PAM) which provided a common set of APIs for authentication that application developers could use. The document defining PAM is a request-for-comments paper (RFC) that was written by Vipin Samar and Roland J. Schemers III of SunSoft, Inc. Specifically, it is OSF-RFC 86.0, October 1995, “Unified Login with Pluggable Authentication Modules (PAM)” (Morgan, 1997). Originally, PAM was developed for Solaris but later on 1996 it was ported to Linux systems as Linux-PAM. (Geisshirt, 2007)

PAM modules are classified into module types that should implement at least one of the four module type functions (Geisshirt, 2007):

1. The authentication module is used to authenticate users or set/destroy credentials.
2. The account management modules perform actions related to access, account and credential expiration, password restrictions/rules, etc.
3. The session management module is used for initializing and terminating sessions.
4. The password management module performs actions related to password change/updates. (Srivistava, 2009)

The illustration of PAM architecture is shown in Figure 2.

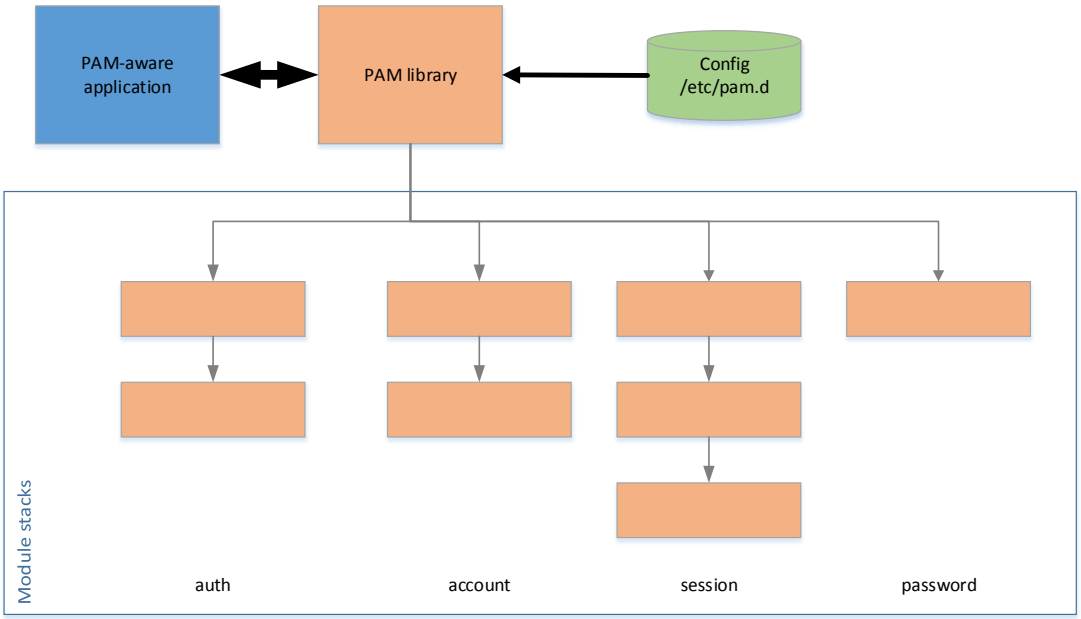

Figure 2. PAM architecture and relation of parts. (Geisshirt, 2007)

PAM provides functional capabilities to implement authentication, such as access control, session management, and more. (Geisshirt, 2007):

Nowadays most Linux distributions have multiple PAM authentication modules that provide support for a Lightweight Directory Access Protocol (LDAP) directory and ability to use Kerberos for authentication (Kirkpatrick, 2008).

3.2 Authenticating with SSH

SSH is used to secure connections and encrypt communication between user and server. The authentication process is handled with public-key encryption and transmissions are encrypted by a cipher agreed by the client and the server for a particular session. Upon connection SSH authenticates a host by verifying that the host is valid and can be communicated with. After establishing that the user is authenticated to verify the user is who he claims to be. (Petersen, 2008)

3.3 Kerberos authentication

Kerberos is a network authentication protocol that provides encrypted client and server communications. The service involves two modules, an authentication server (AS) and a ticket-granting server (TGS). When combined they form a Key Distribution Center (KDC). User is validated based on information stored in the user database (Petersen, 2008). Kerberos is used every time a user logs to an AD-joined computer and when a user accesses a network resource like a file share. AD DC functions as the KDC server handling the tickets between clients and users. (Walla, 2000) In Centrify enabled network Kerberos authentication is a vital part of user authentication to servers. Figure 3 illustrates the Kerberos authentication process where client is accessing a server using Kerberos authentication.

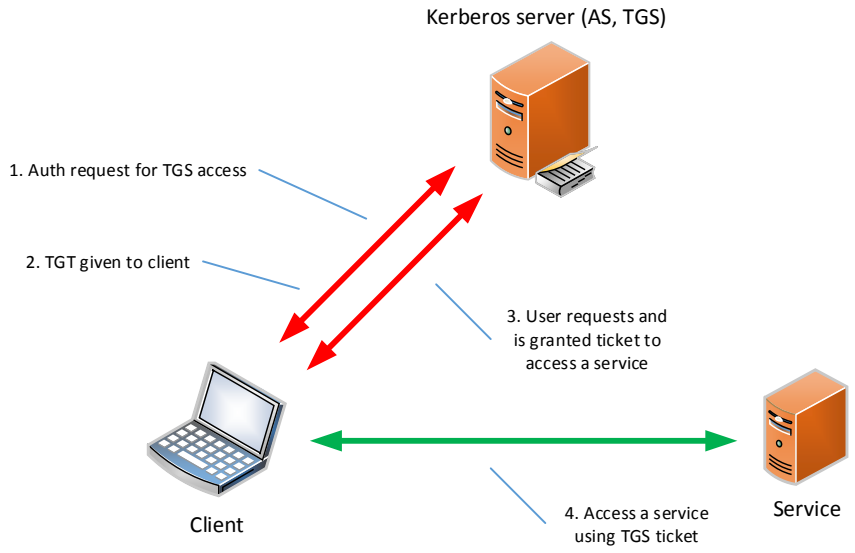

Figure 3. Kerberos authentication.

Kerberos authentication process when authenticating to a server: (Walla, 2000)

1. Authentication request for TGS access
2. TGT given to client
3. User requests and is granted ticket to access a service

4. Access a service using TGS ticket

3.4 Traditional integration between Linux and Windows

Several Linux distributions contain some level of readiness for AD integration, which can be used for file sharing and login access. There are few possible paths to integrate a Linux system to AD and usually they involve solutions like Winbind, LDAP/Kerberos or System Security Services Daemon (SSSD) to establish the connectivity (Cowley, 2013). Not all of the solutions listed provide services like file sharing; however, file sharing can be enabled using Samba to provide Windows networking capability.

As seen in Figure 4, PAM communicates with SSSD or Winbind, which uses LDAP and Kerberos to talk to AD. LDAP is used to perform identification and Kerberos handles authentication. Regardless of the solutions used to integrate the platforms, some prerequisites need to be fulfilled (Cowley, 2013).

- Fully working Domain Name Service (DNS) with forward and reverse lookups. Kerberos authentication will fail if DNS is not operating correctly.
- Time must be in sync on all nodes included in the integration. Kerberos can withstand a small (approximately five minutes) difference in system times otherwise, it will fail.
- *Identity Management for UNIX* feature must be installed in the AD so that *NIX system data like GID, UID and home directory information can be submitted.
- LDAP searches must be enabled from the Linux clients either using anonymous bind or authenticated. Anonymous bind is not recommended in any situation.

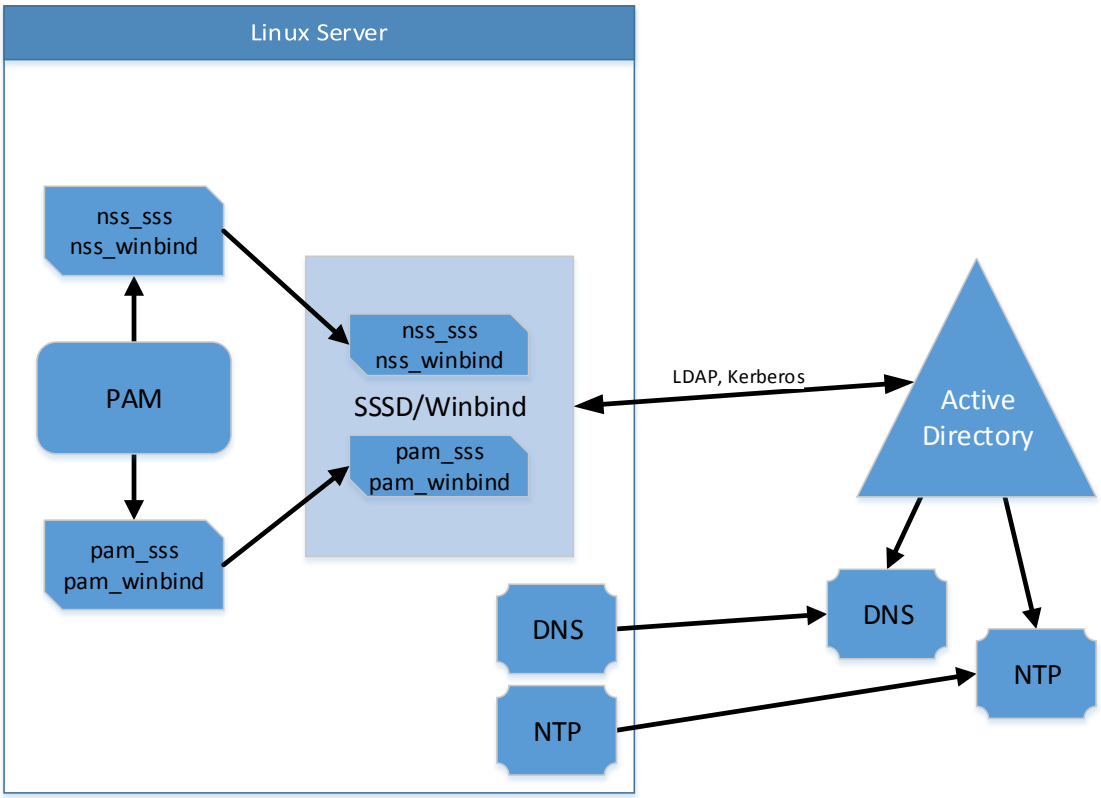

Figure 4. Authenticating to Active Directory on Linux. (Cowley, 2013).

While the native integration of a Linux system to AD seems straight forward and well documented procedure it still requires large amounts of manual configuration work or great effort in producing a deployment system that automates the steps of the integration and that would not be cost efficient when scaling the amount of systems and complexity.

3.5 Privilege Escalation in Linux

In Linux, there are two ways to execute applications as administrative (root) user. User can switch to root user using the *su* command, or by using the *sudo*. *Sudo* command comes from words “substitute use do” or “super user do”. What *sudo* does is

that it enables user to switch to a different user to execute a command, whether it is root or some other user. Configuring *sudo* command is done via command *visudo*, which opens the */etc/sudoers* file for editing. Disadvantages for using *sudo* is that the configuration file is a local file and for effective rights management the configuration needs to be distributed to servers in some way.

4 MICROSOFT ACTIVE DIRECTORY SERVICES

4.1 Benefits of Active Directory

Active Directory Domain Services (AD DS) is a multi-master LDAP compliant database that is used to create a scalable, secure, and manageable infrastructure for user and resource management, and to provide support for directory-enabled applications. (Microsoft, 2013)

In business environments AD services have become the most used solutions in providing services needed to manage information resources such as users, groups and computers. AD DS provides excellent tools for businesses to manage their assets by organizing them into a hierarchical structure and proving an integrated security features for authentication and access control of the resources in the environment. By adding other AD roles such as Federation and Certificate services a company can add the value gained from the Domain Services by using it as a central authentication directory for external services and a distribution system for PKI solution.

AD DS is designed for scalability in mind where the AD forest acts as a security boundary for an organization and scope of administration privileges are defined within the AD forest. In a single domain environment, a forest contains only one do-

main, however an additional domain can be created to the forest to provide partitioning of the information stored in the AD. Separation of domains in the forest could extend the administrative control over different stakeholders for security reasons. For example, in an environment where an office's information assets and administration need to be completely separated from the production server administration.

One of the powerful tools in AD DS are Group Policies that ease the management of complex environments. Policies define setting for the various components of the systems administrators need to manage like:

- Security settings
- Fine grained password setting
- Assign scripts
- Manage settings based on templates

Policies can be targeted based on many different methods or a combination of them, which gives administrator flexibility in defining the practices how to implement the designed policies to environments.

4.2 How information is stored in Active Directory

Network objects are stored into a secure hierarchical containment structure that is known as the logical structure. (Microsoft, 2014) Forests, domains and organizational units (OUs) are the core elements of the AD logical structure. The elements are described in Table 1.

Table 1. Domain and Forest Components. (Microsoft, 2014)

Component	Description
Forest	A forest is the highest level of the logical structure hierarchy. An AD forest represents a single self-contained directory. A forest is a security boundary, which means that administrators in a forest have complete control over all access to information that is stored inside the forest and to the DCs that are used to implement the forest.
Domain	Domains partition the information that is stored inside the directory into smaller portions so that the information can be more easily stored on various DCs and so that administrators have a greater degree of control over replication. Data that is stored in the directory is replicated throughout the forest from one DC to another. Some data that is relevant to the entire forest is replicated to all DCs. Other data that is relevant only to a specific domain is replicated only to DCs in that particular domain. A good domain design makes it possible to implement an efficient replication topology. This is important because it enables administrators to manage the flow of data across the network, that is, to control how much data is replicated and where that replication traffic takes place.
OU	OUs provide a means for administrators to group resources, such as user accounts or computer accounts, so that the resources can be managed as one unit. This makes it much easier to apply Group Policy to multiple computers or to control the access of many users to a single resource. OUs also make it easier to delegate control over resources to various administrators.

