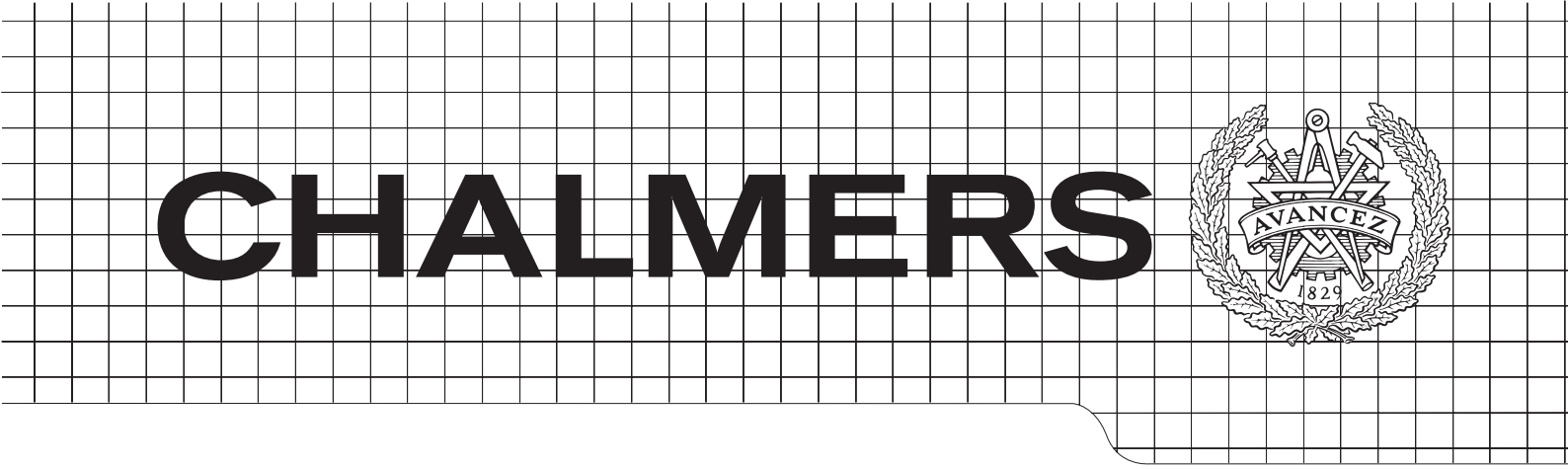

Towards Secure Migration to Cloud Environment

Master's Thesis in Computer Systems and Networks

GUSTAV FREIJ

Department of Computer Science and Engineering
Chalmers University of Technology
Gothenburg, Sweden, 2014
Master's Thesis 2014

GUSTAV O. FREIJ

Examiner: Tomas Olovsson

Chalmers University of Technology
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Department of Computer Science and Engineering
Göteborg, Sweden June 2014

Acknowledgements

I would like to thank my supervisor at Acando, Mattias Liljenberg, for all the help and guidance with my thesis, and for letting me do my work at Acando. I would also like to thank Jonas Grundén, integration consultant and expert at Acando, for all the technical help and problem solving that have been needed during my project. Lastly, I would like to thank my supervisor Zhang Fu and my examiner Tomas Olovsson at Chalmers, for all their help.

Gustav Freij, June 2014

Abstract

As the amount of data traffic increases around the world, development of bigger and more effective systems is needed. Cloud services introduces the possibility of running systems in a scalable and flexible environment. Instead of upgrading On-Premise systems to fit the new demands of data flow, migration of these systems and their corresponding services to the cloud is a good option to be considered. However, migrating systems to the cloud might introduce security questions and reservations, as the data isn't located and controlled On-Premise any more.

This thesis aims to show that a system migrated to the cloud, is at least as secure as hosting it On-premise, when a first step towards a migration is performed. This step is implementing a hybrid cloud solution, based on Windows cloud service called Azure, where the type of the migrated system is of integration nature based on Microsoft's integration platform BizTalk.

The cloud services are still rather new on the market, which in practice means that there still are room for development and standardizations for security aspects. If having no rush into the cloud environments, small steps towards a hybrid solution really seems like a good way to go, as this minimizes security implications. The results provided in this thesis, shows that a hybrid cloud solution is as least as secure as a traditional, On-Premise system.

Contents

Glossary

AD	Active Directory
Azure	Windows cloud environment
BTS	BizTalk Server
CA	Certificate Authority
CIA	Confidentiality, Integrity and Availability
DDoS	Distributed Denial of Service
DMZ	Demilitarized Zone
DOS	Denial of Service
DS	Digital Signature
IaaS	Infrastructure as a Service
IIS	Internet Information Services. (Microsoft's web server)
IPSec	Internet Protocol Security
MAC	Message Authentication Code
MITM	Man-In-The-Middle
PaaS	Platform as a Service
SaaS	Software as a Service
S/MIME	Secure Multipurpose Internet Mail Extensions
SSL	Secure Sockets Layer
VM	Virtual Machine
VPN	Virtual Private Network
WAAD	Windows Azure Active Directory

Chapter 1

Introduction

Integration systems are widely used around the world to transfer sensitive data between different types of business or applications. As the amount of data increases, these systems sooner or later need to be upgraded or replaced which requires large investments. Instead of upgrading to a new system with better performance possibilities, cloud computing could be an option. The concept of cloud computing shortly refers to moving applications, services and even physical resources, from a local environment of the stakeholder to a remote computational environment. When moving a system to a cloud environment, it is referred to as *migrating* the system.

The performance issue of growing systems stated above is solvable with cloud resources. Since these resources are scalable, the cloud architecture works in a way that it scales the resources to satisfy the demands as they increase. When making use of this advantage of cloud computing and the corresponding scalability, maintenance of the in house servers where the systems are implemented, is another aspect that isn't needed any more. Or at least, not in the same extent.[1]

1.1 Background

Among the integration solutions, cloud-based ones are having lots of security concerns, both from the academic and industrial level of view. These cloud-based solutions are rather new on the integration market, and are now starting to compete with the traditional solutions, known as *On-Premise* solutions. These traditional solutions are usually installed on physical/virtual servers located within the company network, and they act as an information hub and a point of delivery for external systems belonging to customers and business partners. A cloud-based integration solution though, is either run on servers at a cloud vendor or run as a col-

lection of services which together implement and form an integration solution[2][3].

When moving data and computation to the cloud, questions regarding security often become a large topic. People often feel secure about their own physical network with firewalls and encrypted communication, but when they are expected to distribute their data into the cloud, reservations often arises.

An integration system is, as the name refers to, a system that integrates different systems to be able to communicate and make transactions between them. Most often, the companies that need to communicate with each other have different systems. When this is the case, an integration system is needed where a special component, the *integration broker*, is a vital part. The integration broker might be seen as the heart of the integration, which acts as a middle-ware that transforms and maps all messages sent among different systems and applications.

Companies have different requirements regarding security and different demands regarding how data is handled and transformed into their specific system. In most of the cases, these companies host and maintain their own servers for these computations, which is a rising cost that follow the success of the business[1].

1.2 Problem Description

A vital part with integration systems is security, which can be categorized in several ways[4]. For example, data transportation security is one type, while how the hierarchical user systems with different rights at the same domain is another[5]. Although these security examples seems trivial, there isn't any clear guidelines or rules of how to make a integration system secure, to the best of my knowledge. The security level differs from system to system, as there are many different aspects that is needed to be taken into consideration for each specific setup[6].

The process of migrating an On-Premise integration system solution to the cloud presents new hindrance in how developers implement different parts of the system. The Confidentiality, Integrity and Availability(CIA) security aspects need to be expanded and wisely delimited in a beneficial way for migrating to the cloud, in order to achieve a better understanding of how an On-Premise integration solution might be able to migrate to the cloud.[7]

As integration systems often connects two or more companies, known as business-to-business systems, where sensitive data is transferred between them, one of the most important requirements is to send data in a secure way. In most cases, differ-

ent companies have different security needs for how data should be handled, which might create conflicts between the endpoints.[4] Another challenge is to achieve security within the companies borders, even if those borders are extended to the cloud. Since data in the cloud often is stored and processed in a shared environment by the nature of cloud computing, identity management and access control is of high interest and a big challenge of cloud computing. Encryption and how the corresponding keys for this are calculated and handled is another big issue, as the companies need to trust the vendors, which is not easy to achieve.

The main challenges addressed in this thesis are:

- **Different systems/companies:** Different companies use, in most of the cases, different underlying systems. This is a challenge as these systems need to be able to interact with each other.
- **No guidelines:** As far as my knowledge takes me, it doesn't exist any guidelines for migrating an integration system based on BizTalk to Windows Cloud environment Azure.
- **Complex systems with many components:** As integration systems most often consist of many underlying subsystems, the complexity of the systems as a whole is arising. Any deeper inspection of these systems, with security in mind, is hard to perform as the developers may not reveal any sensitive information about how these systems are built. The same goes for Windows Azure.

1.3 Purpose

The purpose of this report is to study security aspects of traditional On-Premise integration systems when, partly or as a whole, migrating it into the cloud. A migration in this study, refers to the act of moving functionality and/or applications from a On-Premise integration system, to a cloud vendor. The intention of the work is to give a systematic analysis and a clear understanding in the security performance of systems based of a On-Premise nature versus a system based on Cloud services.

The report also aims to provide help in the decision making when cloud services is up on the board, how a first natural step towards cloud migration might be done, together with general aspects and challenges when decided to evaluate the possibilities for migrating. In this thesis, a ”natural step” refers to using the cloud only as a transportation medium, and not actually making use of any other types

of services such as data storage or hosting applications. Lastly, the outcome from the report will be provided to Acando Consulting AB.

1.4 Scope

The project is initially delimited by Acando Consulting AB, since what type of integration system that should be analysed is already decided. This integration system is based on Microsoft BizTalk, which is an integration platform widely used around the world. There are clear demands for migrating BizTalk and the corresponding components, partly or as a whole, into the cloud but security questions may become barrier.

The intentions for the case study presented in chapter 4 is to provide a high level view of how to take a first natural step of migrating a system based on Microsoft BizTalk. What security challenges that should be addressed is given, as well as approaches to the CIA security aspects are provided. Comparisons and conclusions with the On-Premise system, shown in chapter 3, will be a large part of the work.

The security analysis will be based on the CIA terms, where confidentiality and integrity will have the main focus. Availability is nowadays not a big problem, since most of the cloud vendors provide an availability rate higher than 99%. Some of the most common protocols when dealing with integration systems and their corresponding servers will be compared, as well as the data storage and manipulation that is needed for them to work in an integration system of today.

An attack model is presented in section 2.2, where possible network attacks, threats and vulnerabilities are provided. This will be used to show how the On-Premise solution, as well as the migrated version, handles different security aspects.

1.5 Summary of work contribution

The contribution of this thesis may be divided into two major parts. The first part is based on a traditional On-Premise integration solution and the second is proposing a migrated version of the same system using Windows Azure cloud environments.

