Dokumentation Modul 143

von Yassin Sobhy

Inhalt

[Netzwerkschema 0](#_Toc156287889)

[1. Einführung 0](#_Toc156287890)

[1.1 Zielgruppe 0](#_Toc156287891)

[1.2 Ziele einer Sicherheitsumgebung 0](#_Toc156287892)

[1.3 Begriffserklärungen 0](#_Toc156287893)

[2. Allgemeine Anforderungen 0](#_Toc156287894)

[2.1 Netzwerke 0](#_Toc156287895)

[2.2 Computerhardware 0](#_Toc156287896)

[3. Systemdefinitionen 1](#_Toc156287897)

[3.1 Betriebssysteme 1](#_Toc156287898)

[3.2 Partitionierungen 1](#_Toc156287899)

[3.3 Einsatzzweck 1](#_Toc156287900)

[3.4 Speichersysteme 1](#_Toc156287901)

[3.5 Spezialsoftware 1](#_Toc156287902)

[4. Systemverwaltung 1](#_Toc156287903)

[4.1 Der Zugang zum System 1](#_Toc156287904)

[4.2 Protokollierung 1](#_Toc156287905)

[5. Betriebssicherheit 1](#_Toc156287906)

[5.1 Technische Sicherheitsmassnahmen 1](#_Toc156287907)

[5.2 Benutzerauthentifizierung 1](#_Toc156287908)

[6. Datensicherung 2](#_Toc156287909)

[6.1 Daten, die gesichert werden 2](#_Toc156287910)

[Erstellung und Konfigurierung der VMs 2](#_Toc156287911)

[Funktion der Infrastruktur 3](#_Toc156287912)

# Ein Bild, das Text, Diagramm, Screenshot, Rechteck enthält. Automatisch generierte BeschreibungNetzwerkschema

# 1. Einführung

In diesem Dokument wird festgehalten, wie ich die Infrastruktur aufgebaut habe.

## Zielgruppe

Kunde und IT Verwaltung.

## 1.2 Ziele einer Sicherheitsumgebung

Die Umgebung muss Backups haben, auf aktuellem Stande sein und einen guten Antivirus haben.

## 1.3 Begriffserklärungen

OS = Operating System = Betriebssystem

Partition = Eine unterteilung der Festplatte

NTFS = Dateisystem für die Festplatte

SCP = Secure Copy

SSH = Secure Shell, mit dem kann man sich auf ein Gerät verbinden.

RDP = Remote Desktop, ermöglicht Verbundungen zu anderen Geräten im gleichen Netzwerk.

SQL = Eine Datenbank Art.

# 2. Allgemeine Anforderungen

## 2.1 Netzwerke

192.168.128.0 – 192.168.128.255/24

172.31.64.0 – 172.31.95.255/20

172.31.0.0 – 172.31.15.255/20

## 2.2 Computerhardware

Es gibt keine physische Hardware in dem Sinne, nur die Hardware der Virtuellen Server.

# 3. Systemdefinitionen

## 3.1 Betriebssysteme

Der Backupserver ist mit Windows Server 2022 Standard ausgestattet. Die beiden Linux Server mit Ubuntu Server 22.4.

## 3.2 Partitionierungen

Es gibt je 2 Partition pro Server, die System Partition auf dem das OS\* läuft und dann die Partition\* mit den Daten.

## 3.3 Einsatzzweck

Die Umgebung wird genutzt, um Daten in eine Datenbank abspeichern zu können und eine Webseite hosten zu können. Dazu wird dies noch gesichert im Falle eines Datenverlusts.

## 3.4 Speichersysteme

Der Backupserver ist dafür da, um Backups der beiden Linux Server zu erstellen. Die Backups werden auf der NTFS\* formatierten Festplatte gespeichert.

## 3.5 Spezialsoftware

Der Backupserver benützt SCP\* um die Daten von den Zielservern zu kopieren.

# 4. Systemverwaltung

## 4.1 Der Zugang zum System

Der Zugriff erfolgt über SSH\*. Der Backupserver ist auch über RDP\* möglich.

## 4.2 Protokollierung

Sowie Windows wie auch Ubuntu haben Logs in denen Berichte zu abgeschlossenen Aktionen geschrieben werden.

# 5. Betriebssicherheit

## 5.1 Technische Sicherheitsmassnahmen

Linux ist weit mehr beschränkt was die Angriffsfläche reduziert und Windows hat den integrierten Windows Antivirus. Falls ein weiterer Antivirus gewünscht ist, kann der eingerichtet werden.