In addition to forests, domains and OUs AD logical structure contains containers that resemble OUs in many ways. The domain has seven built-in objects with object type as container. They come pre-installed when the first DC is promoted to network.

Figure 5 shows the structure of the AD and the relations between forest, domain, OU and DC.

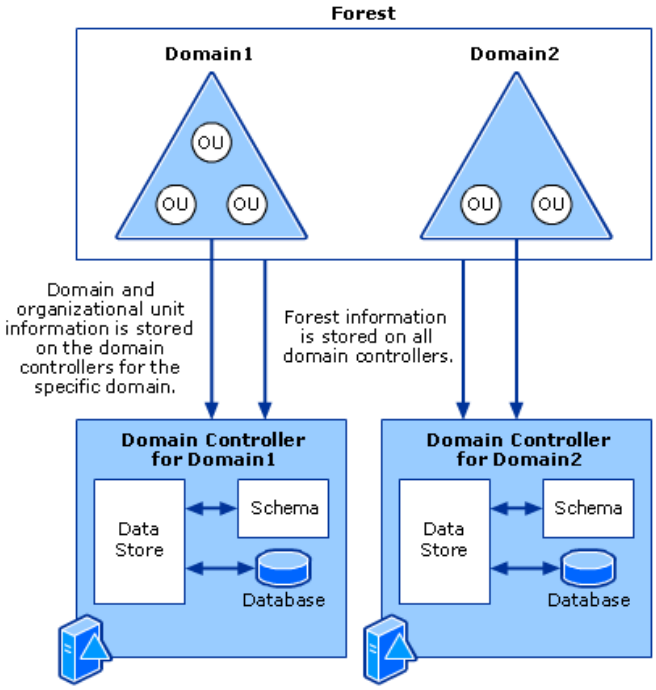

Figure 5. Active Directory Data Structure. (Microsoft, 2014)

4.3 DNS and Active Directory Sites

Centrify agent determines its connection status based on the availability of DNS and AD. If the initial DNS request for a host name is successful, the Centrify agent attempts to connect to the appropriate DC and Global Catalog for its joined domain using the site information found in DNS. Site information is configured using AD native tool *Active Directory Sites and Services.* All subnets that contain servers joined to Centrify managed hierarchy need to be defined in an AD Site. Using the site information, the agent queries DNS for a list of the DCs in its site and attempts to connect to the nearest DC.

4.4 Read-only domain controller

Microsoft introduced Read-only domain controller (RODC) with the Windows Server 2008 release. The new server role is intended to be used in situations where physical security could not be guaranteed. RODC can host read-only copies of the partitions of the AD database and read-only copy of the *SYSVOL* folder contents. With RODC, administrators can selectively cache credentials to provide fast authentication and domain services in locations that lack the security of a datacenter such as branch offices and perimeter networks. (Microsoft, 2012)

In addition to enhancements to security, RODC can simplify the replication topology of and environment. Replacing writable DCs in remote locations reduces the load on actual DCs in datacenters that are replication partners to a large number of other DCs in remote locations. In addition, the use of RODC reduces the number of replication connections in the topology. (Microsoft, 2012)

4.5 Perimeter network

Perimeter network, which is referred commonly as demilitarized zone (DMZ) in the firewall business. DMZ is a closely monitored network from where services can be exposed to public without creating unnecessary risk to organizations internal network. Figure 6 illustrates the traffic flow between DMZ and internal network. Fundamental philosophy of DMZ is to provide multiple layers of security to protect critical assets better. If a hacker breaches the firewall and access to DMZ is gained, lateral movement would not provide significant results. Without the DMZ, the first penetration would expose the internal network and result in serious compromise. Services placed in the DMZ should never be trusted fully because the risk of compromise is higher than in the internal network. (Cheswick, Bellovin, & Rubin, 2003)

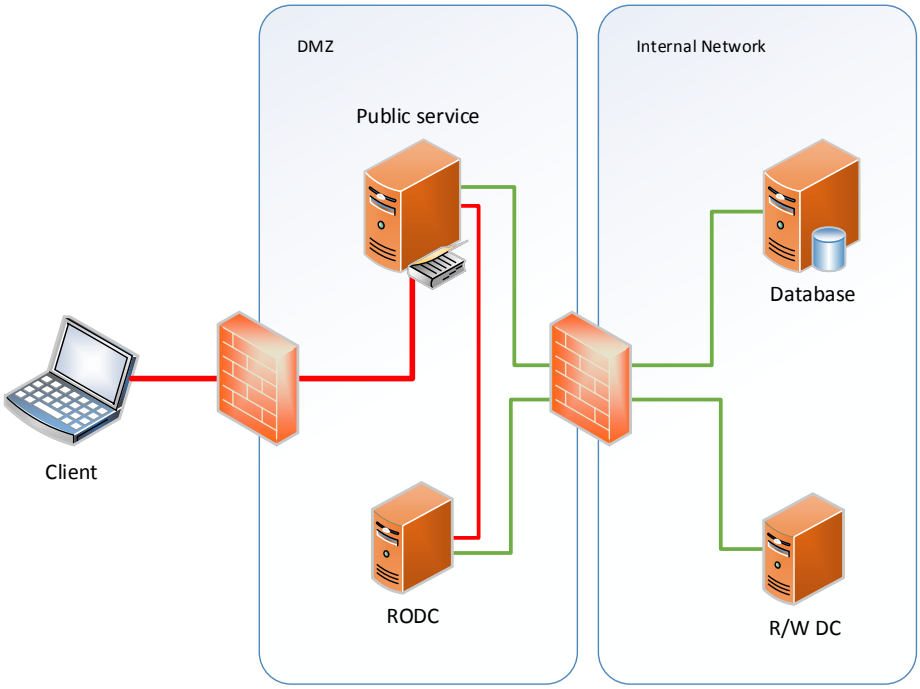

Figure 6. Traffic flow from Internet to perimeter network and from perimeter to internal network.

5 CENTRIFY

5.1 Company

Centrify was founded in 2004 and is based in Sunnyvale, California. The company focuses on IAM market. The company is competing with Server Suite against CA Technologies Access Control, Quest Software, and Beyond Trust on their main business area; however, they are also a major player in the submarket privileged identity management. (Hudson, 2012)

Centrify has acquired major players from IT business as partners e.g. Apple, Microsoft, SAP and HP and from over 5,000 customers globally Centrify lists organizations from several agencies of the U.S. government to global players like Sony, Verizon, Boeing and Toyota. According to them, they are supplying 24 of the Fortune 50 companies. (Centify Corporation, 2015)

5.2 Centrify software suite

Centrify Suite provides an integration layer between Windows and other operating environments. Centrify Suite 2012 enabled secure authentication, authorization, directory service, and configuration management through Microsoft AD (Centrify Corporation, 2012, p. 8). The software establishes the use of features of AD such as group policies, security groups and user management in various environments like:

- UNIX, Linux, and Mac OS X operating environments
- Web and J2EE application platforms, such as Apache, Tomcat, JBoss, WebLogic, and WebSphere
- Database platforms such as DB2, Oracle, and SAP

When a managed system joins an AD domain, it essentially becomes an AD client and relies on AD to provide authentication, authorization, policy management, and directory services. The interaction between the client agent (*adclient*) on the local computer and AD is similar to the interaction between a Windows client and its AD DC.

In the implementation, Centrify is used to manage Linux and Solaris servers and Mac OS X workstations that are joined to AD domain. Centrify provides tools to enable centralized authentication, password management and rights delegation to operational users (external and internal).

5.3 Centrify Components

The Centrify software suite consists of several components that together provide an integration layer between Microsoft Windows-based AD environment and information systems running on variety of other operating systems or application environments.

Components are split to two main categories that include applications running on Windows machines that provide the functionality to manage AD-based objects and agents that run on systems that are integrated to AD.

5.3.1 DirectManage Deployment Manager

With Deployment Manager, administrators can manage the Centrify software inventory, remotely access information about remote systems and use the tool to deploy client software. In addition, the tool can be used to perform basic evaluation of security risks and vulnerabilities. Deployment Manager is especially valuable when deployments are done to environments that have vast amount of unmanaged servers and updating the clients on the managed systems. Figure 7 illustrates the user interface (UI) of the Deployment Manager.

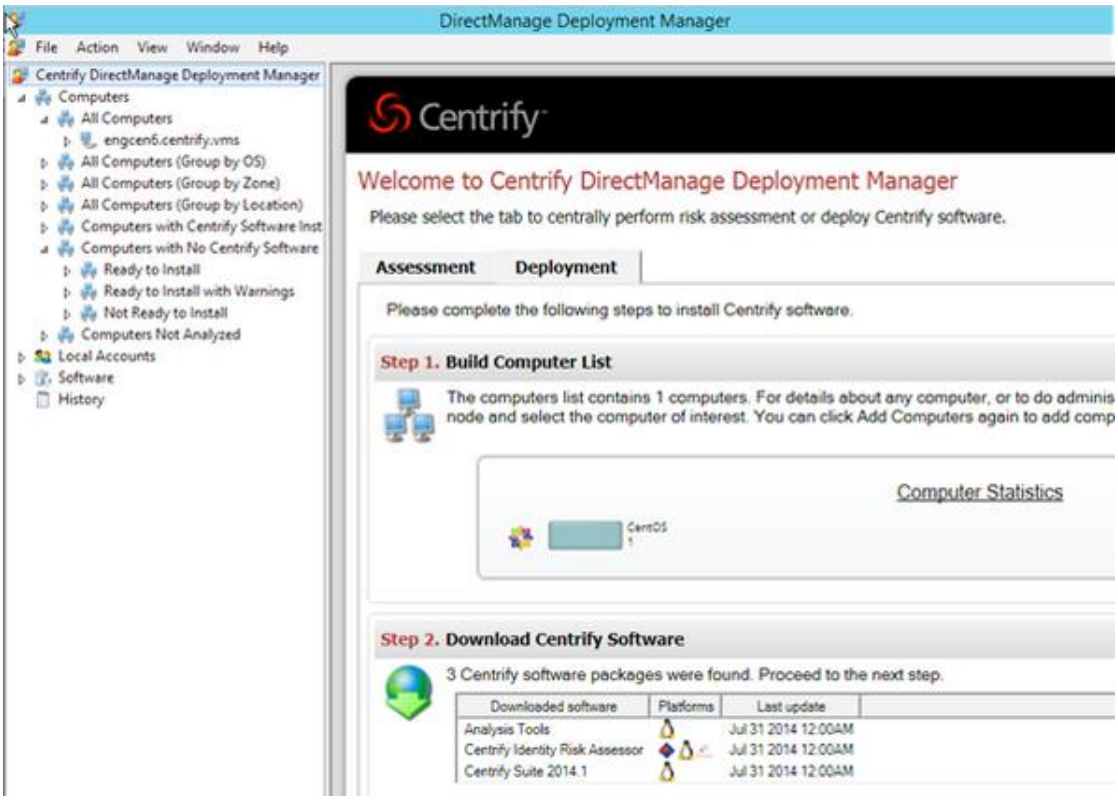

Figure 7. Deployment manager.

5.3.2 Zone Provisioning Agent

The Zone Provisioning Agent (ZPA) enables automated provisioning of user and group accounts to Centrify zones. ZPA monitors zone specific AD groups, which contain user accounts, or groups needed in the specific zone. ZPA adds or removes members from designated zones based on changes made to the defined AD groups. The business rules that control provisioning are stored in AD.

The application runs as a separate service on a Windows server on an ongoing basis and should be always available. ZPA can be installed on two computers but only one instance should be running at a time. The second instance can be used for standby operations like maintenance or the primary node. At least one ZPA should be installed to each forest if multiple forests exist in the environment.

Following components are included in the installation:

- **Zone Property Page Extension** is installed on the same computer as Access Manager, which enables a tab in zone properties for making provisioning configurations.
- **Provisioning Agent Windows service** is the main application handling the provisioning of assets and should be always available.
- **Command Line Interface (CLI)** allows administrators to write scripts for provisioning tasks on demand.

5.3.3 DirectControl Administration Console

The DirectControl Administrator Console is a Microsoft Management Console (MMC) snap-in. It is the primary console for managing Centrify DirectControl properties because it provides access to a full spectrum of management activities specific to DirectControl. DirectControl Administration Console must be installed at least on one computer that can access the domains in AD.

The use of administrative console is illustrated in Figure 8:

1. Administrator manages the system using console access
2. Administrative Console is installed on one server and integrated to AD domain. Administrators can manage the operation and deployment of Centrify from the server.
3. AD domain, which contains all servers, users, groups and computers.
4. Member systems running variety of operating systems such as Solaris, Linux and Mac OS X.
5. DirectControl Agent (*adclient*) used to integrate and manage the client systems.