The first part consists of a detailed description of a traditional On-Premise integration system, based on the Microsoft BizTalk integration platform. Along with the description, this part also consists of a security analysis where security

aspects is discussed together with protection from arbitrary network attacks.

The second part consists of a case study, where a hybrid cloud solution is proposed. The case study provides a description of the different components needed, a detailed description of the proposed solution and a security analysis where protection from arbitrary network attacks is shown. This part of the thesis may be used as a guideline when migrating an integration solution, based on Microsoft BizTalk, to Windows Azure cloud environments.

1.6 Related work

In [5] and [8], cloud security and potential issues and challenges are presented and discussed. The majority of the topics in these papers may be used as a base when thinking about migrating a system into the cloud, and all these topics are highly important and will be discussed in the coming years.

In [9], different cloud security problems are presented. Based on the discussion regarding the issues presented, a model-based security approach is presented used to capture different security views in the cloud. The model is elastic, which in this case refers to that every user in this cloud environment is able to configure their own security properties.

In the papers [2], [10] and [11], different views and definitions of the cloud are presented. Due to the fact that a cloud definition is rather complex and differs a lot, discussions and different views of definitions is a must for future research. These different views and definitions are also needed for me to be able to do my work in a great way, with several angels to look at the problem, the depth of my work will be greater.

In [1] and [7], tools and security analysis for migration to the cloud is presented. These papers aims to help the customer when a migration is up on the board. What security aspects that is needed to be taken into consideration, along with pros and cons with different functions that are supposed to be migrated. The papers presents good helping methods and a great way of thinking, which definitely are needed when a migration process is started and up on the board.

In [12], a case study is presented to show how different applications and their performance is changed when migrating these into Azure. Pros and cons are discussed, different obstacles that might arise are also presented and conclusions regarding these are discussed. In [13], one of the security architects of Microsoft,

walks through the security in Windows Azure. This sort of ”guideline” is very appropriate for my work as several of the security aspects that I will work with is presented in a beneficial way. The two papers discussed in this part of the text will be used a lot during this work as they deal with the very same topics that I mainly will look at.

To the best of my knowledge, such a first natural step towards cloud environment, that is given in this thesis, is not provided in any earlier research projects.

1.7 Report outline

Chapter 2 of the report covers the technical background. In the first part of this section, different techniques and concepts are described that have had a important role during this thesis project, as well as needed knowledge for the reader of the report.

In the next part of chapter 2, security requirements for network application have the focus. The CIA triad is shown and explained, together with threats and vulnerabilities for network applications. Finally, an attack model with possible attacks that might be performed to the upcoming systems is presented.

Chapter 3 deals with the concepts of an On-Premise, integration system. Figures and examples is given of how such a system is built and works will start the chapter, and finally, a security analysis is provided. This analysis shows how the CIA security aspects are achieved, together with a simple business scenario.

In chapter 4, a case study of migrating such an integration system to the cloud is provided. The use of the Windows Azure components, and how these are able to interact with the BizTalk Server components is first shown, to be followed up by challenges and what to be mindful of when migrating such a system to the cloud. As in chapter 3, a system overview with figures is given and finally, a security analysis is provided to show how to secure this kind of systems when making use of cloud services.

In chapter 5, interesting aspects found during the work will be discussed and how one should think in general when migrating a system to the cloud, among with where focus should be in future development.

Conclusion and future work is given in chapter 6.

Chapter 2

Technical Background

This chapter provides a technical background for the reader. Section 2.1 focuses on cloud computing in general, the different cloud service delivery models along with different cloud deployment models. Section 2.2 highlights security aspects for the systems handled in this Master's Thesis, where the CIA-triad is explained, vulnerabilities and threats are provided along with network attacks that are possible to perform against the systems.

2.1 Cloud Computing

As described in [12], a definition of what cloud computing actually means is rather diffuse. In [12], a long table is presented, showing the view of how different authors defined the cloud in 2008. Based on this table, a proposed definition of the cloud is presented as follows: "Clouds are a large pool(such as hardware, development platforms and/or services). These resources can be dynamically reconfigured to adjust to a variable load(scale), allowing also for an optimum resource utilization. This pool of resources is typically exploited by a pay-per-use model in which guarantees are offered by the Infrastructure Provider by means of customized Service-Level Agreements." The definition in [2] is slightly the same, where cloud computing is described as a shared pool of configurable computing resources.

2.1.1 Service delivery models

Within the cloud computing stack, the resources might be delivered in several levels. These levels are referred to as Service delivery models and might be seen as three layers of a triangle. The models have different pros and cons which needs to be taken into consideration when a system is decided to migrate to the cloud partly or as a whole. The three delivery models that exists are called Software as a

Service(SaaS), Platform as a Service(PaaS) and Infrastructure as a Service(IaaS). They are described in the same order later in this section. However, the definition of the models vary widely though, so the main parts of them will be presented.

SaaS

In SaaS, software is provided from the supplier/vendor which the customer could take advantage from. This software is usually applications that are accessible via some kind of interface, such as a web browser or a program. This model is an alternative to locally run applications. The customer does not control or handle any underlying infrastructure of the cloud where the applications are hosted, except from configuration of the application such as user management.[2]

PaaS

In this model, the supplier/vendor provide a set of tools used for developing and managing applications that is or will be created by the customer. The set of tools provided might be seen as a software platform, containing programming languages, libraries and services needed for developing applications. A well-known example is the Google App Engine or Microsoft BizTalk.[12]

IaaS

The IaaS is the most straightforward delivery model which is based on hardware computing resources. IaaS is typically provided by data centers which means that customers doesn't need to maintain the servers and hardware modules. IaaS is characterized by the concept of resource virtualization, which allows the customers to deploy and run their own operating system on top of the virtualized software. [8]

2.1.2 Deployment models

Within the concept of cloud computing, three deployment models have arisen. The main difference between them is the borders of operation, if it is within a company or across several. They are described in the coming subsections.

Private cloud

The private cloud refers to an infrastructure that is operated solely within a single organization and managed by the same or via a third party. The main reason to setup a private cloud within an organization is to optimize the utilization of existing in-house resources along with security concerns, such as data privacy and

and trust that might be a concern when using a cloud infrastructure that is not located within the firewalls of the organization.[5]

Public cloud

The infrastructure that is called public cloud, is the dominant form of the current cloud computing deployment model. This type of cloud computing is used by the general public cloud consumers and the cloud service provider has the full ownership of the public cloud with own policies etcetera. A popular example of a public cloud is Amazon EC2.[5]

Hybrid cloud

The hybrid cloud model is a combination of two or more clouds, any of the types above, that remain unique entities, but are bound together. Organizations use the hybrid cloud model in order to optimize their resources to increase their core competencies, while controlling core activities On-Premise.[5]

2.2 Security requirements for network applications

This section starts with describing the three core pillars of information security. It also includes threats and risks that may harm an integration system and the corresponding company, as well as how different attacks may be performed. The upcoming chapters show how On-Premise systems are built and secured today, as well as how the same security can be achieved when migrating towards a hybrid cloud solution approach.

2.2.1 CIA: The three core pillars

Confidentiality, Integrity and Availability, CIA, is often referred to as the three core pillars in the concept of information security. These three parts are often visualized as one side each of a triangle that entangles the words ”Information Security”.

The CIA security triad is often used as a point of reference when designing and evaluating information system. Every time an application, that handles information in some way, is created or changed, the three CIA criteria must or at least should be addressed.

Confidentiality

The first one is confidentiality, where one definition is ”limiting information access and disclosure to unauthorized users, and preventing access by or disclosure to unauthorized ones”.[14] Confidentiality can be divided into two types. The first type is often related to user authentication and to provide this, some form of unique ID is needed for every user, this along with a password that only the associated user knows, creates trust for the user for the system[3]. The definition in [15] says that confidentiality is ”preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information”, which, from my point of view, summarizes confidentiality in a good way.

The second type of confidentiality is software confidentiality, which refers to trusting the system or application that handles the user's personal data[3]. To maintain the trust of the user, data is often encrypted. Which type of encryption that is used and how the keys are managed are a crucial point of trust. Most often, symmetric encryption is used for encrypting data, since it is significantly faster than the asymmetric ones, which often is applied when changing keys between parties[16].

Integrity

The area of integrity refers to trustworthiness of information resources and is defined as data should not be changed inappropriately, whether by accident or deliberately malign activity[14]. In other words, the data integrity part should guarantee that the data actually comes from the person or entity the receiver think it did, rather than an imposer. In [15], the integrity concept is declared as ”the process of guarding against improper information modification or destruction, which includes ensuring non repudiation of information and authenticity”.

There are two types of integrity that is particular relevant for this thesis, data- and software integrity. Data integrity refers to protecting data from unauthorized deletion, modification or fabrication, where software integrity is protecting the software from the same issues. To achieve integrity, Digital Signatures(DS), certificates and Message Authentication Codes(MAC's) are the main methods used for providing this. How keys for these algorithms are exchanged between parties, together with the selected type of algorithms used for the calculations, are of high interest for the users to feel more confident about integrity of exchanged messages.

Availability

The term availability refers to the property of a system being accessible and usable upon demand by an authorized entity, including the systems ability to carry on

operations even when some authorities misbehaves in the data, software or hardware [3]. The definition in [15] says ”ensuring timely and reliable access to and use of information”, which is putting demands to the provider of the system. A system that is not available at the time you want it, is almost as bad as none at all, as [14] states.

For the customer that uses the system, availability must not only refer to that the system is up and running. It also refers to that the system is scalable, which means that if the amount of data that needs to be processed is having a peak, the system should be able to scale the resources. This prevents the system from being slow and the user experience feels better. In many cases, the system and the data to be processed is time-sensitive, which means that a system that is able to scale on demand is very important[17].

2.2.2 Vulnerabilities

The definition of what vulnerability means differs from case to case. In [18], NIST states the definition of a vulnerability as ”A flaw or weakness in system security procedures, design, implementation, or internal controls that could be exercised (accidentally triggered or intentionally exploited) and result in a security breach or a violation of the system’s security policy.” In other words, a vulnerability is a weakness in some kind of system, which might allow an attacker to reduce the protection of the three core pillars of information security, CIA. To be able to exploit a vulnerability, the attacker needs to have at least one kind of tool or technique that can make use of a system’s weakness. A vulnerability is a security flaw of a system, which can be seen as the surface for the attacker to use[19].

Vulnerability tests should be performed on an ongoing basis. These tests should aim at identifying unexpected security threats as they arise. When such a vulnerability is found, this should be fixed as early as possibly to ensure the minimization of threats and attacks towards these[20].