## 5.2 Benutzerauthentifizierung

Die Benützer verbinden sich mit einem privaten SSH Key. Zuvor muss allerdings der öffentliche Schlüssel in den Linux Systemen registriert werden.

# 6. Datensicherung

## 6.1 Daten, die gesichert werden

Die Webseiten auf dem Webserver werden zusammen mit den SQL\* Datenbanken gesichert. Das Betriebssystem wird nicht gesichert.

# Erstellung und Konfigurierung der VMs

Als erstes habe ich die den Windows Backup Server auf meinem lokalen Laptop installiert. Ich habe ihm 60 GB Speicher gegeben da dies für die kleine Menge, die wir backuppen am Anfang völlig ausreichen wird. Falls der Speicher knapp wird, kann man ihn nach belieben vergrössern.

Danach folgten die beiden Linux Server in der AWS Umgebung. Hier hatte ich einbisschen mehr mühe, obwohl ich schon mit AWS gearbeitet habe. Immer wieder gab es fehler und etwas funktionierte nicht wie ich es wollte. Insgesamt musste ich ca. 10 neue Server aufsetzen. Der eine Server war ein Web Server und der andere eine SQL Datenbank. Beide Server sind mit SSH zugänglich, haben aber eine SSH Key Authentifizierung, mit welcher man sich nur Verbinden kann, wenn man den richtigen Private Key hat. Auf allen drei Servern habe ich SSH Scripte erstellt mit denen ich einfach, ohne jedes Mal den Befehl mit dem passenden Schlüssel einzugeben, mich mit dem gewünschten Server verbinden kann. Jeder Server hat Zugriff auf jeden. Ausserdem ist der lokale Backup Server per RDP erreichbar.

Ganz wichtig war auch die richtige konfigurierung der Sicherheitsgruppen. Davon habe ich zwei neue erstellt. Die eine für den Web Server habe ich «WEB» genannt, und die für den SQL natürlich «SQL». Beide Sicherheitsgruppen haben die Ports 22 und 80 offen. 22 für SSH und 80 für http. Die SQL Sicherheitsgruppe hat zusätzlich noch den Port 1433 offen für MSSQL. Das sind die eingehenden Regeln. Ausgehend ist alles erlaubt.

Alle Server haben eine fixe IP bekommen, vor alle bei den AWS Servern war mir das wichtig da, wenn ich einen neuen Server aufsetzen musste, die ganze Zeit eine neue IP zugewiesen wurde und alle Scripte, welche auf den IP Adressen basieren, nicht mehr funktioniert hätten und wieder mühsam angepasst hätten werden müssen.

Der Web Server hat als Anwendung Apache2 installiert, damit auch eine Webseite zustande kam, auf welche man zugreifen kann. Dazu kommt auch noch PHP. Der SQL Server hat nur MSSQL installiert. Beide haben kein Passwort für den Root konfiguriert, was für die Aufsetzung von Vorteil war und mir auch einige Nerven gespart hat. Wenn es gewünscht ist, kann natürlich ein Passwort festgelegt werden.

# Funktion der Infrastruktur

Die ganze Infrastruktur funktoniert so, dass die beiden Linux Server sebstständig ein Backup erstellen. Beim SQL Server wird logischerweise die SQL Datenbank gesichert und in ein Dump File gespeichert, anders wie beim Web Server. Dort werden die ganzen Web Dateien gesichert und in eine komprimierte Datei abgespeichert. Dazu wird direkt, wenn das Backup automatisch mit einem Cronjob ausgeführt wird, ein anderes Script ausgeführt welches automatisch das ältere Backup vom Server löscht. So wird sichergestellt das immer nur das neuste Backup auf den Linux Servern liegen und nicht zu viel Speicher verbraucht wird. 10 Minuten später führt der Windows Backup Server seinerseits ein Script aus, welches per SSh auf den jeweiligen Server geht und die Backup Datei mit SCP (Secure Copy) hinüberkopiert. Danach wird die Datei in einem zugewiesenen Ordner abgelegt. Dieser Vorgang ist mit der Windows Aufgabenplanung automatisiert. Falls die Sicherung auf seiten der Linux Server länger dauern sollte, durch grössere Backups zum Beispiel, kann auch die Zeit, bis das Kopieren in Kraft tritt, individuell angepasst werden.