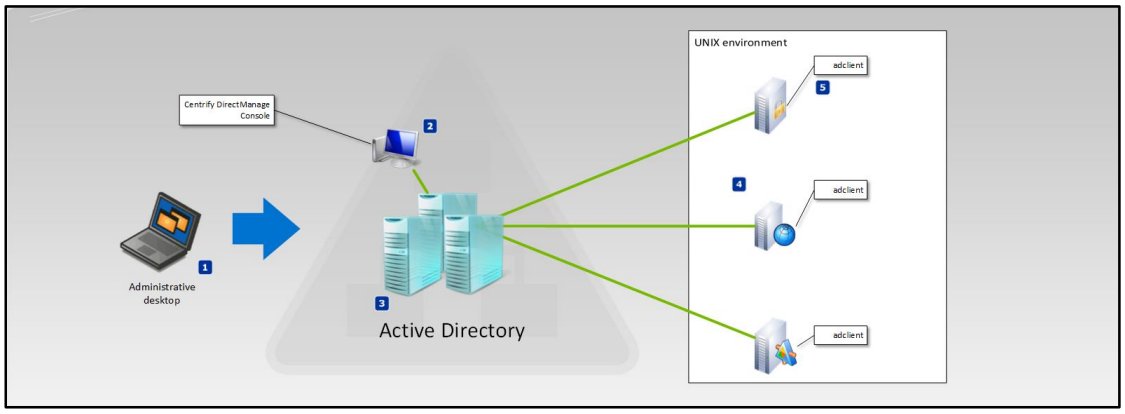

Figure 8. Simplified view of AD integration.

5.3.4 DirectControl Agent

The Agent enables UNIX, Linux or Mac OS X systems to communicate with AD like a Windows system. The client agent performs several tasks on the managed system related to communication and authentication and enables the implementation of Group Policies on the non-Windows client systems.

Agent installation consists of several different modules that manage different aspects of the integration (Figure 9). During the agent installation, several commands like programs are also installed that can be used to manage the systems.

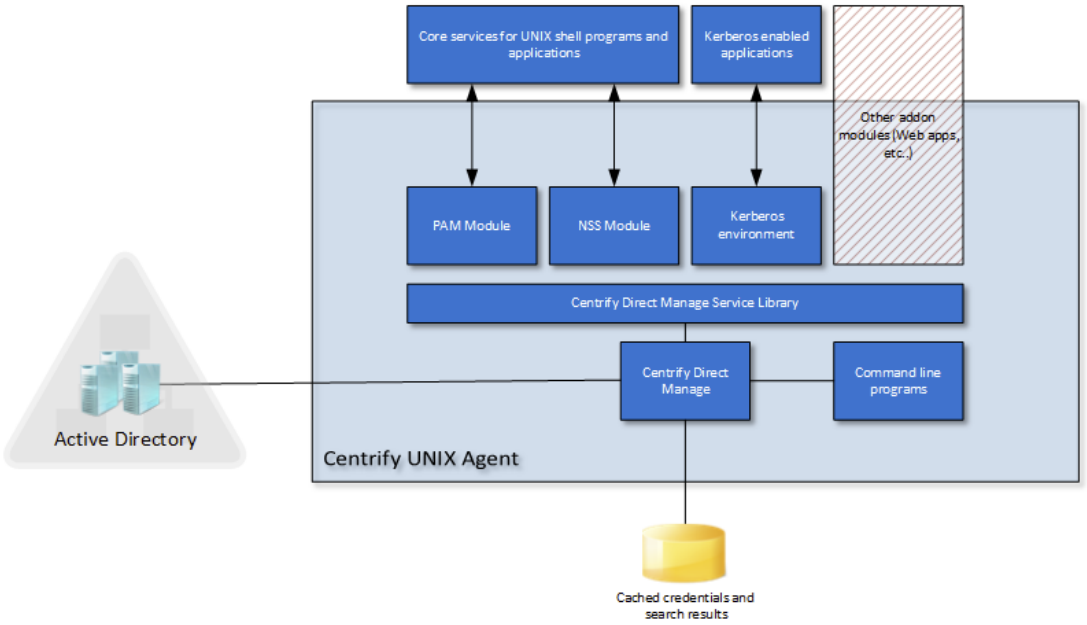

Figure 9. Centrify UNIX agent overview.

Adclient process is the core that handles all direct communication with Active Director. The agent contacts AD when there are requests for authentication, authorization, directory assistance or policy updates. Then the credentials or other information are passed to the program that need this information. (Centrify Corporation, 2012, p. 26)

6 MANAGING SYSTEMS WITH CENTRIFY

6.1 Zones

Centrify allows two fundamentally different solutions in creating Zones. Classic Zones were introduced before Hierarchical Zones and they represent a common bundle of systems, users and groups that were identified from company environment. In order to account for multiple bundles, multiple zones needed to be created as illustrated in Figure 10.

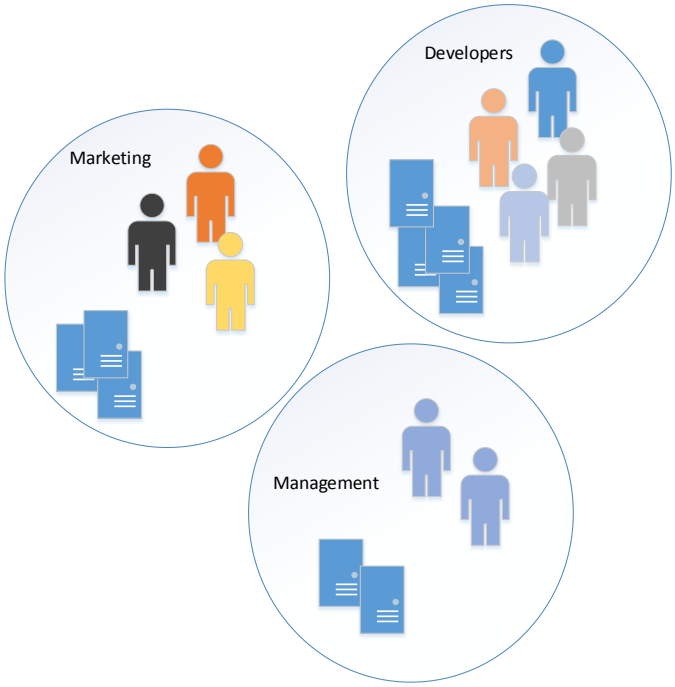

Figure 10. Classic Zones from bundles.

With creative use of PAM overrides and Group Policies, some of the Zones could be combined; however, this can be rather complex and difficult to audit. The fundamental reason why classic Zones are not used is that inheritance of roles and rights is prevented. It is not possible to define a group of users who could perform privileged commands or log in to multiple Zones. Instead, using classic Zones, each UNIX user needs to be added to each Zone, and then roles and rights in each Zone are configured, and after that, the roles to users are assigned in all Zones.

Zone is an administrative point and with hierarchical zones, inheritance allows administrators to define UNIX attributes for users and groups in a parent zone and then have that attribute remain identical in all subsequent child zones. However, use of hierarchical zones allows override of inheritance at a granular level. In Figure 11 users *"john"* and *"roger"* locate in the parent zone and are usable in the child zone due to inheritance; however, in this example user *"john"* has UNIX attributes that are overridden in the child zone.

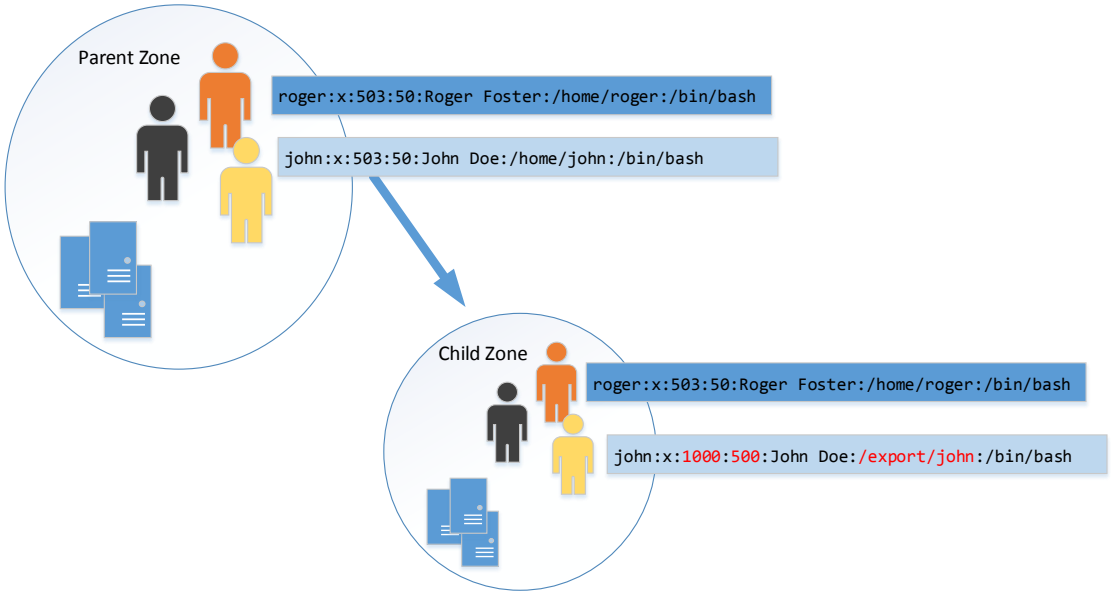

Figure 11. Hierarchical Zones allow inheritance.

There is plenty of functionality in using the overrides since they can be used in multiple levels through the hierarchy. For example, user could have a specific UID in every server in every zone except one where it is overridden at server level.

6.2 Access management using hierarchical zones

The built-in authorization facility (DirectAuthorize) centrally manages and enforces role-based entitlements for fine-grained control of user access and privileges on UNIX and Linux systems. By controlling how users access systems and what they can do on those computers, DirectAuthorize enables administrators to lock down sensitive systems and eliminate uncontrolled use of root accounts and passwords.

Centrify has four different levels of access when using the hierarchical zone structure as seen in Figure 12. As a best practice, the first zone that is created should be a Global zone and all other zones will be child zones for that. Defining access rights to Global zone is considered as universal access and that will grant access to all UNIX

hosts regardless of which zone they are joined to. The next level of access is zone access that will grant user access to all UNIX hosts in the current zone. The third access level is computer override access where an individual or group is granted specific access to a single host. The final level of access is computer role access that gives granular levels of access to a computer or a group of computers.

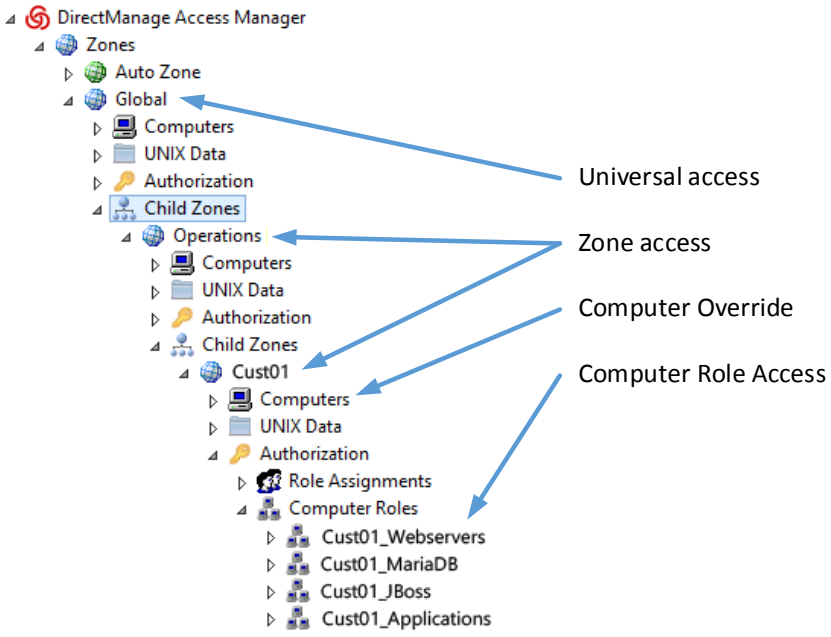

Figure 12. Types of access.

6.3 Computer authorization roles

One of the great benefits of computer roles is that the access can be controlled simply by grouping systems together. Computer groups are normal security groups in AD that preferably reside in a specific OU(s) and follow a defined naming scheme so they are easy to recognize as computer groups used for Centrify computer roles. The groups can be pre-created to AD or they can be created during the creation of computer role. It is recommended that if there are several groups that need to be created they are pre-populated using PowerShell or other scripting tool.

Creating computer authorization roles always when new system groups are added to the environment is recommended since the creation is a one-time operation and managing access to the group of systems can be done from AD after that by adding users to access roles or by expanding the system group with new servers. Figure 13 illustrates the creation of a computer role to Centrify DirectManage Access Manager.

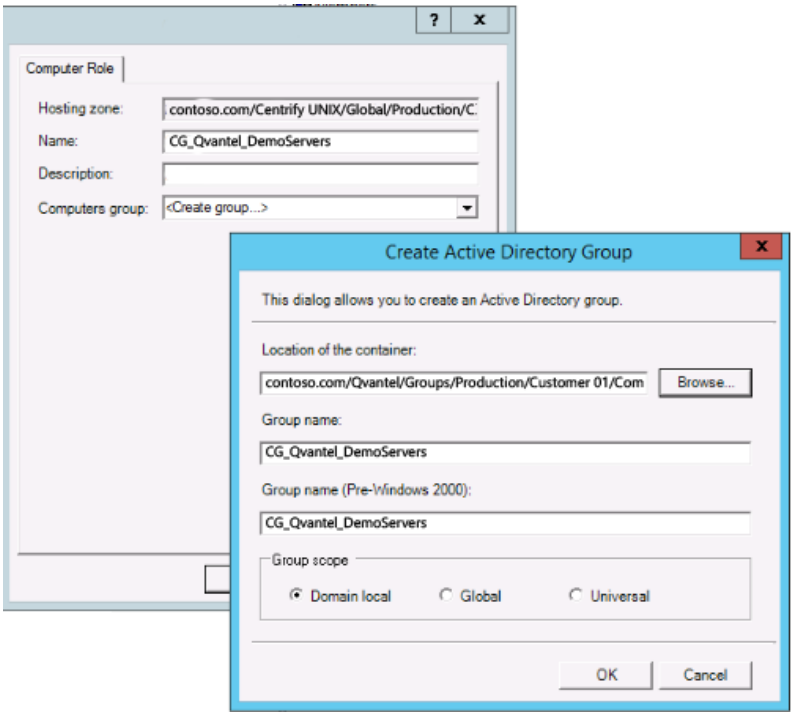

Figure 13. Creating computer group.