2.2.3 Threats

A threat is a possible danger that might exploit a weak spot to breach security and cause harm to a system. These threats are mainly created by attackers who wants to intrude into a system and get inside of it, or just wants to bring it down. The attacker might use several methods or technologies for attempting to compromise the security in a malicious way, and the goal with these kind of attacks is to obtain data and information that might, for example, give personal financial gain, industrial espionage for some other company, or just for fun.[18] Analyses of the threats

to a system can help the developers of it to specify security policies, policies that needs to be implemented for securing the resources of the company[20].

Attacks that constitutes threats can be divided into several categories, where two of these are particularly relevant for this thesis, network-and host-based attacks.

Network attacks

External attacks, known as network attacks, are performed by individuals outside of the targeted network/company. With these kind of attacks, there is often a clear goal to be achieved. The external attacks usually involve different types of eavesdropping or identity spoofing to gather information about how to enter the system. While inside the network, attacks based on the Man-in-the-Middle approach might be used to obtain data from certain hosts. Shortly, there are many ways to actually get inside a system and later obtain data from it. There exist several methods and prevention mechanisms for securing a system against these types of attacks[21][18].

Host-based attacks

The internal attacks, known as Host-based attacks, do not need any effort to get into the system, as the attacker already is inside. These kind of attacks might originate from someone that is unhappy with the company, where sensitive information may get sold to some rival company, as an example[21][18].

The host-based attacks do not need to be especially advanced, since the attacker in this case already is inside the network. Attacks, such as Man-in-the-Middle, might be performed between other employees/hosts of the company to achieve information from some other department, which the attacker should not have access to.[18]

2.2.4 Network Attacks

This section covers some arbitrary network attacks which are possible to perform in order to gain access to a network and obtain data from the system while inside of it.

Eavesdropping

Network Eavesdropping, also called network sniffing, is an attack that consists of capture and read packets to and from one or more targeted hosts. The main goal is

to capture sensitive information, such as passwords or any confidential information, that the attacker could take leverage from.[22]

Identity spoofing

Identity spoofing, also known as IP address spoofing, occurs when an attacker is able to determine and use an valid IP address without being authorized to do so. The main goal for this spoofing attack is to identify and chart computers within the chosen network. As routers typically ignore the source IP of the packets sent through, supposing certain rules for this is not set, packets are able to pass through and the attacker might send malicious data in a beneficial way.[23]

Man-in-the-middle(MITM) Attack

A MITM attack is simply, as it sounds, a case where the attacker intercepts a legitimate communication between two parties. As an example, think of an HTTP transaction where a TCP connection is up and running between the hosts. The MITM then splits this connection into two new TCP connections, one for each direction. The MITM attacker then works as a proxy server between the hosts, and is being able to read, insert and even modify the origin data that is supposed to be sent between the two endpoints. The two hosts have no intention to think that the connection is malicious, as long as the MITM sends nothing that seems suspicious.[22][24]

Denial of Service(DoS) Attack

The main goal for a DoS attack is to make resources unavailable for the purpose it was designed. It is an attack that is almost impractical to defend against as the attacker focus on consuming bandwidth and other resources that is needed for the system under attack. When a system receives a very large number of requests of diverse types, as the DoS attack is particularly used for, the system may stop providing correct service to legitimate users. The DoS attack can be extended to a DDoS(Distributed Denial of Service) attack, which improves the power of the attack by using multiple computers or systems to attack one system at the same time.[22][24]

Application-Layer DDoS attack

The application-layer attack is of DDoS nature where the target for the attack is to extinguish special functions or features of a certain application. The attack flood the source with, what looks like, legitimate traffic which should consume all the targeted resources. When the intended functions are exhausted, the attackers

might be able to bypass normal access controls and in that way get in control of the application. When controlling the chosen application, the attacker might read, add, delete or modify data in a beneficial way. The attacker is then able to, for example, introduce a virus to the application which aims to spread throughout the whole network.[22][25]

Port scanning

Port scanning is considered as the first stage of a network attack. The nature of this attack aims at finding open ports and services running in a system, which later can be exploited by attackers to carry out further attacks.[26]

There exists various types of port scanning mechanisms, such as SYN, TCP, ACK, FIN or UDP scan. All these types of attacks are using different vulnerabilities in transport or network protocols. Depending on the answer from the request sent, the attacker that performs the scan is able to determine if the port is open or not.[26]

Chapter 3

On-Premise Integration System

In this chapter, the most fundamental parts of an On-Premise integration system, based on Windows BizTalk, is presented. How Windows and the integration broker BizTalk is used to achieve a secure system is shown, together with how such an integration systems is protected against the attacks provided in section 2.2.4.

3.1 System overview

An On-Premise integration system is defined to as "the process of bringing together subsystems into one system, while ensuring that these subsystems are able to work out together"[27]. In the case of an On-Premise system based on the Windows platform, the BizTalk server, from now on referred to as BTS, acts as the *integration broker*. The main objective for this is to process incoming data, map it into a proper output form and redirect it to the intended destination.

In the coming subsections, the process for sending and receiving data for an On-Premise solution is presented.

3.1.1 Receive data

The BTS cannot, in practice, receive files in a direct way as it doesn't consist of any server software. This leads to that, for every service the system should support, a corresponding server for the specific service is needed, from where BTS collects the data. When data is collected from the server, it is passed through a receive adapter[28]. The receiving adapter task is to read the incoming bytes received, construct a BizTalk message of it and adding basic information, such as transport and receive location details [29].

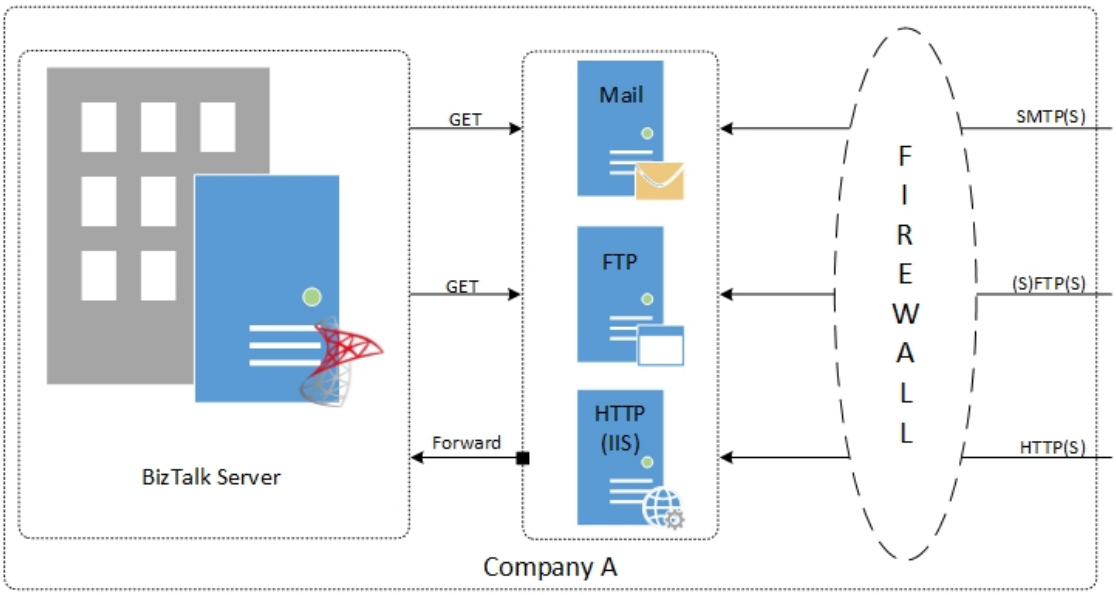

Figure 3.1: On-Premise receive

In Figure 3.1, some common data transfer scenarios are shown. Clients from outside the company sends data or requests to the corresponding server, from which the BTS collects it at a certain point in time. The BTS might poll within specific intervals, or have some certain time set when it should check for new data. When the data is received, this is processed in a suitable way and sent towards the target deeper in the system, which for example might be a database. This approach goes for SMTP and FTP.

Regarding HTTP requests though, the scenario is of another nature. Whenever the IIS(the Web Server) receives a request of such type, it is forwarded to the BTS which processes it immediately. This is because most of the time, these kinds of requests are time-critical, while services such as FTP and SMTP mentioned above most often are not.

3.1.2 Send data

When data is supposed to be sent out of the system, server software isn't needed, as per definition of the client-server nature. In the case when BTS is used, sending adapters are needed to be able to transmit data from the system. These adapters receive a BizTalk message, maps it into a package of correct type and send it through a sending port to the intended target[30].

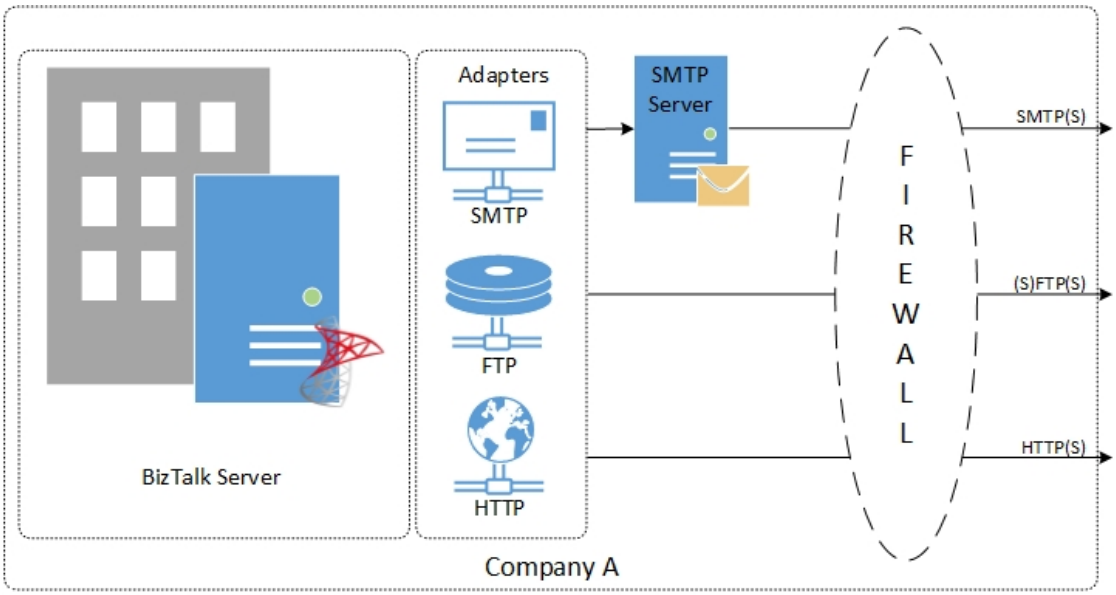

Figure 3.2: On-Premise send

As seen in figure 3.2, data that is supposed to be sent via FTP or HTTP from the BTS, does not need any special processing as the packets is just pushed from the adapters, through the firewall and out from the system.

For data sent via SMTP, the process is slightly different. In this case, the adapter push the data to the SMTP server, which is located inside of the firewall. The SMTP server processes the data in SMTP manner, and then sends it out of the system.

3.2 Security overview

In this section, a security overview with examples is provided. The overview will be of a more general nature, as specific details of any certain system seems unmeaningly in a thesis like this. First out, an example scenario is provided which then will be used in the coming sections when the CIA aspects are treated.

There exists as many different security implementations as there exists solutions to integration systems. By the book, security should be implemented in various ways to achieve assurance to the aspects of the CIA triad. But in reality, the process of securing data and the corresponding transfers is a matter of cost, which in fact often leads to generally insecure systems[31]. When security isn't a must, such as government rules or personal identity handling, security implementations

are often left out.

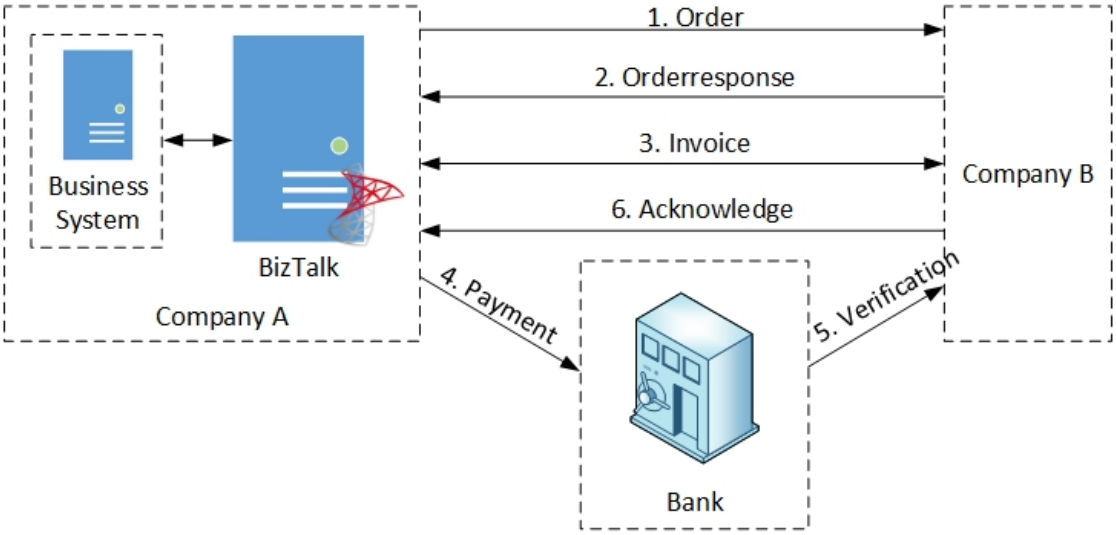

Figure 3.3: A simple reference system

In figure 3.3, a reference system is shown that will be used for the coming sections when dealing with how CIA might be achieved. Company B has a business system of some unspecified type, and has gotten some order request from company A. Company B responds with an order response, which tells if the order is possible or not, together with an invoice if it is ok. The invoice consists of different segments, such as payment method and the amount of money claimed for the order. When the invoice is received at company A, the bank is contacted and the transaction is done, of which a verification is sent to company B, which sends an acknowledgement message to company A telling that everything is ok.

3.2.1 Confidentiality

In the example given in figure 3.3 above, the BTS handles all the messages to and from the system. Confidentiality is most often handled by different types of encryption schemes, where key handling is having a major role.

The BTS supports both sending and receiving trusted data, while securing that the data it processes is secure, as it supports public key encryption for both outgoing and incoming messages[32]. The BTS uses the Secure Multipurpose Internet Mail Extensions(S/MIME) standard for this operation, where both RSA and Diffie Hellman encryption certificates are supported for calculating the public keys. The encryption algorithms DES, 3DES and RC2 are all supported out of the box, while

other algorithms can be implemented by developers.[33][31]

The BTS relies heavily on security provided by certificates, both for signing and encrypting data. Given the example in the figure above, message number 3 and 4 are most likely encrypted, since it contains sensitive information. Company A and company B might have agreed on a key using RSA or Diffie Hellman, that is supposed to be used whenever sensitive information should be exchanged between the parts. If not, they might have public certificates provided by some Certificate Authority(CA), that the counterpart uses for encryption. When message 4 is sent to the bank, company A most likely uses the bank's public key when dealing with the secret payment information.

Confidentiality of data can be achieved in many different ways. When dealing with BTS and integration system though, encryption certificates is the most frequently used method.

3.2.2 Integrity

The demands for integrity is often high, especially when dealing with business systems like the example in figure 3.3, even if it seems rather simple. The knowledge that no one have tampered with data on the way, particularly when dealing with sensitive information, is one of the cornerstones for achieving trust between parties.

As mentioned in section 3.2.1, BTS relies on security provided by certificates for encryption, which also goes for the integrity part[32]. Messages that don't have encryption demands might contain sensitive data that have demands of staying consistent. When encryption isn't necessary, signing is a good way of assuring that no one has tampered with the message.

As for achieving confidentiality, BTS uses the S/MIME standard for signing messages, this goes for the integrity part as well. The BTS supports SHA-1 and MD5 signing algorithms for verifying digital signatures, where the SHA-1 algorithm is used for signing outbound messages. Other signing algorithms, that are not supported out of the box, can be implemented by the developers of the systems.[34]

Parties that need to exchange sensitive data are able to get a certificate for signing messages from a CA, with a private and a public key used to assure the integrity of the exchanged messages. To be able to exchange signing keys and such, the BTS supports RSA, as mentioned in section 3.2.1, and the DSS(Digital Signature Standard)[34].

3.2.3 Availability

High availability for a BTS environment focuses on recovering non functional components that might disrupt the functionality of the BTS deployment[35]. The high availability achievement in this deployment aims at making errors and failures transparent to external applications and systems, to make sure that all services continue functioning correctly with minimal disruption[35].

Constructing a system that provides high availability, involves implementing redundancy for each component involved in the specific system. This approach is applicable for the BTS environment as well, where multiple host instances and clustering the BizTalk Servers are the main focus[35].

By separating areas of functionality into different hosts and tiers in BTS, redundancy can be provided for each host. The deployment of several instances of BizTalk Servers within the same system achieves high availability[36]. Different approaches for clustering the BizTalk Servers exists, such as A/A(Active/Active) and A/P(Active/Passive). In A/A, both clusters are active and load balancing is achieved preventing resources from being exhausted. In A/P mode, one of the cluster acts as a backup that kicks in if the main cluster is stopped for some reason[37].

3.2.4 Security summary

Out of the box, the BTS supports both sending and receiving data in a secure manner. The BTS can handle public key cryptography, achieved through well-known key exchanging methods, together with some of the encryption standards for securing confidentiality of the data passed through the system. The BTS heavily relies on certificates, both for encryption and for signing messages, where CAs might be involved. The developers are able to implement any type of encryption and signing/hashing algorithms that the organization demands, as the BTS is rather free for configuration. For achieving a system with high availability, clustering of BTSs and by that creating a redundant data environment, is also a possibility.

3.3 Attack protection

In this section, protection to the network attacks given in section 2.2.4 is provided for the On-Premise integration solution.

3.3.1 Eavesdropping

Eavesdropping is more or less possible to be done on every flow of traffic. As long as the attacker knows the source and destination IP-address, the flow of information between the two parties can be investigated. If the transported data or the communication link between the hosts isn't secured, this attack can leak highly sensitive information. During an FTP session for example, the user name and corresponding password is sent in clear text in the authentication phase and the upcoming data transfer after this phase is sent unencrypted.

To protect against eavesdropping attacks, one should use encrypted communication between the hosts. Also, the data transmitted between them should be encrypted to achieve even higher security. Both of these prevention mechanisms can be implemented with the BizTalk platform[37].

3.3.2 Identity spoofing

As for eavesdropping, Identity spoofing is also more or less able to be performed on every flow of traffic between hosts. The attack is used to get knowledge of the underlying components within the network that is attacked, which later on can be used for further attacks, such as DoS or MITM.

Protection against Identity spoofing attacks can be achieved by external firewall rules, use the signing/certificate functions that BizTalk provides or SSL to secure all communication that interacts with the BizTalk platform[37].

3.3.3 MITM

As MITM-attacks intercepts communication between two parts, where the attacker acts as a proxy server in between, the hosts does not know that there is someone in between. If there exists a VPN tunnel between the hosts, such an attack isn't possible as long as you don't have the connection credentials. But if not, such an attack is possible and to prevent this, once again the BizTalk function that handles signing and certificates can be used to prevent such an attack. Or at least, prevent that the data sent between the hosts isn't tampered with. If keyed hashing of the messages sent between the hosts is used though, the actual message can be retrieved by the attacker, but not tampered with as long as the attacker doesn't know the key for hashing the message. This may, of course, lead to that the attacker retrieves sensitive information but if information is very sensitive, it should be protected already before it leaves the sending system.[37]

3.3.4 DoS Attack

DoS attacks are used to overwhelm a system, in order to exhaust the resources and preventing proper use of the system. The BizTalk server can be configured in a way such that only known parties are able to send data, preventing unknown parties to send messages to the BizTalk server.

Another protection mechanism for DoS attacks that the BizTalk server provides is size limiting of the messages received. This together with turning off unnecessary features, which reduces the potential attack surface and protects the system for attacks of DoS nature. If the organizations system isn't protected by any hardware of software firewalls, the IPSec protocol might be used here which is able to prevent certain types of DoS-attacks by the authentication mechanisms this protocol involves.[38]

3.3.5 Application-Layer Attack

The Application-Layer Attack is of DoS and DDoS nature, where the difference is that the attack is supposed to exhaust chosen functions of a certain system with the potential outcome of a way in to the system. The protection mechanisms for this type of attack is very much of the same type as for protecting DoS attacks, such as only known sending parties are accepted or size limiting of messages received.

3.3.6 Port scanning

Port Scanning is used to find what ports that are open for external communication, among with achieving knowledge about what services that are used by the underlying system. What type of port scan that is used depends on what type of information the attacker wants.

Protection to port scanning is achieved through the use of a firewall. The firewalls are denying outside access to a network, which is a vital part when dealing with protection mechanisms for network attacks. With this in mind, it is extremely important that unused services, ports and entries to the system are closed, otherwise a potential attacker might get access to a system in an unnecessary simple way.

Consider that some company terminate a deal with some other company. This company have had a certain way through the firewall, trough a specific port. When the deal is terminated and if the port is forgotten to be closed, an potential attacker might find out about this port through a port scan and have a simple way in to the underlying system, as this port isn't handled and checked or checked for traffic

no more. With this in mind, closing ports after closing deals is very important to keep a system secure and to secure it from harmful port scanning attacks.