6.4 Direct Authorize Do (dzdo)

UNIX commands that require elevated permissions can be defined in the */etc/sudoers* configuration file run by using the *visudo* command. DirectControl provides similar functionality, however, the commands that are configured through command rights, are executed with the *dzdo* command and use the DirectControl authorization store rather than a */etc/sudoers* configuration file.

If a user is assigned a role that includes privileged command rights, the user can run those privileged commands by invoking the *dzdo* command, any command line options, and the privileged command name. The *dzdo* command provides functionality similar to the UNIX *sudo* command to enable a user to execute a command using another user account. A user assigned to a role that includes this right can then execute the command as root by typing a command similar to the following:

```
$ dzdo service crond start
```

Another example for common *dzdo* use would be the change to another user within the UNIX system.

```
$ dzdo –s
```

The dzdo command with the option **-s** commands to open an elevated shell as is stated in the man page of the *dzdo* command.

```
-s        Runs the shell specified by the SHELL
          environment variable if it is set or
          the shell as specified in the user's
          UNIX profile.
```

To change to another user with an interactive shell following command is used:

```
$ dzdo –iu roger
```

6.5 Authorization role definitions

The golden rule with authorization roles is – No role means no access. Authorization roles could be considered as toolboxes that contain all the tools user needs to get

the job done. This means that the authorization role contains access rights and command execution rights for that role that is given to a group of users.

Role definitions are located in every zone within the logical structure. They are inherited to child zones and can be used in any zone they have been inherited.

Figure 14 shows an example of an authorization role, the privileged commands and PAM accesses granted.

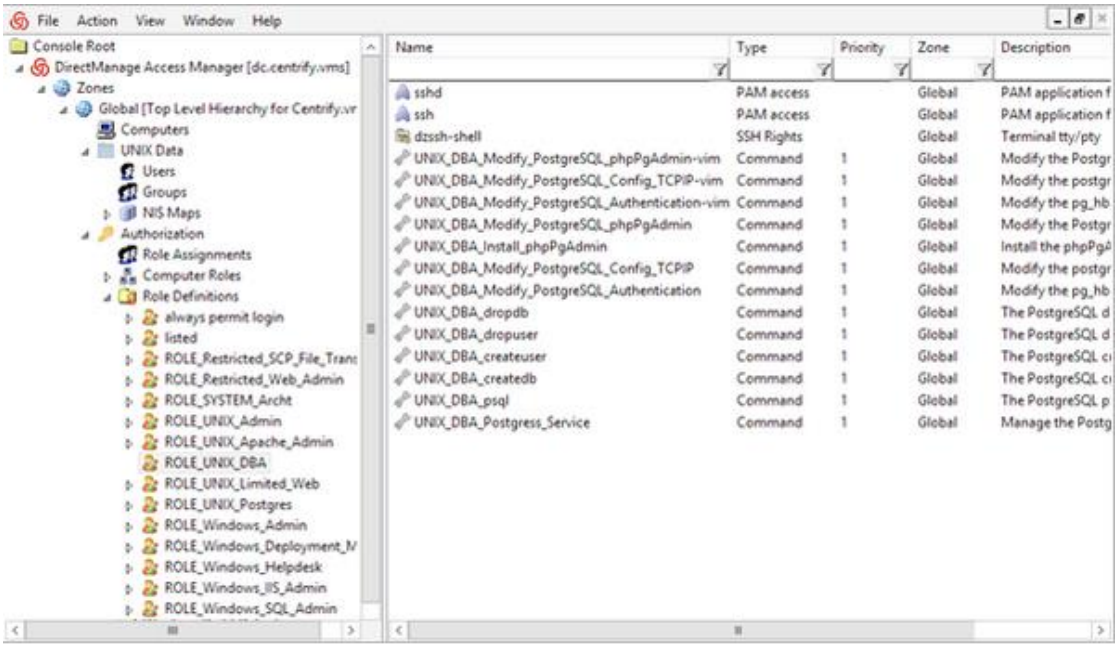

Figure 14. Authorization role definitions.

The authorization roles are assigned in an object from DirectControl and they are connected with a group from AD that contains the desired set of users. The connection can be done in all four levels of access mentioned in chapter 6.2. Basic rights delegation is emphasized in Figure 15 where a group of users located in AD is granted an authorization role and authorization targeted to a groups of computers (computer authorization role). As a result, the users can access all of the servers in the computer groups and have the set of privileges in use that were granted via the authorization role.

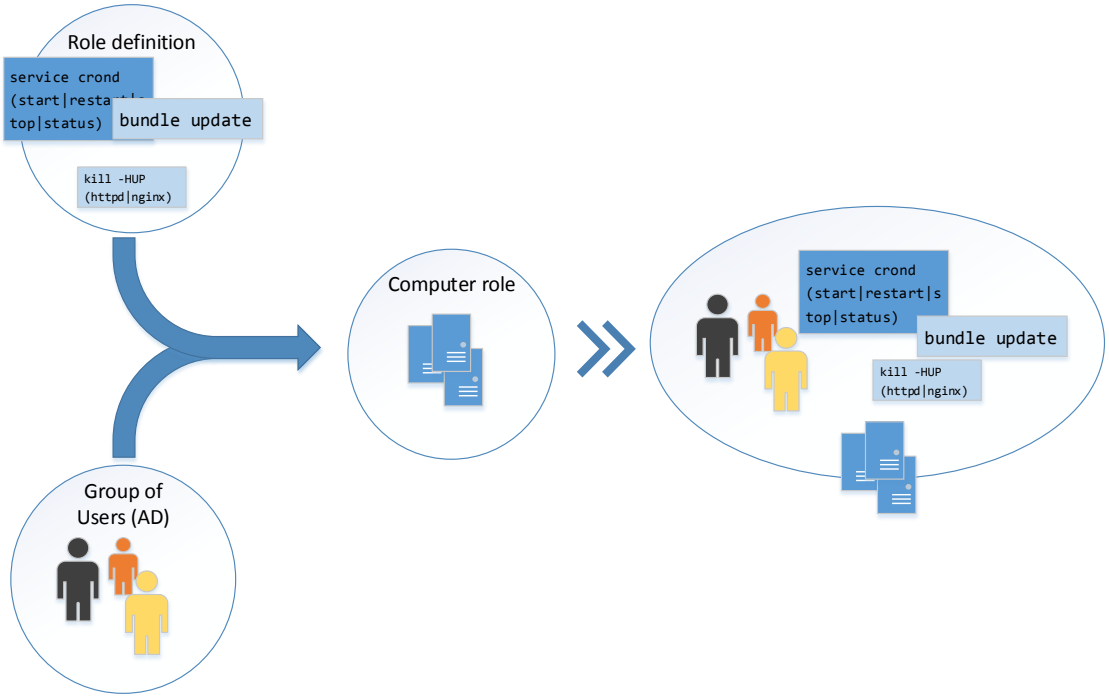

Figure 15. Authorization of access and commands.

6.5.1 DirectControl for PAM services

In Linux and UNIX environments, many services are programmed to use PAM to control access. Centrify supplies default PAM rights for Secure Shell (SSH); however, administrators can add additional rights to other programs depending on the environment and need. In addition to default services, common programs that can be managed through DirectControl are Telnet, FTP, and graphical desktop. Use of wildcards is possible to perform pattern matching for the application name, for example, to match all PAM-enabled applications containing the string ftp string **ftp**. Table 2 describes the predefined PAM access rights that are populated to every zone created to the hierarchical zone structure.

Table 2. Predefined PAM access rights.

PAM right	Access granted
login-all	All PAM applications on a computer joined to the domain. This PAM access right allows users to log on and use any PAM-enabled application. This right uses the wild card (*) character to match all PAM application names and is included by default in the predefined UNIX Login role. You can add this right to any role for which you want global PAM access.
ssh	SSH sessions on Debian and Ubuntu 6 and 7. This PAM access right allows users to log on remotely using SSH connections on Debian and Ubuntu computers joined to the domain.
sshd	SSH sessions on all Linux and UNIX computers except Debian and Ubuntu 6 and 7. This PAM access right allows users to log on remotely using SSH connections on all other distributions of Linux and UNIX computers joined to the domain.

6.5.2 User attributes in Active Directory

Centrify stores user attributes in AD structure at a location specified in the initial installation. All users that are provisioned to Centrify have a set of attributes that form a part their identity in the network including:

- Login name
- UID
- Primary Group (GID)
- GECOS which is defined as user full name
- Default home directory
- Default shell

If the information is changed in AD, it will be visible on all member servers. However, if any of the attributes is changed on a server level using the overrides, the changes are not reflected to AD attributes visible in Figure 16.

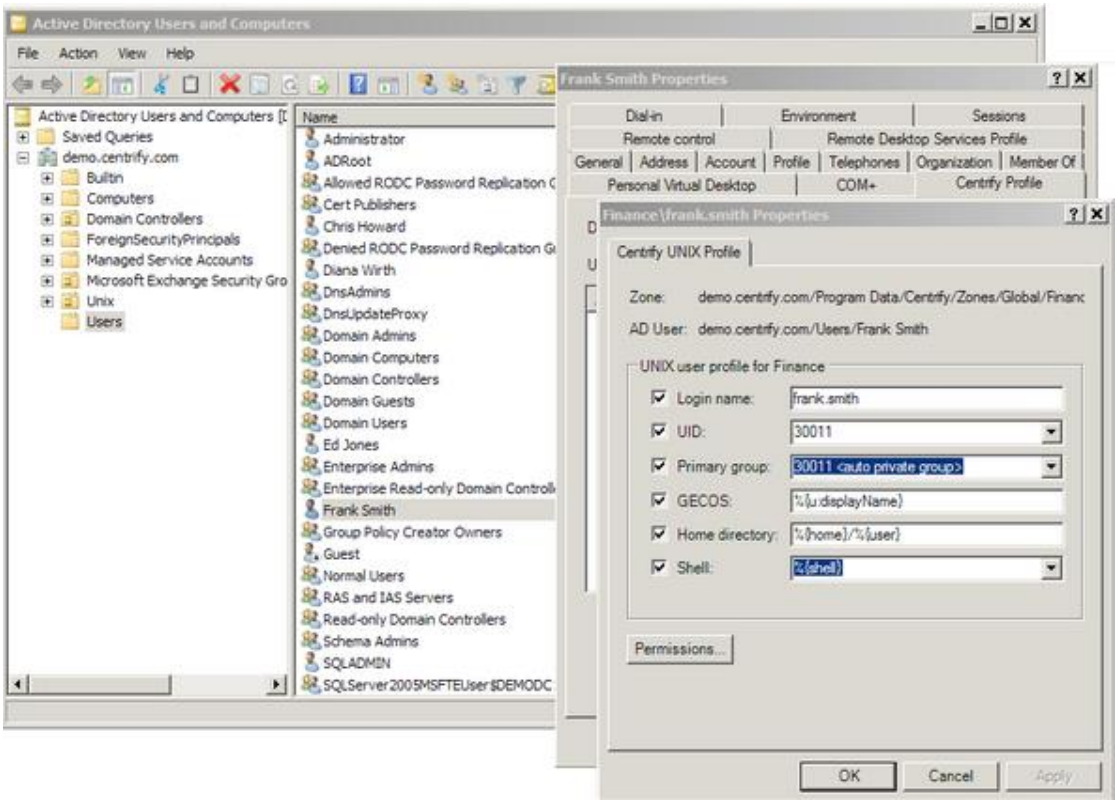

Figure 16. User attributes in Active Directory.

7 OVERVIEW OF THE ORIGINAL ENVIRONMENT

7.1 Environment description

The company provides BSS from a private cloud platform infrastructure that constructs mainly of Linux servers and few instances of Solaris. Total number of servers exceeding over 300 virtual and physical included. In addition to production servers, there are many servers providing support functions like monitoring, backup, networking and file services just to mention few.

The user base can be divided to three main groups. Developers, administrators and support staff. Most of the user accounts reside in the first two groups. Account management practices varied between different sub environments, which made the administrative teams work quite challenging, and time consuming.

7.2 User and access management

With Linux and UNIX systems, there are ways to implement authentication solutions that are centrally managed like LDAP or Network Information Service (NIS). If UNIX server is not attached to any external authentication source, it uses local configuration file to handle password authentication. Only a small portion of servers was part of LDAP authentication service and most of them were servers that provided support for company's internal functions.

Privileged access rights for users were managed using *sudo* and the configuration was managed locally on each server. Service requests that involved some form of access or user management were very time consuming.