3.3.7 Attack protection summary

Protection mechanisms for the attacks are possible in most of the given cases. All external connections to the system should be secured in some way, either by encrypting all the data sent or by encrypting the whole communication channel. This together with a well configured firewall will create a good protection for the system. Regarding attacks of the DoS nature though, these are much more difficult to protect against. Size limiting for all the messages sent to the systems might help, along with configuring the firewall in a good way as already mentioned. Closing services, ports and other entries that are not used by the organization any more will not only protect against these kind of attacks, but also for attacks based on port scanning.

Chapter 4

Case study: Towards a hybrid Azure cloud solution

This chapter consist of a case study for migrating an integration system into a hybrid solution, based on Windows Azure. The intention with this chapter is to describe the biggest steps and challenges when migrating a system of integration nature. The outcome of this case study should aim to help and assist in a migration process, where safety aspects and requirements are the main focus.

The main reason why a hybrid solution is chosen to be evaluated is that this transition is the most natural. Instead of migrating the entire system, only parts will be migrated, and the BizTalk server will stay in-house in an On-Premise manner. The hybrid solution is a good way towards migrating a whole set of systems to the cloud.

4.1 The use of Azure

"Azure is Microsoft's application platform for the public cloud".[39] This platform can be used in many different ways, such that for building web applications, store data or creating virtual machines for developing software[39]. For the hybrid cloud solution that is examined in this case study, several components of Azure are used which are presented in the coming subsections.

4.1.1 Execution Models

The most basic thing a cloud platform does is to execute applications. In this case study, the most relevant execution models that Azure provides are Virtual Machines(VM), Cloud Services and Web Sites.

A VM can be created on demand whenever needed. This VM might then be used in any way wanted, such as installing an IIS for hosting web services, using it for cloud services like Office 360 or, for example, setting up a SQL server[39].

4.1.2 Data management

Applications need data, and different kinds of applications need different kinds of data. Azure provides several different ways to store and manage data that addresses the different needs from case to case. The three main ways of storage that Azure support is relational storage(SQL), fast access to simple typed data(NoSQL tables) and unstructured binary large object storage(Blobs). For keeping data stored in any of these ways consistent, the data is automatically replicated across three different computers in the corresponding Azure data center, which host the services[39].

4.1.3 Messaging

No matter how it's used, the underlying code of applications frequently needs to interact with other code. Sometimes the interaction is simple, while in other cases, more complex interactions are required. Azure provides two methods for this out of the box, *message queues* and the *service bus*.[39]

The queue method is of a simple nature, one application places a message in a queue from where the message may be read by another application. Azure queues are designed to support standard queuing scenarios, such as decoupling application components which increases scalability and tolerance for failure, as the server side logs all of the transactions executed against the queues.[40]

The service bus, is in fact also a type of queue which is part of a broader Azure messaging infrastructure. The service bus supports queuing and publish/subscribe methods, integration patterns and web service remoting. The service bus supports ordering guarantee, which not is the case for the message queues described above, as well as mutual exclusion and batched sending.[39][40]

4.1.4 Identifying

Identity handling is a very important process that is needed for letting systems and applications decide how a specific user should be able to interact with the system. As for On-Premise solutions where *Active Directory* is used, Azure provides a cloud based version, named *Azure Active Directory*.[39]

Azure Active Directory works in the same way as most directory services. It stores information about users and the organizations they belong to, letting users login and supplies them with different tokens which are used for proving their identity to parts of the system and different applications. The Azure Active Directory will also let companies to synchronize the user information with the AD running On-Premise in a local network.[39]

4.2 Windows Azure and BizTalk

To be able to make use of Windows Azure and a hybrid integration system based on BizTalk, a few mandatory components are needed. These components include a handful different services that need to be used in order to exchange and transport data from an On-Premise integration system to the Windows Azure cloud environment.

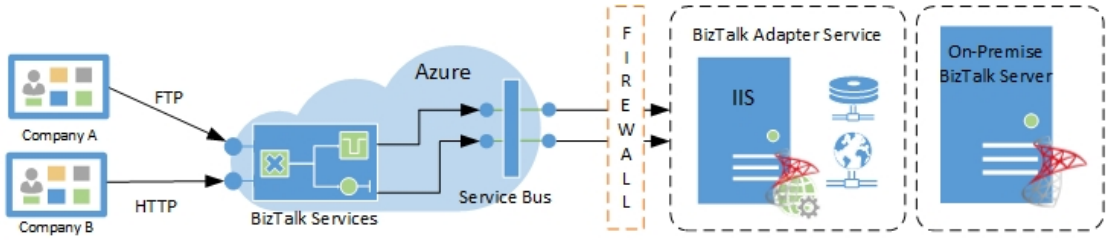

Figure 4.1: BizTalk Services

In figure 4.1, a delivery chain is given where the services needed for integrating an On-Premise system is shown. The three main components that are needed is the Azure BizTalk Services, the Azure Service Bus and the BizTalk Adapter Service.

At a first glance of the components in figure 4.1, it might feel a bit confusing. But in fact, it is only the two cloud components that is new compared to the On-Premise solution shown in the previous chapter. The BizTalk Adapter Service is more or less used in the same way as the BizTalk Adapter Pack is used in the traditional On-Premise solution, but this time they interact with an Internet Information System(IIS), which is the name of Windows web server, for transmitting data towards the cloud instead.

This set of components, mentioned above, are able to provide Enterprise Application Integration(EAI) and Business to Business(B2B) communications through the Microsoft Azure cloud, and is explained in the coming subsections.

4.2.1 Windows Azure BizTalk Services

The Windows Azure BizTalk Services(WABS) is a service in Microsoft Azure that provides means of hosting an integration solution that is running and managed through the cloud. This Windows Azure service was made available for customers in November 2013, and is a cloud platform for integration referred to as Integration-Platform-as-a-Service, IPaaS and has the same basics as a PaaS but with applications focused only on integration. The main components within the WABS are *Bridges* and the *Service Bus*[41][42].

4.2.2 Bridges

The Windows Azure Bridges are used to receive, process and forward messages, both internally within the cloud and externally. The Azure Bridges are able to receive messages through different protocols such as (S)FTP(S) and HTTP(S), which after possible processing, are able to be forwarded in many different ways such as (S)FTP(S), HTTP(S) or to the service bus.

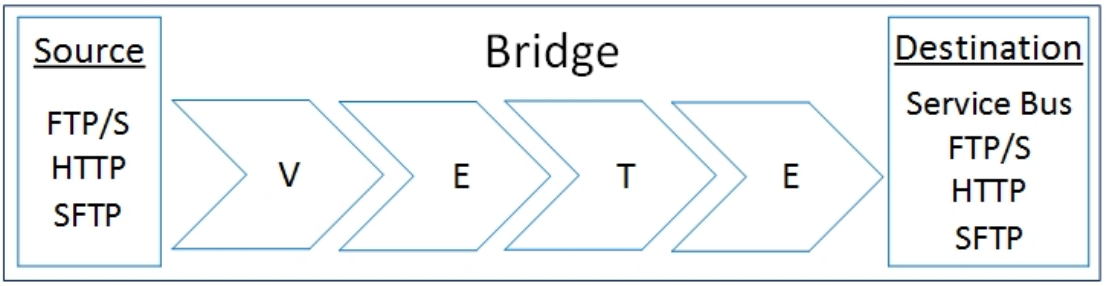

Figure 4.2: BizTalk Bridge

The architecture for the Azure Bridges is given in figure 4.2, where the three main components are shown. A message is received in some way and then pushed forward to the *pipeline*. The pipeline is able to process the message in several steps, where the main functions are *Validation*, *Enrich* and *Transformation*, where the *Enrich* stage can be performed two times. This process is also known as VETE[43]. None, one or all of these functions might be applied on the message passed through the pipeline, and then pushed forward towards the chosen destination which might be a specific server(protocol routing), or the service bus which is explained in the next section.[44]

The ability to process incoming data traffic by using the BizTalk Services Bridge, feels very smart in my opinion. Instead of doing all the processing with the On-Premise resources, lots of work can be done with the Azure resources and different

types of bridges can be implemented and used depending on the type of the message. The ability to just pass through messages without any processing in the cloud is another good aspect with the bridges, this by the simple reason that sensitive data isn't tampered with in the cloud, where trusting issues may arise.

4.2.3 Service Bus

The Windows Azure Service Bus provides communication and interaction between software. This service bus is a multi-tenant cloud service, which means that the service is shared by multiple users, and there exist different communication mechanisms which connect applications in different ways, which are *queues*, *topics* and *relays*.[45][46]

The *queues* that the Azure service bus provides allows a one-directional communication between two or more parts. A sender sends a message to the service bus *queues*, and a receiver picks up that message at some later time.[45][46]

The second component of the service bus is the *topics*, which is of a publish/subscribe nature, allowing one message to be replicated to a set of subscribers that match some specific criteria. The receivers subscribe to a certain topic that is needed or appropriate for their business and when a message arrives to this, it is passed through filters which tells which subscribers that should receive the message.[45][46]

The last component of the service bus is the *relay*. This component provides a bi-directional communication, which not is the case for the queues and topics. The relays is the component of the service bus that is used to communicate with applications and systems outside of the Azure cloud. The service bus relays acts in the application-layer, which means that the communication is able to traverse firewalls and the NAT-protocol with no complications, as each application establishes a TCP connection[45][46].

This kind of mechanism for traversing firewalls in order to achieve a more secure communication link to the cloud, seems very good from my point of view. Since the adapter service and the IIS are doing the work in the application-layer through the service bus relay, the organizations is able to feel more secure since specific openings in the firewalls might be left out.

4.2.4 Windows BizTalk Adapter Service

The BizTalk Adapter Service allows an On-Premise system to communicate with the Service Bus Relay in Windows Azure. The BizTalk Adapter Service uses the

adapters provided by the BizTalk Server, in order to achieve the communication link to the cloud. The BizTalk Service Adapters are hosted in an IIS, and leverage the Service Bus Relay in order to traverse firewalls and NATs, as mentioned in the Service Bus section above.[47]

4.3 Security challenges for migrating to Azure

How to think and act regarding security challenges when migrating a system to the cloud isn't clear at all, at least not from my point of view. When a system is supposed to migrate, lots of aspects are needed to be taken into consideration, and security challenges will arise. The CIA-security aspects must be carefully considered, as these aspects founds the core of information security. When helping an organization to migrate, that specific system must be evaluated for it's specific needs. While some certain aspects are needed in some particular case, these aspects might not be needed when dealing with another system and another case. In my opinion though, some core pillars exist and these are presented below.

Confidentiality

As [51] states, when an organization gives up direct control over security parts and aspects, a high level of trust is needed for the provider of the service. All data that is processed or stored outside some organization's firewall and possibly a DMZ, brings a higher level of risk compared to hosting the system On-Premise. As the physical security is lost when migrating, no knowledge of exactly where the resources are running is concerning[5]. The organization must trust the provider, where a potential threat is that insiders from the provider might access the system hosted in Windows Azure. The level of trust is more or less based on the amount of direct control the organization is able to exert on the external service provider and the corresponding services[51].

Identity management, data sensitivity and privacy of information are major issues when migrating to Azure[5]. The challenge with mutual authentication is addressed in [9], as the identity and authenticity management does not extend to the cloud. One way to achieve extended authenticity is by using two authentication systems, one internal hosted On-Premise and one external which is hosted in Azure[51]. With such a solution, the external and internal identity management is separated from each other. It is of high interest for organizations to have a fully working and secure authentication system cause if not, the users that exploits the system can have incorrect rights which might cause harm to the organization. One, well-configured, or two AD are needed to achieve secure and good identity

management for a hybrid Azure solution.

Integrity

Data that is needed to be stored at a cloud vendor is, by the nature of cloud computing, stored in a shared environment. The level of trust and how identities are managed, both discussed above, is of high interest for the organizations to feel confident. Diverse types of access control keeps data away from unauthorized users, the same goes for encryption. Data must be secured while at rest, in transit and in use, and access to the data must be controlled[51]. To achieve this, key distribution is something that needs to be questioned. How keys are calculated and where the keys are stored and how, is of high interest for an organization that is planning to migrate a system to Azure. Another issue for protecting data is the actual location of where the data is stored. The provider must be able to show where the data and the possible replicas are stored, in order for the organization to feel secure about not breaking any laws for data storage [5]. Once data crosses a national border, it is extremely difficult to guarantee protection under foreign laws and regulations[51].

Availability

Availability is a term that is sensitive for organizations that are planning to migrate to the cloud, as this is rather important for having the organization working at all. If a system or data isn't reachable, it might yield devastating implications. Even with 99.999% reliability of a system, the yearly downtime will be 0.0876 hours[51]. If the data is consistent and redundant among many data centers, this will not be a problem since traffic will be routed to another destination when the main replica is repaired. If the provider only have one replica of the data though, problem will arise, both for the provider and for the organization that trusts it[9].

4.4 Migrated system overview

In this section, an overview for the migrated hybrid integration system is presented. How communication between parties is achieved, where the process of sending and receiving data is shown from a higher point of view. How security is achieved is then presented in section 4.5.

4.4.1 Receive data

To be able to create a hybrid integration system, using Azure, the components explained in section 4.2.1 are needed. The basic idea with a hybrid solution is to let

all traffic and data transfers pass through the cloud, where only one communication path is used between the On-Premise integration system and the cloud.

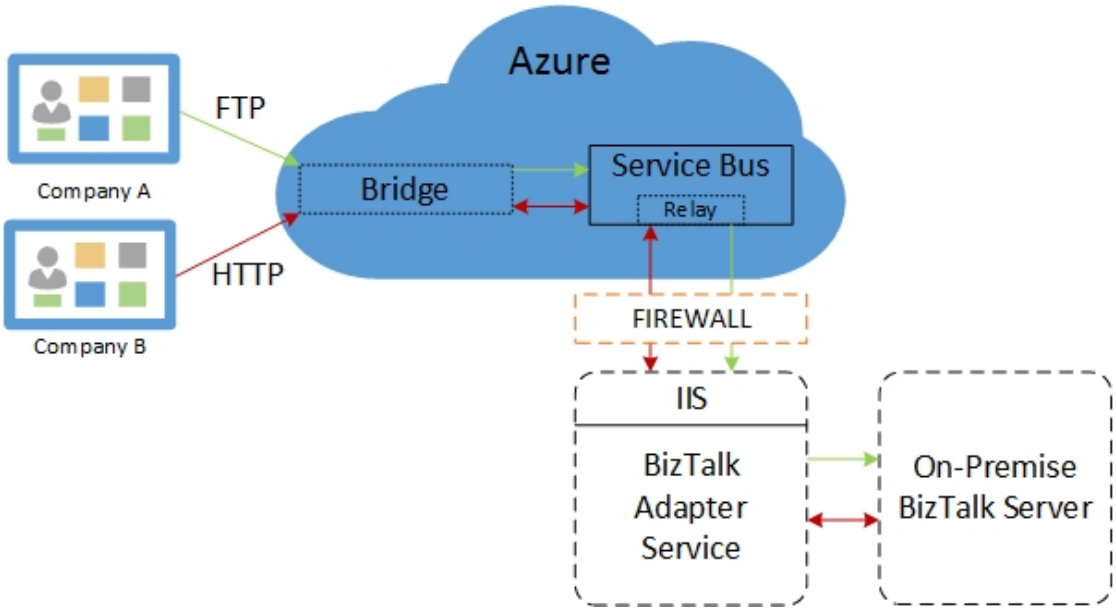

Figure 4.3: Hybrid Integration System: Receive data

In figure 4.3, an implementation overview of a hybrid integration system is provided, where the needed components for this scenario to work are shown. Clients from the outside connects to a *Bridge* provided in Azure, which process the data and forward it to the given destination, in this case the destination is the *Service Bus*. The On-Premise IIS handles the communication with the Azure Service Bus and transfers the data towards the On-Premise BizTalk Server, with assistance of the *Adapter Service*.

Two different transfer methods are given in figure 4.3, FTP and HTTP transfers. For Company A to send files via FTP, the files are dumped at a bridge via a certain FTP source[48]. The bridge process the data if needed or just pass it through towards the *Service Bus Relay*[48]. The On-Premise IIS is then responsible for transferring the data from the Relay and to the On-Premise BizTalk Server, which for example might store it in a database.

Regarding HTTP transfers, connections to some IIS located On-Premise is also achieved through a *Bridge* in Azure. In this case, a request-reply bridge is used, which forwards the request to the *Service Bus Relay* from where the IIS collects it and passes it through and into the On-Premise system[49]. If a response is needed, which often is the case when dealing with HTTP transfers, this response

traverse the same way backwards towards, in this specific case Company B, which is explained in the next section[50].

4.4.2 Send data

Sending data from an On-Premise integration system, through Azure and towards a specific destination is not especially complex. As described in section 4.2.2, a bridge act as a source, processing the data if needed and then routed towards some specific destination. In the sending scenario though, where data is supposed to be sent from the On-Premise integration system, a bridge is configured to receive data from the Service Bus. The On-Premise system is connected, as described shown in figure 4.3, to the Service Bus via the On-Premise IIS. The data that is supposed to be transmitted is sent via the IIS to the Service Bus Relay, which is configured to forward the data to a specific bridge. This bridge process the message if needed, and then forwards the message towards some specific destination outside of the Azure cloud.

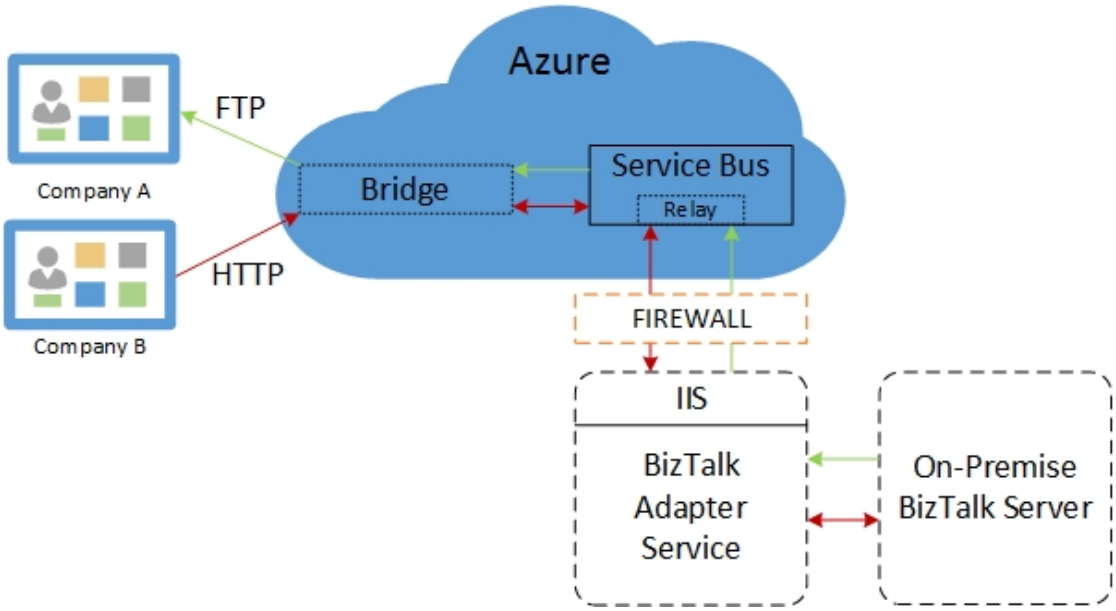

Figure 4.4: Hybrid Integration System: Send data

In figure 4.4, the process of sending is shown. This process is more or less the same as receiving, although the messages are sent in the other direction. As seen, the green arrows(representing FTP transmit) are directed in the opposite way.

4.5 Security overview of the migrated system

In this section, a security overview of the migrated hybrid solution is provided. As the proposed solution for how to take the first natural step towards cloud migration, presented in section 4.4, in practice only uses Azure for external connections and passing through data, security mechanisms should be implemented already before the data is transmitted towards the intended source. The main reason for using such a hybrid solution, is as mentioned before, to get rid of all external communication paths through the internal firewall, and replacing these with only one to the service bus in the cloud. This is a smart approach in my opinion, which should make the organizations feel more secure against intruders and different network attacks. As services are migrated to the cloud, and all external connections and corresponding traffic should traverse through it, two new types of connections are needed. These connections are the communication path for the external organizations to the cloud, and the connection between the On-Premise solution and the service bus in the cloud.

4.5.1 Connection to Azure

For the On-Premise integration system being able to communicate and receive data from other parties through Azure, a static communication path to the Service Bus Relay is needed. The bridge that is receiving data and possibly processing it, set the service bus as destination from where the On-Premise IIS fetching the data and passes it to the BizTalk Server. The IIS creates a Service Bus Relay Binding, which acts as a listener[52]. For the On-Premise IIS to be authenticated, being able to set up the listeners and the corresponding bindings, credentials and certificates is checked by the WAAD(Windows Azure Active Directory) in the cloud [53]. The connection between the On-Premise IIS and the Azure cloud is based on HTTP(S).

For external connections from companies that should send data to the Bridges, authentication is achieved through the same WAAD as above. Login credentials and certificates are checked, for giving the external connections the rights for using the services[54]. All external connections that need to be authenticated are managed in the cloud, while the internal authentications are still managed with the On-Premise AD. In this manner, the internal AD is never exposed in the cloud.

4.5.2 Confidentiality

The proposed hybrid migrated solution proposed in section 4.4 does not use any virtual machines or storage methods provided by Azure. The proposed solution only make use of the BizTalk Services for potential processing of data and passing

it through Azure. End-to-end connections between parties are established, through Azure, where mechanisms for providing confidentiality already are achieved when the sensitive information leaves the sending system. Which keys, how these are exchanged between parties and what encryption algorithms that should be used between parties is still a major challenge.