7.3 Configuration management

All configurations on the servers were managed manually including privilege escalation and user management. A set of pre-written configurations was deployed during installation to maintain a managed server environment. The servers were kept in good condition despite that there were no configuration management applications like *puppet* or *chef* in use, which was partly due to strict automated monitoring configured to alert if any of the predefined conditions was exceeded. In addition, the administrative team was small and were well aware of the status in the environment.

Although the initial installations were quite well managed, the change management to existing installations brought challenges. There were no centralized means to deploy configuration changes, which left room for human errors. The changes in configuration could not be tracked automatically and up-to-date documentation relied on individual effort by the individual making the change.

7.4 Cost of administrative effort for IAM

Managing user accounts and access rights in an environment that did not have a centralized management point was very time consuming if the tasks included several users or servers needing rights delegation. Simple tasks were handled in quite short time that varied from 10 minutes to 1 hour. If the amount of changes grew, the amount of administrative effort grew linearly in relation. This was verified by investigating issues in the company's issue tracking system and the times used to resolve issues relating to access management.

The plan was to study all issues related to access management in the issue tracking system. However, there were difficulties to filter issues that were interesting to this study and the work was more or less manual. A set of issues was filtered manually

that only introduced access management relating tasks before the centralized authentication was deployed and this can be seen in Table 3. The set included simple issues that were resolved with a minimal effort and issues that were time consuming without proper authentication solution. Almost all of the tasks in Table 3 would have been resolved in less than 30 minutes using IAM solution.

Table 3. Access management issues.

Issue	Created	Time Spent
3182	04.01.2013 10:06	1 h, 0 min
3263	30.01.2013 14:27	0 h, 30 min
3306	18.02.2013 10:31	8 h, 15 min
3311	19.02.2013 09:41	1 h, 0 min
3313	19.02.2013 10:47	0 h, 5 min
3318	19.02.2013 15:59	0 h, 20 min
3334	24.02.2013 21:36	1 h, 45 min
3335	26.02.2013 10:24	4 h, 0 min
3344	27.02.2013 16:21	2 h, 5 min
3419	02.04.2013 13:28	1 h, 0 min
3481	19.04.2013 14:37	0 h, 20 min
3536	17.05.2013 14:41	5 h, 30 min
3574	27.05.2013 09:51	5 h, 20 min
3575	27.05.2013 13:10	0 h, 10 min
3591	30.05.2013 09:22	0 h, 15 min
3597	30.05.2013 16:09	6 h, 45 min
3646	12.06.2013 13:14	0 h, 30 min
3647	12.06.2013 17:25	2 h, 0 min
3672	24.06.2013 15:30	0 h, 15 min
3707	08.07.2013 16:23	8 h, 0 min
3733	30.07.2013 10:09	0 h, 30 min
AVERAGE		**1 h, 30 min**

The average time spent to a simple access change in the original environment was approximately 30 minutes, and complicated access rights took an average of 6 hours and 20 minutes. The average time spent on all access management issues before the authentication overhaul was 1.5 hours. The administrative team specialists that normally are the most expensive resources did all work regarding access management, which was due to the composition of the system so rights management could not be separated to cheaper resources like Service Desk.

The goal was to greatly reduce the time spent on access management issues so that access rights already configured to the system should be taken care of with a 5 to 15-minute effort as well as moving the access management work away from the system management team to another group or team. The change was planned to free resources in the infrastructure specialist team and to distribute the IAM workload to respected stakeholders.

8 CREATING THE PLAN

8.1 Planning as a process

The planning work started based on the information gathered while working in the environment and conducting discussions with the administrators. The planning focused on the idea of centralizing all authentications with a scalable solution while maintaining easy management of rights. Centrify was already selected to be used in with AD during the initial AD implementation and that helped somewhat in the planning process. Since Centrify Suite is a critical part of the IT infrastructure, the testing was carried out before starting the planning of the actual production deployment. However, that is not addressed in the thesis due to the scope of the thesis. (Centrify Corporation, 2012)

The requirements of the solution were listed to provide a base for the design:

- Single user account for operative environments
- Enforced password policy
- Automatic lockdown
- Easy management
- Minimizing administrative effort
- Auditing the usage of servers
- Uniforming the environment
- Automating tasks

The project was split to seven phases, which were called as Deployment life cycle by Centrify Deployment & Planning guide (Centrify Corporation, 2012, pp. 15-16). The project phases followed the standard Information Technology Infrastructure Library (ITIL) Plan Do Check Act (PDCA) model seen in Figure 17. Since the evaluation was done before the actual project, it was left out from the planning.

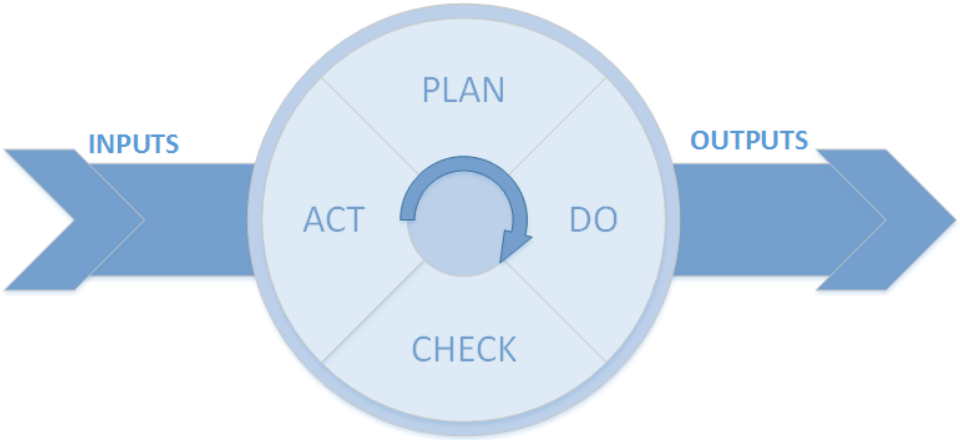

Figure 17. PDCA model also known as Deming circle. (Moen & Clifford)

8.1.1 Design

During the design phase, an additional analysis on the goals and requirements was made. The current state of the environment and organization were also important inputs in order to select the proper deployment model and management options for the to-be state. The primary outcome of the phase was a design document of the environment that defined how the integration would be implemented and how the specific elements would be configured. Licensing requirements were measured based on the amount of servers in active use and what the transition time for servers would be to use the licensed features.

8.1.2 Implementation and Pilot Deployment

In the implementation and pilot deployment phase, the solution was installed to existing AD infrastructure. Applications were installed to management servers and administrative access was configured based on decision made in the projects design phase and in AD design. The main result of the phase was to produce a stable platform for the administrative team to start piloting the solution with a limited group of users and servers.

Users for the pilot deployment were selected based on their duties and access level. For maximizing the yield from centralized authentication, all selected users were from the administrative team and from the Qvantel on-duty team responsible for all the production systems outside normal office hours. These two user groups were selected based on two clear facts:

- Access to all servers
- Motivation to use only one account for all systems

In addition, a third group of users was selected from the development teams to provide insight to what rights and roles needed to be defined and implemented. Development team that was working on a project where three new server environments were installed. The development team provided the most valuable experience in the actual rights management with Centrify, and the knowledge was later capitalized in the rollout phase when rights and roles were defined and tested. During the piloting phase, an iterative cycle was used to develop the authorization roles that would eventually provide all the needed access and execution rights that were needed without undermining the security of systems.

The servers for the pilot were running Oracle Enterprise Linux, and no Solaris servers were in active use during the pilot. That did not turn out to be a major issue in the rollout since the amount of Solaris servers was marginal. Due to that, the Solaris servers were handled case by case.

8.1.3 Testing and validation

Testing was conducted as an ongoing process during the pilot. Anytime the pilot users gave feedback, adaptations and fixes were implemented. Centrify features needed to provide core functionality, security hardening and user management. Those were tested as a whole by the system administrator team.

The testing lacked clear structure and not as organized it should have been, which had an effect on the initial deployment. This was due to lack of deep knowledge on the product and a clear plan what needed to be tested. In general, tests were made to solve any issues that were raised by the members of the pilot and the team in charge of the deployment.

8.1.4 Roll-Out

Rollout plans were created after sufficient testing, and verification with one development team was done during the pilot. The server infrastructure was segmented into logical entities and a rollout plan was created for them. Entities were defined from already running customer environments and internal company servers. Rollout phases themselves consisted of their own projects that contained the same principal phases planning, deployment, and roll-out.

8.1.5 Management and Evolution

Before selecting Centrify solution to be used as the solution for the system integration it was defined that administrative tasks should follow the guidelines of Windows environment. Best practices and known ways of working with AD are extensively documented. Using those principal guidelines in management of an environment such as Qvantel's would prove to be efficient and time saving.

Evolution was not planned beyond server environment roll-out since the need was only to provide a solid base for centralized authentication solution. Centrify provides a wide variety of integration tools to applications that could be used to extend the AD data to web applications, mobile device management and such.

8.2 Technical planning

8.2.1 Planning Active Directory changes

Planning the AD to meet the requirements of Centrify, the deployment was based on information gathered from Centrify and Microsoft documentation. As a general principle, it was defined that nothing should be placed into the root level of the domain

and Centrify data container should not be placed in any active OU or container. Figure 18 illustrates the relation of high level OUs. Centrify OU contains all the information written and accessed by the Centrify applications. Subsidiaries have child OUs that contain users, groups and computers for that specific subsidiary.

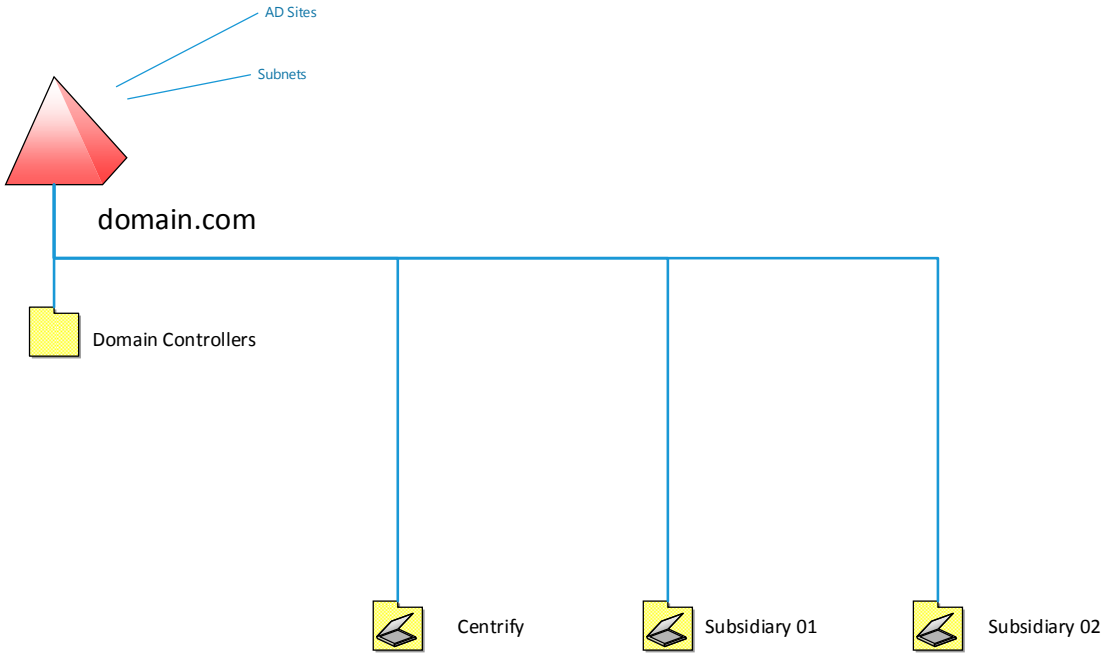

Figure 18. AD hierarchy in a high level.

Access rights regarding AD were planned following normal Windows practices. The initial planning of resources for the management team was at a very high level since there were no pressing requirements that would have forced the separation of administrative access rights between different OUs. Since the planning was made using the best practices, more detailed access management was very easily planned and implemented later on.

8.2.2 Planning Centrify Zones

Zones planning was clear from the start since the structure was similar to AD logical structure at a high-level. The author reflected the administrative duties against the hierarchical zones structure described in the documentation and explained in chapter 6.1 of this thesis. Figure 19 illustrates the high-level plan of the logical structure. Global zone is the highest level that contains the information needed in all environments such as global administrative accounts and roles. The author also wanted to separate office IT environment from our private cloud platform in the logical structure so we planned separate parent zones for those. Office zone contains all users, roles and computers for office environment and Servers zone was dedicated to our private cloud resources. All computer objects were designed to reside in each tenant's zone and no computer objects were deployed to parent zones.

8.2.3 User provisioning

To enhance the security in the access management it was decided to differentiate accounts used in the office system and in our private cloud platform. There was a requirement that UID needed to persist throughout the hierarchy and that was established by planning the provisioning of all users of the Servers zone and its child zones to the parent zone. GID did not need to be persistent since the resources used in the customer zones were unique to the customer. Therefore, groups provisioning was designed so that groups specific to the tenant were provisioned to the respective zone only. Exceptions were groups that were somehow related to system management and those were provisioned based on the actual need during implementation.

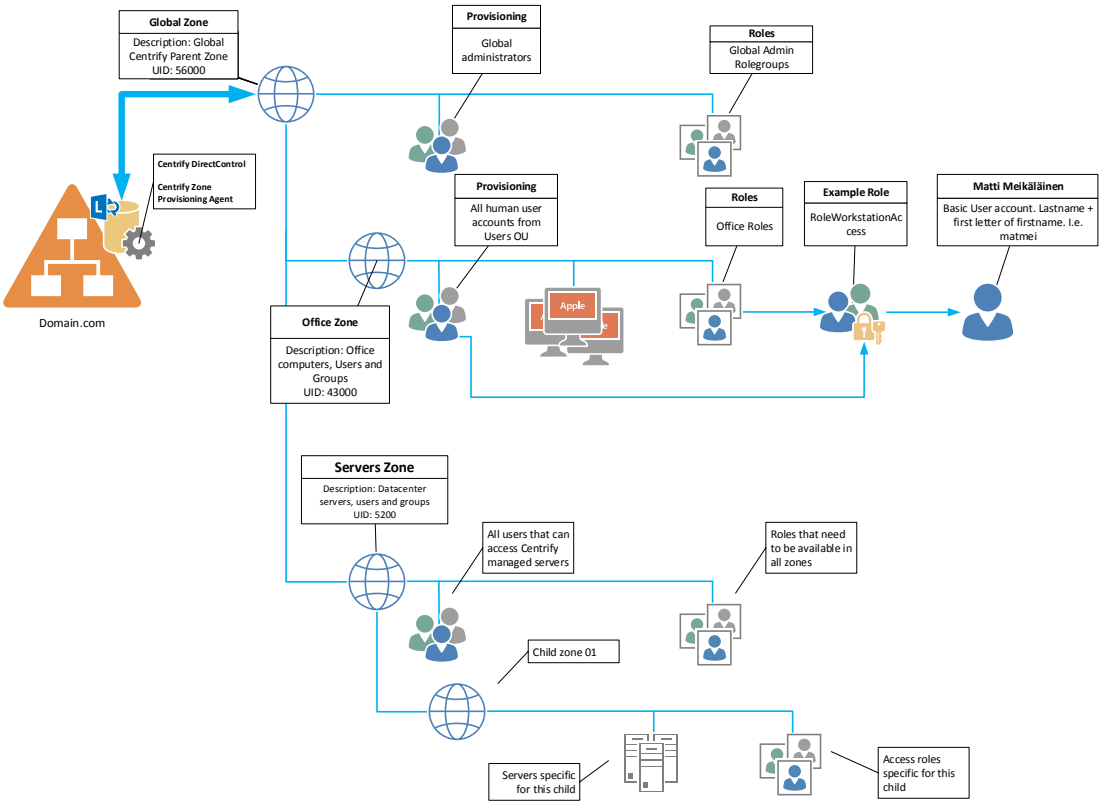

Figure 19. Initial plan for Centrify Zones.

8.2.4 Client deployment

New servers

Qvantel had an automated system for server deployment. Client installation for new servers was planned to be implemented in the deployment scripts. This was a logical choice since we wanted to further automate our operations and server deployment. Deploying client to fresh installs did not exhibit any problems even when implemented.

Existing servers

Client deployment to the existing servers was recognized as one of the most challenging steps in the project so a great deal of effort was put into planning. The servers in Qvantel platforms were deployed in a highly segmented network infrastructure and manual installation of the clients to over 200 servers was not an efficient way for the deployment. The use of Deployment Manager was the obvious choice for doing the deployments and that decision brought requirements that needed to be handled.

- SSH access from the Deployment Manager to servers
- AD domain needed to be found from DNS
- Access rights to servers defined in Deployment Manager

Using the Deployment Manager also presented security requirements that needed to be included in the planning. Deployment Manager has the ability to store account credentials for UNIX login users and services accounts, including the password for each managed computer, in the Deployment Manager database. The passwords are encrypted with the access token of the AD user who adds computers to Deployment Manager. Therefore, for security purposes the following requirements were presented:

- Deployment Manager would not be executed from a laptop. Instead, a server inside our infrastructure shall be used.
- No shared accounts would be used to operate the application.
- A strong password and password enforcement policies needed to be deployed for users allowed to add computer information using Deployment Manager.

Because AD had just been implemented to our network, DNS provided by the DCs was not used on our Linux and Solaris server. During the planning, it was decided to make the DNS changes in phases during the client deployment. Deployment Manager was planned to be installed on a Windows server with all other Centrify applications

and that server was to be used as a management server. After designating the management server, firewall rules were planned regarding the deployment.

8.2.5 Firewall rules

Figure 20 illustrates the primary firewall changes that needed planning regarding the deployment. As stated in chapter 8.2.4, in addition to AD communications, additional openings were needed. These were ssh port tcp/22 to enable Deployment Manager operations on company servers. Investigations were made regarding the list of ports in Table 4, and ports that were not needed were left out the of the communication between Linux clients and AD.

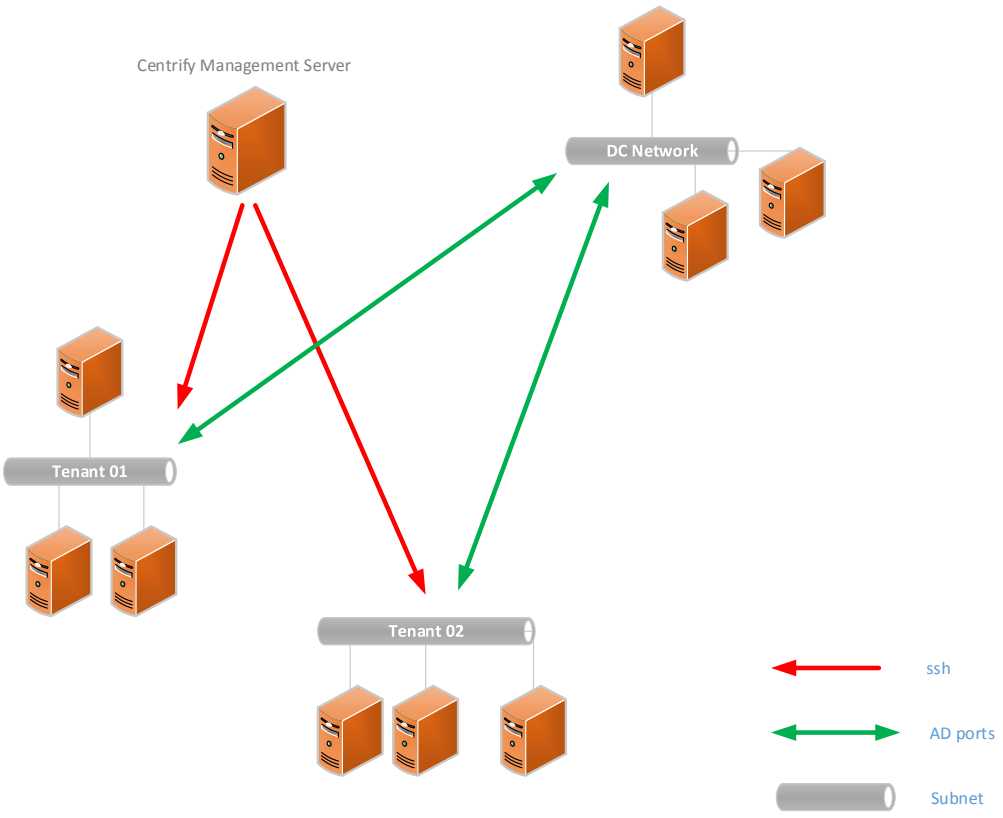

Figure 20. Access through firewall.

When communicating with servers that located in DMZ networks the communication to AD should be limited by placing RODC in an isolated network. The RODCs would

then handle the authentication from the servers in DMZ networks. The risk was acknowledged during the planning; however, due to the nature of our services and network infrastructure RODCs were not implement during the initial deployment but the possibility to implement them in a later phase was not dismissed.

By default, AD replication remote procedure calls (RPC) occur dynamically over an available port through the RPC Endpoint Mapper (RPCSS) by using port 135. Dynamic AD RPC traffic was restricted to a specific port using Microsoft knowledge base article KB224196. This kind of traffic happens between DCs and should be taken into consideration when deploying additional DCs or RODCs.

Table 4. AD port requirements. (Microsoft, 2014)

Protocol and Port	AD and AD DS Usage	Type of traffic
TCP and UDP 389	Directory, Replication, User and Computer Authentication, Group Policy, Trusts	LDAP
TCP 636	Directory, Replication, User and Computer Authentication, Group Policy, Trusts	LDAP SSL
TCP 3268	Directory, Replication, User and Computer Authentication, Group Policy, Trusts	LDAP GC
TCP 3269	Directory, Replication, User and Computer Authentication, Group Policy, Trusts	LDAP GC SSL
TCP and UDP 88	User and Computer Authentication, Forest Level Trusts	Kerberos
TCP and UDP 53	User and Computer Authentication, Name Resolution, Trusts	DNS

TCP and UDP 445	Replication, User and Computer Authentication, Group Policy, Trusts	SMB,CIFS,SMB2, DFSN, LSARPC, NbtSS, NetLogonR, SamR, SrvSvc
TCP 25	Replication	SMTP
TCP 135	Replication	RPC, EPM
TCP Dynamic	Replication, User and Computer Authentication, Group Policy, Trusts	RPC, DCOM, EPM, DRSUAPI, NetLogonR, SamR, FRS
TCP 5722	File Replication	RPC, DFSR (SYSVOL)
UDP 123	Windows Time, Trusts	Windows Time
TCP and UDP 464	Replication, User and Computer Authentication, Trusts	Kerberos change/set password
UDP Dynamic	Group Policy	DCOM, RPC, EPM
UDP 138	DFS, Group Policy	DFSN, NetLogon, NetBIOS Datagram Service
TCP 9389	AD DS Web Services	SOAP
UDP 67 and UDP 2535	DHCP	DHCP, MADCAP
UDP 137	User and Computer Authentication,	NetLogon, NetBIOS Name Resolution
TCP 139	User and Computer Authentication, Replication	DFSN, NetBIOS Session Service, NetLogon

8.2.6 Migration of existing users and rights

The solution enables migration of existing users to AD with rights, and that is established by analyzing configuration files on client servers. However, early on a decision was made not to import any of the existing users. Administrative accounts used in

the environments were left as they are to provide a backup in case problems were encountered with AD authentication. All other accounts were planned to be removed after a transition period when it was confirmed that duties could be handled using the new AD accounts with corrects access rights.

8.2.7 Access rights

Administrative access rights

By default, normal users do not have enough rights to do many of the operations Centrify requires to function. Access rights are managed through AD with native tools like Active Directory Users and Groups (ADUC) or the new Active Directory Administrative Center (ADAC). Regarding administrative access, that much effort was not allocated into planning it since the management team was not so large and the team members had to operate in many different areas, which in a large company would be done in separate teams. A decision was made to come back to this after the implementation was ready and some experience was gathered in the management tasks. It should be noted that administration access is managed by AD so the segregation of administrative duties would not present challenges when implemented if required.

Access roles, authorization roles and rights

When the planning of the deployment was started, extensive knowledge, how developers were actually using the servers was not acquired. Because of that, access rights were not planned on a desired level. There was a vision how access rights were meant to be used; however, planning and implementation was limited by our knowledge in the possibilities of DirectAuthorize. The Access rights were planned based on the vision what was wanted and what would be the best solution for the environment and the team did not focus on whether those would be possible to implement with Centrify.

9 IMPLEMENTATION

9.1 Hierarchical zone structure

The solution relied on hierarchical zones structure of Centrify management. It became clear, based on planning that using two main zones to separate operational environments from our office environment was the best option. Office appliances and workstations were placed in their own zone under the *Global zone* and operational servers were placed in their own zone structure. Both branches were connected via *Global zone* that enables the adding, removal and modification of objects for both branches from one point with a minimal administrative effort.

The operational zone contains *child zones* named after the customer using predefined naming convention. The environment contained servers, command rights, computer authorization role groups, authorization roles and configuration. All user accounts were provisioned to Operational zone. That way the users had the same basic settings (GID, shell, etc.) in all environments joined to AD with Centrify. Only customer environment specific groups were provisioned to the corresponding customer zone using provisioning groups.

9.2 Active Directory configuration

AD structure was prepared to meet the requirements acquired during the planning phase. The logical structure needed to provide a clear separation of Qvantel Group

companies (Qvantel Finland, Qvantel India, and so on) for the administrators to manage different authorization scenarios with the least amount of effort. During the time of planning there was no actual need to separate production and office environments to different domains in the AD forest but the implementation was carried out keeping in mind that in the future the forest could contain more than one domain.

9.2.1 Organizational Units

Centrify stores the information and configuration in AD and for that, it needed a location. Centrify can store the data anywhere in the AD, however, it was recommended by Centrify (Centrify Corporation, 2012, p. 38) that a separate high-level OU was created especially for Centrify and placed at root. Consolidating all UNIX data under one high-level OU-structure saves administrative effort in the day-to-day operations when special instructions were not needed for normal OU management. It also enabled separation of duties without affecting the rights in other OUs.

OUs were added to the domain for Centrify data, groups, users, servers and workstations to form a base for the structure. Figure 21 shows an obfuscated sample of the logical structure that was implemented. The implementation had two basic design rules that were followed:

1. Centrify data is stored outside any Company structure (Centrify UNIX). The contents of the Centrify OU structure are not modified manually but only by Centrify applications, and the functionality is not transparent to the user.
2. Every subsidiary of Qvantel Oy has its own OU structure that contains all the resources that the company owns. That includes groups, computers and users.

Regarding Centrify functionality, security access delegation was made to Centrify UNIX OU to allow Centrify ZPA to make modifications to the parent OU and all descendant objects. This is required for the provisioning application to read and write objects and object attributes to AD.