4.5.3 Integrity

As mentioned in section 4.5.2, end-to-end connections between the parties are established, which is sending messages through Azure in a hybrid manner. As for encryption, mechanisms for providing integrity, such as signing a message, should be performed before the message even leaves the system. While the message is inside of the Azure borders though, and as example being transferred between a bridge and the service bus, it is internally protected by SSL authentication, which is provided between all the internal components of Azure[55].

4.5.4 Availability

One of the main advantages provided by cloud platforms is robust availability based on extensive redundancy achieved with virtualization technology. Windows Azure provides numerous levels of redundancy to provide maximum availability of customers data. All customer data is replicated within Windows Azure to three separate nodes, which minimizes the impact of hardware failures. The customers can also take leverage from the geographically distribution of the Windows Azure datacenters, by choosing to replicate and synchronize data between them.[7]

The physical security, the Microsoft facilities where the Windows Azure servers run, is as mentioned above geographically distributed around the world. These facilities is designed to run 24/7 and employs various measures to help protect from power failures, physical intrusion and network outages. As for example, access to these facilities is limited to a very small number of operations personnel, who must regularly change their administrative access credentials, and each datacenter has a minimum of two sources for electrical power.[7]

4.5.5 Security summary of the migrated system

It is understandable that organizations are doubtful regarding the security when migrating their systems. What the organizations do not consider, is that their hole collection of systems definitely does not need to be migrated at once. Instead, they could use the approach given in this case study which is based on using the cloud services to handle the external connections and pass traffic through it.

The same end-to-end security mechanisms can and should still be used, just like before, where the only difference is that the organizations do not have any direct contact with each other. That is a great benefit when dealing with firewall configurations, as these configurations should aim for being as bulletproof as possible for incoming connections.

4.6 Attack protection

In this section, the network attacks from section 2.2.4 is revisited. How, with the help of the Azure resources, protection for these attacks is achieved, and what to think of is provided. This section ends with a smaller summary of the protection mechanisms.

4.6.1 Eavesdropping

The VMs Virtual Switch, which is a virtual switch that handles traffic to and from the VMs, prevents sniffer-based attacks against other VMs on the same physical host. There also exist switches higher up in the rack, used for restricting what IP and MAC addresses can be used by the VMs and therefore mitigate spoofing attacks on internal networks. It is worth to mention that these Virtual Switches do not differ from a regular hardware switch, except from the fact that it is virtual.

To be able to eavesdrop traffic inside of the Azure environment, an attacker would first need to compromise a VM tenant for getting administrator rights on that specific VM. When having these rights, the attacker then needs to find and use some vulnerability in the hypervisor that handles the VMs, to get into the physical machines for obtaining system account rights on that specific machine. At this point, the attacker would only be able to see inbound traffic to the compromised host, as per definition of the Windows Azure hypervisor and VM's environment[56].

4.6.2 Identity spoofing

To address the possible identity spoofing problem, VLANs are used to partition the internal network. At the VM Switch, the switch that handles traffic to and from the VMs, filters are always in place per default to block broadcast and multicast traffic, with the exception of what is needed to maintain diverse DHCP leases. Furthermore, internal communication is always encrypted and mutually authenticated over an HTTPS connection, which provides a secure transfer path for configuration and certificate information that cannot be intercepted.[56]

4.6.3 MITM

As stated in section 2.2.4, a MITM attack is a case where an attacker intercepts a legitimate communication between two parties. As for any type of systems and communication between them, an attacker is able to pretend being the intended host in both directions. But of course, there exists prevention mechanisms like using VPN-tunnels or transmitting data through SSL or SSH, which is standard and provided out of the box, where the cryptographic keys are exchanged in some secure way.

4.6.4 DoS Attack

Windows Azure is having a load balancing infrastructure within their pool of resources. This balances partially mitigates DoS attacks, coming both from internal and external networks. For external connections, Windows Azure VMs are only accessible through public Virtual IP Addresses(VIPs), and all VIP traffic is routed through Windows Azure's load-balancing infrastructure. Azure monitors and detects internally initiated DoS attacks and removes the offending VMs/accounts from the network.[56]

4.6.5 Application-Layer DDoS Attack

As for DoS attack, the application-layer attack is prevented with the Azure load balancing infrastructure to mitigate these kind of attacks. To be able to perform such an attack, first an external connection to the cloud is needed. This connection can only be accepted through the organizations WAAD, where only granted users are accepted. These granted users are not a potential threat to the organization, which imply that these kind of attacks can be more or less excluded as a threat. Nevertheless, if one of these granted users is starting to perform such attacks, the load balancer will prevent harm to the system and block them. When that is the case, the user responsible for the attack must be evaluated for further collaboration.

4.6.6 Port scanning

The only ports which are open and addressable on a Windows Azure Virtual Machine, are those explicitly defined by the user. This applies for both internal and external ports within the customers cloud space. Unauthorized traffic is blocked with switch packet filtering, as the Windows Firewall is enabled by default on every VM created.[56]

As all ports are blocked by default when creating and starting up a VM in Azure,

the customer is responsible for what ports that should be open and possible rules for this. There exist abilities for setting up a separate firewall, if the one provided by Windows per default is not good enough for some reason, giving the developers full control over what ports that should be opened or not.

4.6.7 Attack protection summary

Virtual switches among with VLANs is one of the main components for achieving security within Azure. For an adversary to be able to get control over the physical resource, these switches and VLANs are needed to be traversed in some way, to get in contact with the Hypervisor that control the VMs. That is a process that feels extremely hard, in my opinion. Denial of service attacks are handled with load balancers, where the source of the ”attack” in the end is blocked, until this account is checked by an administrator. That is a prevention mechanism that feels both good and smart in my opinion.

For network security in general, I believe that the developers just have to be smart. All sensitive information should of course be encrypted, or at least signed/hashed to make sure that no one can change it. If the information is chosen not to be encrypted though, then no other than the developers can be blamed if the sensitive information leaks out, since eavesdropping and packet sniffing always is doable if the communication channel is unencrypted.

With the mechanisms for security protection against these arbitrarily network attacks given above, the system is at least as secure as the traditional On-Premise solutions, if not even more secure, since all data exchange is encrypted and there exists only one way in and out from the On-Premise system.

Chapter 5

Discussion

This chapter contains a discussion around the work and research done in this Master's Thesis project. The discussion treats migration to cloud services in general, my findings of taking a first natural step towards a hybrid cloud solution and a more specific part that handles Azure and BizTalk.

5.1 Cloud services

When facing the problem that the current data and computational resources is not sufficient any more, such an organization have to make big decisions and most likely spend a huge amount of money for upgrading. Questions regarding how much data transmissions will increase during the next coming years is hard to estimate, which yield big problems when designing and deciding about how big the resources should be of the new system that needs to be invested in.

Nowadays, the organizations have the opportunity to expand and/or move the systems to the cloud. All cloud vendors have solutions for all problems, at least they make it look like that, but in reality the facts are of another art. Migrating to the cloud is a huge benefit when talking about scalability and resource possibilities, as this in practice does not have any limitations. If hundreds of different machines are needed to interact with each other, at all time or just during a transmission peak, then it's not a problem as long as the company pay for it. The scalability is a big advantage, but how about security?

When hosting a system On-Premise behind own firewalls and maybe other security measures, the organization most likely feels secure as every security aspect they might think about most likely is secured in what they think, the best way. When decided to migrate the system, partly or as a whole, security aspects are

having the main focus and is holding back the organizations. That is something that feels kind of natural, as the company in practice use a cloud vendors hardware to store and process their sensitive data in a environment, potentially shared with some of the rival organizations.

All cloud vendors out there are flashing with how secure their cloud services are. That's something that is partially true, as all vendors are trying to create security mechanisms to meet the requirements from the potential customers. As cloud computing mostly make use of VMs, these can be configured in any way the customer wants. This is a big advantage, since security might be implemented in the very same way as it should have been On-Premise, in-house of the company. Although, the connection to the cloud along with how the internal security within the cloud borders is handled, is a big security question mark.

Using CAs and encrypted communication paths to secure the authentication parts for connecting to the cloud services should do the trick. The company basically need to trust the cloud vendor that the data in the cloud is isolated from other, but if the company is insecure on this point, make use of cryptography functions and encrypt all the sensitive data, which for me feels very natural. When discussing a solution for a cloud environment with a cloud vendor, all those arising security questions should be asked and if no comfortable answer is able to be given, then the organization might consider to wait until these security aspects are fixed, if they ever will be.

5.2 Azure and BizTalk

When dealing with integration systems and the arising amount of traffic, cloud resources seem to be extremely appropriate. Since the amount of transmitted traffic between organizations differs from day to day, making use of the scalability of the resources provided by the cloud vendor seems perfect.

By only having one external connection path from the On-Premise system, as proposed in the case study, maintenance and configuration of the firewall becomes much easier. As all external connections are handled by the cloud services, attack protection is not on the organizations table any more since the cloud vendor should provide protection mechanisms for this, if not, another cloud vendor probably should be considered.

By taking such a first natural step provided in this project, the migration part of some organization systems does not need to feel as something extremely hard

or big thing to do. Using the Azure cloud and the corresponding services shown in the case study, seems both smart and good. This assumption is based on the fact that the organizations doesn't need to handle the external communications through their own firewall, which is a big help for both the developers and the owners of the system, in a security perspective of view.

Nevertheless, network security problems are still needed to be addressed in the same way as before. Some parts, like the DoS attack, is protected per default by using Azure though. When configuring some solution while migrating to the cloud, the same policies and rules within the organization will still be needed to be configured.

As seen in the case study in chapter 4, the same security can be achieved in the migrated hybrid solution using Azure. With the thoughts above in mind, taking this first step towards the cloud environment really feels like the right thing to do, as hosting and maintaining own systems On-Premise slowly will be suppressed in the future.

Chapter 6

Conclusions

The purpose of this Master's Thesis project was to evaluate whether a first step of migration towards a hybrid cloud solution, is as least as secure as the corresponding On-Premise system is today. This report addresses the main concepts of the cloud environment, challenges for migrating a system to the cloud and different security aspects that are needed to be taken into consideration. The report is providing the reader with information of how to be able to take a first step towards a hybrid migration to the cloud, using Windows Azure and the integration platform Microsoft BizTalk.

Before starting to migrate any systems, partly or as a whole, a lot of different aspects are needed to be taken into consideration. When decided to migrate, much time is needed for evaluating different aspects to prevent from rushing into something that not feels completely comfortable. One of these important aspects is different types of security, while another one is to achieve trust for the chosen cloud vendor, even though the companies probably never will get to know exactly what is happening within the borders of the cloud. Nevertheless, the economical benefits of using the cloud services elastic and scalable model, is attracting organizations when their current pool of systems is in need of an upgrade.

Within this project, the security in the cloud is shown to be as least as secure as the common On-Premise solutions, while connection authentication might be seen as problem. Or at least, I think that this is something that will prevent organizations from migrating. Nevertheless, if organizations follow the case study provided in this thesis, where all the needed components are given and description of how they work and interact with each other, companies can achieve a migrated hybrid solution that have the same level of security. If the fact that only one external connection from the On-Premise system to the cloud also is taken into account, then the system is even more secure than the traditional On-Premise solution.

6.1 Future work

Two main things should be taken into consideration in the future. The first is to create a better system with a higher trust for adding external connections and corresponding authentication mechanisms, while the other handles case studies and guidelines for using the cloud resources much more by migrating more subsystems to the cloud and making use of even more VMs.

If a framework for external connections and the corresponding authentication mechanisms is created, which in some way convince possible organizations that this is a good and secure solution for authentication to and within the cloud environment, the use of cloud resources will raise in a high rate.

The other area that I think needs more research, is how to take the next steps for migrating. As shown in this thesis, a first natural step is shown where aspects and potential challenges is shown. If a similar work is done, where this natural step is taken to the next level where more subsystems are migrated, then I think that this also is something that will raise the use of cloud resources in the future. Guidelines of how to make use of data storage and using VMs for processing data already in the cloud, are probably of high interest for potential future customers.

Bibliography

[1] Xiao, Zhifeng, and Yang Xiao. "Security and privacy in cloud computing." Communications Surveys & Tutorials, IEEE 15.2 (2013): 843-859.

[2] Subashini, Subashini, and V. Kavitha. "A survey on security issues in service delivery models of cloud computing." Journal of Network and Computer Applications 34.1 (2011): 1-11.

[3] Zissis, Dimitrios, and Dimitrios Lekkas. "Addressing cloud computing security issues." Future Generation Computer Systems 28.3 (2012): 583-592.

[4] Lombardi, Flavio, and Roberto Di Pietro. "Secure virtualization for cloud computing." Journal of Network and Computer Applications 34.4 (2011): 1113-1122.

[5] Popovic, Kresimir, and Zeljko Hocenski. "Cloud computing security issues and challenges." MIPRO, 2010 proceedings of the 33rd international convention. IEEE, 2010.

[6] Lori, M. "Data security in the world of cloud computing." Co-published by the IEEE Computer And reliability Societies (2009): 61-64.

[7] Kaufman, Charlie, and Ramanathan Venkatapathy. "Windows Azure™ Security Overview." go. microsoft. com (2010).

[8] Almorsy, Mohamed, John Grundy, and Ingo Müller. "An analysis of the cloud computing security problem." Proceedings of APSEC 2010 Cloud Workshop, Sydney, Australia, 30th Nov. 2010.

[9] Rosado, David G., et al. "Security analysis in the migration to cloud environments." Future Internet 4.2 (2012): 469-487.

[10] Khajeh-Hosseini, Ali, et al. "Decision support tools for cloud migration in the enterprise." Cloud Computing (CLOUD), 2011 IEEE International Conference on. IEEE, 2011.

[11] Sengupta, Shubhashis, Vikrant Kaulgud, and Vibhu Saujanya Sharma. "Cloud computing security–trends and research directions." Services (SERVICES), 2011 IEEE World Congress on. IEEE, 2011.

[12] Vaquero, Luis M., et al. "A break in the clouds: towards a cloud definition." ACM SIGCOMM Computer Communication Review 39.1 (2008): 50-55.

[13] Lenk, Alexander, et al. "What's inside the Cloud? An architectural map of the Cloud landscape." Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing. IEEE Computer Society, 2009.

[14] School of Medicine, University of Miami. (2014, Feb.) "Confidentiality, Integrity and Availability (CIA)". [Online] `http://it.med.miami.edu/x904.xml`

[15] William, Stallings, and William Stallings. Cryptography and Network Security, 4/E. Pearson Education India, 2006.

[16] Curphey, Mark, et al. "A guide to building secure web applications." The Open Web Application Security Project 1 (2002).

[17] Hill, Mark D. "What is scalability?." ACM SIGARCH Computer Architecture News 18.4 (1990): 18-21.

[18] Stoneburner, Gary, Alice Goguen, and Alexis Feringa. "Risk management guide for information technology systems." Nist special publication 800.30 (2002): 800-30.

[19] Whitman, Michael E. "Enemy at the gate: threats to information security." Communications of the ACM 46.8 (2003): 91-95.

[20] TechRepublic. (2014, Mar.) "Understanding risk, threat, and vulnerability". [Online]. `http://www.techrepublic.com/blog/it-security/understanding-risk-threat-and-vulnerability/#`

[21] Tech-FAQ. (2014, Mar.) "Responding to Network Attacks and Security Incidents". [Online]. `http://www.tech-faq.com/responding-to-network-attacks-and-security-incidents.html`

[22] OWASP. (2014, Mar.) "Network Eavesdropping". [Online]. `https://www.owasp.org/index.php/Network_Eavesdropping`

[23] Tech-FAQ. (2014, Mar.) "Network Attacks". [Online]. http://www.tech-faq.com/network-attacks.html

[24] Tanase, Matthew. "IP spoofing: an introduction." Security Focus 11 (2003).

[25] Microsoft. (2014, Apr.) "Common Types of Network Attacks". [Online]. http://technet.microsoft.com/en-us/library/cc959354.aspx

[26] Gadge, Jayant, and Anish Anand Patil. "Port scan detection." Networks, 2008. ICON 2008. 16th IEEE International Conference on. IEEE, 2008.

[27] techopedia. (2014, Apr.) "System Integration (SI)". [Online]. http://www.techopedia.com/definition/9614/system-integration-si

[28] Microsoft. (2014, Apr.) "Adapters in BizTalk Server". [Online]. http://msdn.microsoft.com/en-us/library/aa561360.aspx

[29] Microsoft. (2014, Apr.) "An XML Guru's Guide to BizTalk, Part 2". [Online]. http://msdn.microsoft.com/en-us/magazine/cc163695.aspx#S5

[30] Microsoft. (2014, Apr.) "BizTalk Adapter Architecture". [Online]. http://msdn.microsoft.com/en-us/library/aa546073(v=cs.70).aspx5

[31] Jonas Grundén, Integration Consultant at Acando Consulting AB. 2014. Personal communication.

[32] Microsoft. (2014, Apr.) "Encryption and Signing Certificates". [Online]. http://msdn.microsoft.com/en-us/library/aa559690.aspx

[33] Microsoft. (2014, Apr.) "Certificates that BizTalk Server Uses for Encrypted Messages". [Online]. http://msdn.microsoft.com/en-us/library/aa559843.aspx

[34] Microsoft. (2014, Apr.) "Certificates that BizTalk Server Uses for Signed Messages". [Online]. http://msdn.microsoft.com/en-us/LIbrary/aa547244.aspx

[35] Microsoft. (2014, Apr.) "Creating a Highly Available BizTalk Server Environment". [Online]. http://msdn.microsoft.com/en-us/library/aa560847.aspx

[36] Microsoft. (2014, Apr.) "Sample BizTalk Server High Availability Scenarios". [Online]. http://msdn.microsoft.com/en-us/library/aa578057.aspx

[37] Young, Charles, et al. Microsoft BizTalk Server 2010 Unleashed. Sams Publishing, 2011.

[38] Microsoft. (2014, Apr.) "Mitigating Denial of Service Attacks". [Online]. http://msdn.microsoft.com/en-us/library/aa561923.aspx

[39] Microsoft. (2014, May.) "Introducing Azure". [Online]. http://azure.microsoft.com/en-us/documentation/articles/fundamentals-introduction-to-azure/

[40] Microsoft. (2014, May.) "Azure Queues and Service Bus Queues - Compared and Contrasted". [Online]. http://msdn.microsoft.com/en-us/library/hh767287.aspx

[41] TechNet Blog. (2014, May.) "Microsoft is pushing Windows Azure BizTalk Services". [Online]. http://blogs.technet.com/b/wikininjas/archive/2014/03/23/microsoft-is-pushing-windows-azure-biztalk-services.aspx

[42] Microsoft Blog. (2014, May.) "Service Bus, Workflow, BizTalk Server, and Windows Azure BizTalk Services". [Online]. http://blogs.msdn.com/b/mdoctor/archive/2013/06/06/service-bus-workflow-biztalk-server-and-windows-azure-biztalk-services.aspx

[43] Microsoft. (2014, May.) "What are Bridges?". [Online]. http://msdn.microsoft.com/en-us/library/hh689768.aspx

[44] Kent Weare's Integration Blog. (2014, May.) "Introducing Windows Azure BizTalk Services Preview –Part 1". [Online]. http://kentweare.blogspot.se/2013/06/introducing-windows-azure-biztalk.html

[45] Microsoft. (2014, May.) "Azure Service Bus". [Online]. http://azure.microsoft.com/en-us/documentation/articles/fundamentals-service-bus-hybrid-solutions/

[46] Channel 9. (2014, May.) "Service Bus Introduction". [Online]. http://channel9.msdn.com/posts/Service-Bus-Introduction/

[47] Kent Weare's Integration Blog. (2014, May.) "Windows Azure BizTalk Services Preview (Part 2) –BizTalk Adapter Services SAP Integration". [Online]. http://kentweare.blogspot.se/2013/06/windows-azure-biztalk-services-preview.html

[48] Microsoft. (2014, May.) "Tutorial: Using BizTalk Bridges to Insert Flat File Messages into an On-premises SQL Server". [Online]. http://msdn.microsoft.com/library/azure/hh949811.aspx

[49] Microsoft. (2014, May.) "Tutorial: Using BizTalk Service Bridges to Send and Receive Messages from Service Bus Relay Service". [Online]. `http://msdn.microsoft.com/en-us/library/jj158971.aspx`

[50] Microsoft. (2014, May.) "Hosting WCF Services with Service Bus Endpoints on IIS". [Online]. `http://msdn.microsoft.com/en-us/library/hh966775.aspx`

[51] Jansen, Wayne A. "Cloud hooks: Security and privacy issues in cloud computing." System Sciences (HICSS), 2011 44th Hawaii International Conference on. IEEE, 2011.

[52] Microsoft. (2014, May.) "Service Bus Bindings". [Online]. `http://msdn.microsoft.com/en-us/library/hh410102.aspx`

[53] Microsoft. (2014, May.) "Azure Active Directory". [Online]. `http://msdn.microsoft.com/library/azure/jj673460.aspx`

[54] Microsoft. (2014, May.) "Azure Active Directory Authentication Protocols". [Online]. `http://msdn.microsoft.com/en-US/library/azure/dn151124.aspx`

[55] Microsoft. (2014, May.) "Manage Certificates". [Online]. `http://msdn.microsoft.com/en-us/library/azure/gg981929.aspx`

[56] Marshall, Andrew, et al. "Security best practices for developing windows azure applications." Microsoft Corp (2010).