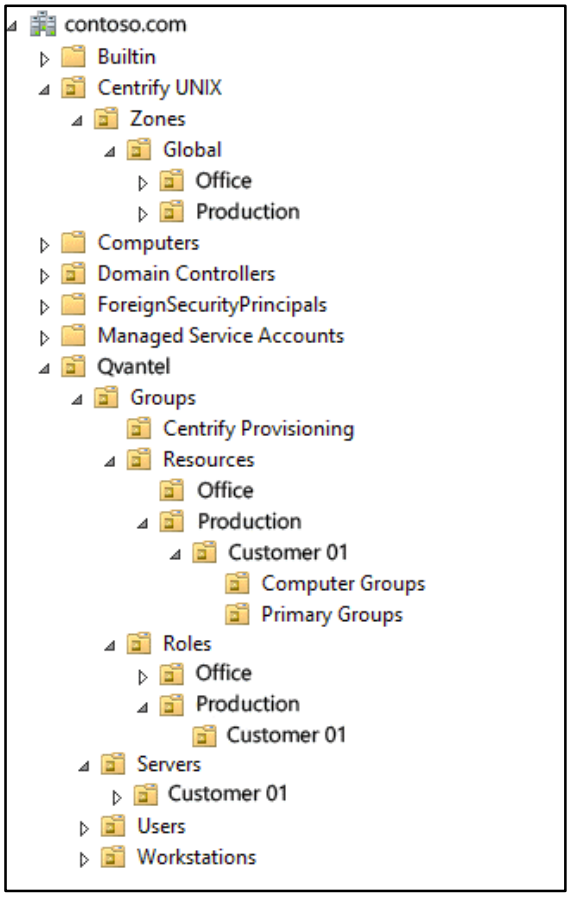

Figure 21. Active Directory logical structure.

9.3 Users, groups and rights management

At start the project access rights management was outlined to follow Microsoft's best practices. The model was defined to be used in Windows environments and that needed some level of planning how a successful implementation could be achieved in a mixed environment.

Users were created in an AD OU that had all the account used in operational environments and users were then provisioned to Centrify hierarchical structure using ZPA. Initially, the users did not have access to any system and all access rights were granted using company access management process where a supervisor defines the role of a user and access rights are provisioned by system administrators.

In the model (Figure 22), every resource has its own AD security group named in a manner that reflects the resource.

- Resources in this context can be databases, file shares, servers or some other module that contain authentication or functionality that need to be included in a role.
- Resource groups are standard AD security groups that relate to the specific resource. A resource can have multiple groups relating to it if there are different access levels such as read, modify or full access. However, one resource group should not be tied to multiple resources.
- Role groups are also standard AD security groups. They contain all the users for a specific role like Financial, HR or IT. In addition to users, role groups are members of the resource groups. The resource groups joined in the role depend on the role and its functions.

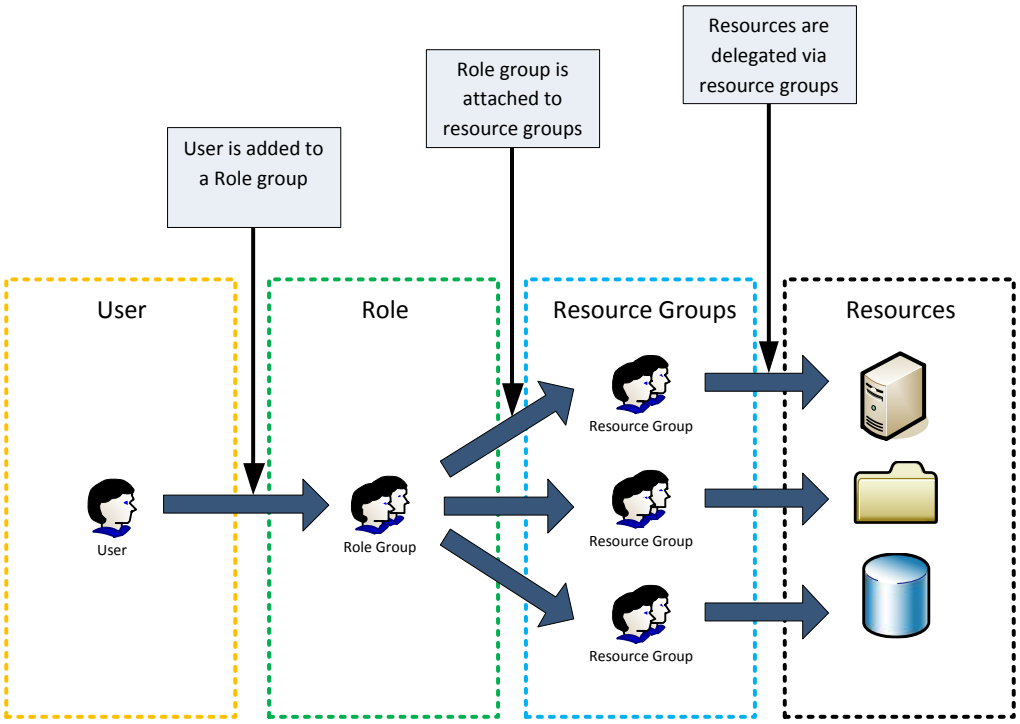

Figure 22. Access rights delegation.

9.3.1 First stage: Resource management in mixed environment

At first, some mistakes were made when Windows practices were followed quite literally in Centrify. That resulted in an excessive amount of security groups since resource groups were created for every right for a server (Table 5), and the same pattern was repeated for 20-30 servers resulting in over a hundred security groups for resources.

Table 5. Resource groups for a server.

Server 1	
Group	**Description**
RES_Server1_Login	Resource group to allow ssh login to the server.

RES_Server1_app1	Resource group to allow execution of sudo commands for app1.
RES_Server1_app2	Resource group to allow execution of sudo commands for app2
RES_Server1_PG_group1	Resource group to do a primary group override for a user.

This was mainly because at the time there was significant time pressure to get centralized authentication implemented and that resulted in decisions that were quick and dirty since there was not enough experience on the product and in its testing. In this model, use of *sudo* was familiar to all in the administrative team and the decision was made to use *sudo* until a proper study was made into using DirectAuthorize. Centrify has the ability to write /etc/*sudoers* configuration centrally through GPO and it was used to configure resource groups to allow specific command in our servers. As a result, the project was able to move forward in although there were problems to be solved before existing servers could be expand to existing servers in our environment. Figure 23 illustrates the implementation using AD security groups and /etc/*sudoers*.

1. Users were members in a role group, which was a member of resources (2).
2. Resources contained all needed rights for the role.
3. Resources were configured to a server using GPO or Centrify DirectManage where configuration was made to the computer object.
4. Access rights for specified role were deployed to server sudoer's configuration or any other configuration file that was specified i.e. /etc/centrifydc/group.ovr for group manipulation for the role in this example.

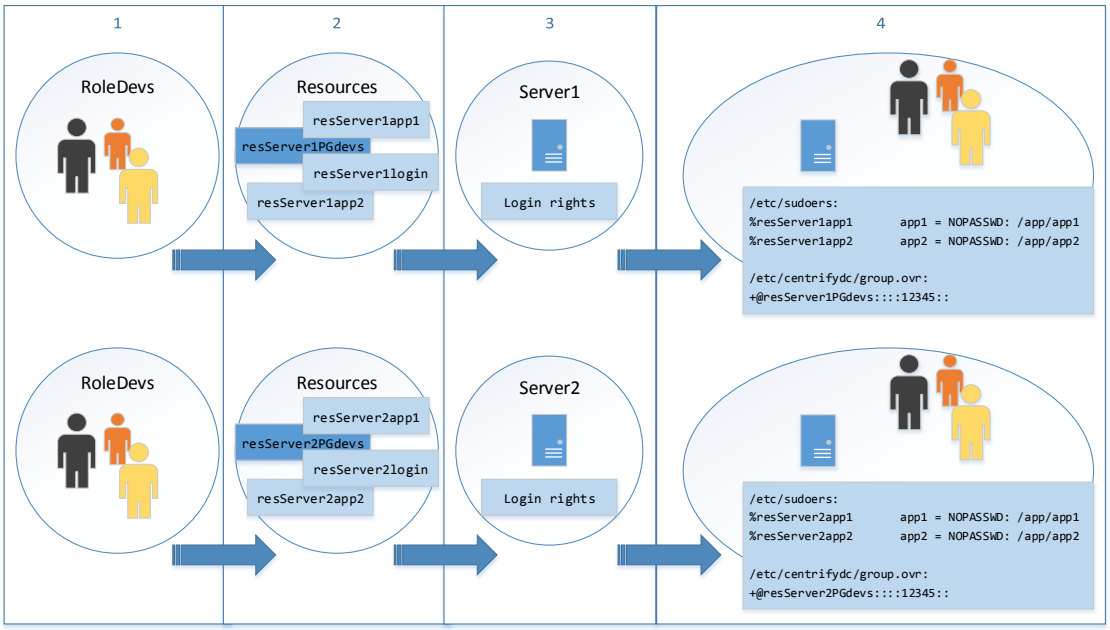

Figure 23. First stage of implementation.

Since *sudo* is managed via sudoers file, which is a local file in a server and cannot be managed centrally, several GPOs had to be created and those were limited to a subset of servers in an OU using security groups. That meant significant amount of micromanagement and things that needed to be remembered e.g. resource groups, GPO configuration, limiting GPOs, binding roles to resources. This work needed to be done when a new server was added or a server was removed.

9.3.2 Second stage: Rights management through DirectAuthorize

Once operational status was established using the quick and dirty solution described in the previous chapter, the development of a more stable solution was started so that the administrators could perform access management with a minimal administrative effort. That meant more testing of granular access and rights management functionality provided by the DirectAuthorize. Centrify client setup includes binaries for the DirectAuthorize itself and also a Centrify customized package of OpenSSH

which was not installed to our environment because that had a negative effect to our existing patch management process.

It was unclear what DirectAuthorize functionality required the custom OpenSSH to be installed to servers. Testing of *dzdo* with different rights and configurations soon revealed that all needed functionalities could be used without the custom OpenSSH package. As a result, design work was started to completely change the implemented access and rights management with Centrify. As an extra challenge planning needed to be done on how the already implemented solutions could be migrated without any major downtimes. A positive feature in this was that only internal users were using the systems so scheduling and planning such maintenance were done with reasonable effort and the impact was not that critical if some problems were discovered.

The new design focused on minimizing administrative effort when changes occur and once the initial work was done, the management would not require a great amount of time and training compared to the previous state. Authorization roles described in chapter 6.5 were used with computer authorization roles (6.3) to efficiently delegate all needed rights for users in the servers. A huge improvement to the earlier implementation was the high granulation of access rights and with this method all the main features of Centrify could be harness to our benefit. The second iteration is visualized in Figure 24.

1. Users are members in AD role groups.
2. Role groups are members in specific resource groups that are used in situations where a functionality was wanted to achieve, which cannot be done any other way like primary UID, or GID overrides which are used in the figure.
3. Authorization roles were built in Direct Access Manager that are like toolboxes for a specific group of people. The role contains all the required commands and PAM roles that are needed.

4. AD role group (1) and authorization role (3) are linked in computer authorization group. One AD role can be linked to multiple computer groups with proper authorization roles and the access rights are automatically distributed to all servers that are members of the specific computer groups.
5. As a result, proper access management that can be scaled with ease.

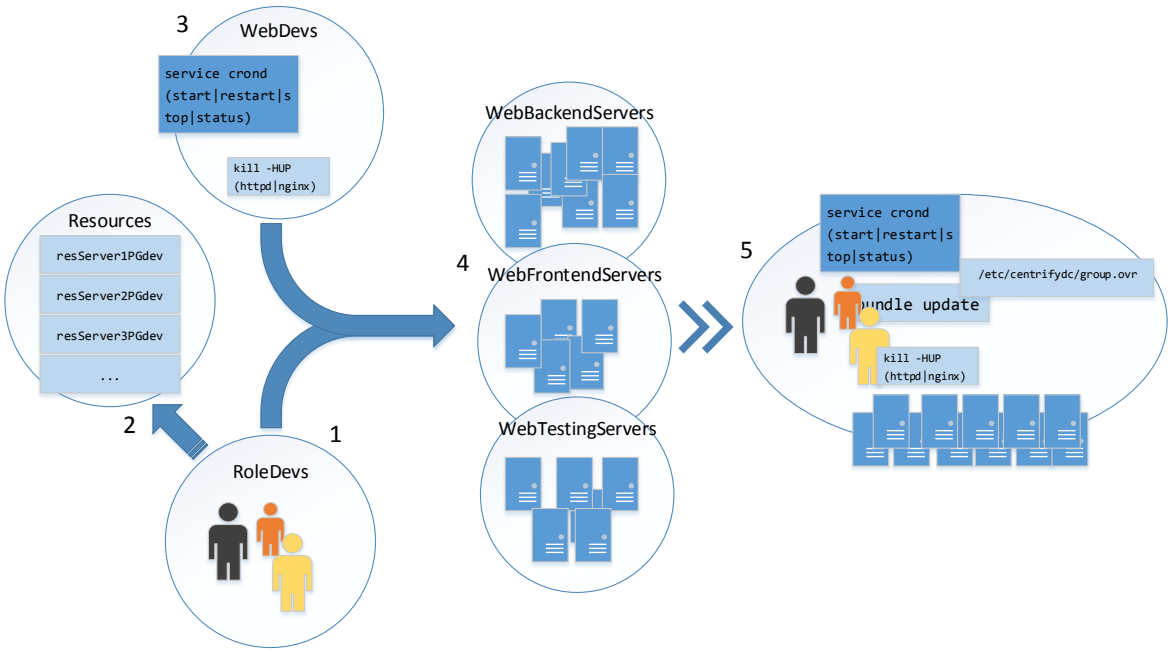

Figure 24. Rethinking access management.

9.3.3 Challenges with sudo to dzdo

Moving away from *sudo* to *dzdo* was slightly problematic since the two applications were not functioning identically. Centrify states in their documentation that *dzdo* should be a replacement for *sudo* but it was not. There are fundamental differences in how the application handles certain syntaxes used in scripts. Behavior is different in some cases because of a bug in sudo version that dzdo is based on. After lengthy discussions in with Centrify support, they acknowledged the bug and provided the following workaround:

```
[user@server]# sudo -iu testuser /bin/sh -c "echo test"
test

[user@server]# dzdo -iu testuser "/bin/sh -c "echo test""
(empty output)
```

Workaround is to use additional single quotes.

```
[user@server]# dzdo -iu testuser "/bin/sh -c 'echo test'"
test
```

Also, *dzdo* differs from *sudo* so that it wants the first command as a single parameter and the rest of the command line as the second parameter. An example illustrates this:

```
[user@server ~]$ dzdo -i -u web cd "/tmp && pwd; id -nu; umask
-S | grep -q \"g=rwx\" && echo ok"
/tmp
web
ok
```

The developers noticed challenges with *dzdo* quickly when automated scripts ceased to function properly and workarounds were implemented. All of the problems were somehow related to escaping of quotes or processing parameters of the command like explained above. Further problems in the behavior of *dzdo* have not been discovered after running the solution in production for almost two years.

10 RESULTS

10.1 Overall experience

Overall, the project was a success and it provided the administrative team tools to manage large amounts of users with a reasonable effort. The amount of configuration work at first was overwhelming due to misinformation and lack of experience, however, eventually solutions were found to solve the problems when the team got familiar with the new products. In total, the project took 1.5 calendar years to see through since the experts were not dedicated only to this project and there was no pressure to get it completed in a tight time frame. This chapter presents the results gained and findings made during the project.

10.2 Experiences from the planning

Initially, the vision was clear what achievement was wanted in the end. The author had years of experience from Windows access management and infrastructure architecture which in the author's opinion was far ahead what could be achieved using only Linux systems in terms of easy access management. The challenge in planning was the Centrify product and the lack of knowledge regarding it. How did it need to be integrated? Could it do what was done with Windows? How did the rights management actually work with our server infrastructure? There were many questions floating around and there were little facts to provide the answers. The decision to follow the same ideology used in Windows environments was the supporting idea from the beginning of the planning and that paid off in the end.

The planning eventually presented two major challenges that were the role access planning and the existing environment rollout. It was difficult to plan the access roles

at first since the administration team did not have a good visibility what the developers were actually doing in the environment and what kind of access rights were required, which resulted in planning the access rights at quite a high level and many adjustments were made during and after rollout. A better understanding of developer activities would have helped a great deal when designing access roles.

Lack of competence regarding Centrify Suite was a major factor that affected the entire project in addition to just planning. No experience existed on the product or its capabilities in the scale that was visioned. The software provider offers face-to-face trainings and web trainings and the author highly recommends participating in a training before planning the deployment because it saves plenty of time and effort.

10.3 Implementation challenge

Naturally, the challenges faced during the planning phase affected the implementation by extending the project life span. Implementation was carried out in iterations since features and changes needed to be tested in a real environment with real users. Some level of testing was done on the functionality before deploying the software, however, most of the functionality regarding scripts and file system accesses was not tested during initial implementation. Since a new infrastructure was being built for a client at the same time, it proved to ease the implementation phase since the test could be done at the same time as building the solution in a new environment that was not in production during the implementation. That way there was no danger for data being destroyed or compromised or people being blocked from doing their work. With the first environment, the iterative approach was excellent for gathering experience and practices that were capitalized during deployment to existing environments.

The second set was quite similar to the first, and many of the configuration practices adapted in the iterative phases could be used. Since this was the first actual deployment to a group of servers that were in production, development had to be done to our process to include all details that would have somehow affected the deployment and applications in production. Lessons were learned from our initial installations and that was capitalized amazingly well when planning the access roles. At the same time, roles were simplified even more; thus, access management would be simple; however, roles were still providing the needed level in access segregation.

The last big push included the oldest part of the managed environment and it contained the most challenges since the infrastructure was quite old. It had remnants of a time when processes were not followed properly and practices were under developed. Adaptations were made on methods from the previous two stints and the solution was gradually deployed. Because there were servers that were not based on our server templates problems were faced when deploying configurations and settings. There were different Linux distributions that did not have the same configuration files in use and that resulted in unwanted functionality. Most problems were faced with servers that were used by developers for testing and staging. On those computers, users had various access configurations in place. The negative functionality of the legacy servers resulted in a new project to replace all of the outdated servers with new installations that were based on managed template. This was the most time consuming of all deployment phases while having significant pressure due to other projects.

10.4 Influence to access management

A centralized solution for access management provided a way to integrate the mixed environment into a centralized authentication and administration point that provided more than access management. The solution brought adherence to company security policy that required extensive effort to uphold earlier. It also introduced

lower costs through economies of scale and automation that were the primary goals in the beginning of the project. Simple access management tasks did not show a great improvement in the time spent to resolve them; however, in a situation where a large amount of access rights had to be delegated, the solutions quickly showed their worth. Especially, when new environments are created, most of the access management effort is done in the configuration phase and afterwards the management is trivial work that can be assigned to typical first or second tier experts. This frees the system administration team to other tasks that require that level of expertise.

In chapter 7.4, the author studied the administrative effort that was allocated to access management duties and the research goal of this thesis was to see what implications the solutions would have to management costs and resource usage. The author gained an extensive amount of experience in managing the access rights requests since its main responsibility resided on him. The tasks that contained access requests and the configuration was already implemented were trivial after the solution implementation regardless of how many users or access rights were to be delegated. The time spent varied from five to 15 minutes, which was significantly shorter than before. After an environment reaches a level of maturity where configuration effort is almost nonexistent, the operational cost will be very small and the solution starts to redeem the capital expenditure.

Without the centralized authentication solution, the company would have needed an extensive amount of administrative effort to cope with the rapid growth. When the project started there were around 100 employees working in the company and most of them had some level of access to servers now managed by the solution. Since then the company has grown to over 250 employees and has external users that also need to be managed. The amount of servers has grown at an average rate of 100 servers per year. In the light of these numbers, it very easy to conclude that without a centralized authentication solution it would have been incredibly difficult to adapt to the

workload and that would have meant additional recruitment for access management tasks.

Another aspect is the ever-growing information security awareness that has been fueled by the major events in the recent years like Snowden leaks and major breaches in security like Target and Sony cases. Customers in the telecommunications sector have always had a high level of information security awareness; however, they have raised the bar to a new level. To deal with audits the organization must have the capability to present that user and access management are handled systematically and securely. Segregation of duties is necessary, and there needs to be a capability to show evidence that it is attended to. Centrify solution provides excellent tools for the company to comply with the requirements received from customers and regulation.

10.5 Further development

This thesis focused on bringing Centrify IAM solution in the organization's infrastructure to enhance the access management. The focus of the project was the deployment and reaching a steady state in operations. Since then reviews have been done on other features that the supplier is offering to expand the capabilities and one of the most interesting features is the audit feature presented in the Enterprise edition of Centrify Suite. The audit capabilities in our environment should be enhanced, and the audit feature would probably be a suitable addition to our monitoring capability.

Another property, which the infrastructure is lacking regarding access management, is reporting. Centrify suite has built-in reporting features; however, to fully gain the potential from the reports they need to be automated and formatted to serve the purpose. Reports could be sent to supervisors automatically from Centrify for access rights review without the need for manual actions.

Automatic access downgrade would be a feature worth investigating. If a user has not used a specific access, it would send notifications to the user that access rights will be downgraded if user is not using them. After a specified time, the access right for that system would be set to an agreed minimum or removed to prevent users from “collecting” unneeded access rights.

REFERENCES

Bertino, E., & Takahashi, K. (2011). *Identity management : concepts, technologies, and systems.* Boston, London: ARTECH.

Centify Corporation. (2015). *Centrify Customers*. Accessed on 1 March 2015. Retrieved from Centify Corporation: http://www.centrify.com/customers/

Centrify Corporation. (2012). *Centrify Suite 2012 Planning and Deployment Guide.* Accessed on 12 March 2014, from Retrieved from Centrify.com: http://www.centrify.com/downloads/products/documentation/suite2012/ga/centrify-dc-deployment-guide.pdf

Cheswick, R. W., Bellovin, M. S., & Rubin, D. A. (2003). *Firewalls and Internet Security, Second Edition.* Addison-Wesley.

Cowley, C. (2013). *Integrating RHEL With Active Directory*. Accessed on 4 January 2015. Retrieved from Just Another Linux Blog: http://www.chriscowley.me.uk/blog/2013/12/16/integrating-rhel-with-active-directory/

Finn Partners. (2014). *Centrify Survey.* Accessed on 1 March 2015. Retrieved from http://www.centrify.com/downloads/public/Centrify-Password-Survey-Summary.pdf

Geisshirt, K. (2007). *Pluggable Authentication Modules: The Definitive Guide to PAM for Linux SysAdmins and C Developers: A Comprehensive and Practical Guide to PAM for Linux: How Modules Work and How to Implement Them.* Retrieved from Books24x7: http://common.books24x7.com.ezproxy.jamk.fi:2048/toc.aspx?bookid=25591

Gillham, B. (2000). *Case study research methods.* London, New York: Continuum.

Hudson, S. (2012). *Centrify Private Vendor Watchlist Profile: Bridging Linux/Mac/Mobile Platforms for Increased Security and Managed Access Control via Microsoft AD.* Accessed on 1 March 2015. Retrieved from IDC Vendor Profile: http://www.centrify.com/media/1549/idc-vendor-profile-centrify.pdf

Kirkpatrick, G. (2008). *Authenticate Linux Clients with Active Directory*. Accessed on 2 January 2015. Retrieved from TechNet Magazine: http://technet.microsoft.com/en-us/magazine/2008.12.linux.aspx#id0060006

Lukka, K. (2001). *Konstruktiivinen tutkimusote.* Accessed on 23 April 2015. Retrieved from Metodix.com, Menetelmäartikkelit: https://metodix.wordpress.com/2014/05/19/lukka-konstruktiivinen-tutkimusote/

Micki , K. N., & Tipton, H. F. (2012). *Information Security Management Handbook, Sixth Edition, Volume 5. Chapter 4 - Privileged User Management.* Retrieved from Books24x7.

Microsoft. (2012). *What is an RODC?* Accessed on 8 September 2015. Retrieved from Microsoft TechNet: https://technet.microsoft.com/en-us/library/cc771030(v=ws.10).aspx

Microsoft. (2013). *Active Directory Domain Services Overview*. Accessed on 26 February 2015. Retrieved from TechNet: https://technet.microsoft.com/en-us/library/hh831484.aspx

Microsoft. (2014). *Active Directory and Active Directory Domain Services Port Requirements*. Accessed on 9 April 2015. Retrieved from TechNet: https://technet.microsoft.com/en-us/library/dd772723%28v=ws.10%29.aspx

Microsoft. (2014). *Active Directory Structure and Storage Technologies*. Accessed on 26 February 2015. Retrieved from Microsoft TechNet: https://technet.microsoft.com/en-us/library/cc759186%28v=ws.10%29.aspx

Moen, R., & Clifford, N. (n.d.). *Evolution of the PDCA Cycle.* Accessed on 23 March 2015. Retrieved from http://pkpinc.com/files/NA01MoenNormanFullpaper.pdf

Morgan, G. A. (1997). *Pluggable Authentication Modules for Linux*. Accessed on 3 January 2015. Retrieved from Linux Journal: http://www.linuxjournal.com/article/2120

National Institute of Standards and Technology. (1997). *An Introduction to Computer Security: The NIST Handbook.* National Institute of Standards and Technology.

National Research Council. (2003). *Who Goes There?: Authentication Through the Lens of Privacy.* National Academies Press.

Petersen, R. (2008). *Linux: The Complete Reference, Sixth edition.* Retrieved from Books24x7: http://common.books24x7.com.ezproxy.jamk.fi:2048/toc.aspx?bookid=23662.

Qvantel. (2015). *About Qvantel*. Retrieved 1 March 2015. Retrieved from Qvantel: http://www.qvantel.com

SANS Institute. (2014). *Critical Security Controls for Effective Cyber Defense*. Accessed on 27 February 2015. Retrieved from https://www.sans.org/critical-security-controls/

Schneider, N. (2003). *Linux Authentication Systems*. Retrieved from Linux Geek Net: http://www.linuxgeek.net/documentation/authentication

Solomon, M. G., & Kim, D. (2012). *Fundamentals of Information Systems Security.* Jones & Bartlet.

Srivistava, V. (2009). *Understanding and configuring PAM*. Accessed on 3 January 2015. Retrieved from http://www.ibm.com/developerworks/library/l-pam/

Walla, M. (2000). *Kerberos Explained*. Retrieved from Microsoft Developer Network: https://msdn.microsoft.com/en-us/library/bb742516.aspx?f=255&MSPPError=-2147217396

Verizon Enterprise Solutions. (2015). *2015 Data Breach Investigations Report.* Retrieved from Verizon Enterprise: http://www.verizonenterprise.com/DBIR/2015/