
BACHELORARBEIT

Herr
Richard Schubert

**Prototyping zur Umstellung von
Telnet-Servern auf eine
Kommunikationsschnittstelle nach
Stand der Technik**

Mittweida, 2019

Fakultät Angewandte Computer- und
Biowissenschaften

BACHELORARBEIT

Prototyping zur Umstellung von Telnet-Servern auf eine Kommunikationsschnittstelle nach Stand der Technik

Autor:
Herr Richard Schubert

Studiengang:
Angewandte Informatik

Seminargruppe:
IF16Ws-B

Erstprüfer:
Prof. Dr.-Ing. Thomas Beierlein

Zweitprüfer:
Dipl.-Inf. Jan Friedrich

Einreichung:
Mittweida, 07. Oktober 2019

Verteidigung/Bewertung:
Mittweida, 2019

Bibliographische Beschreibung:

Schubert, Richard: Prototyping zur Umstellung von Telnet Servern auf eine Kommunikationsschnittstelle nach Stand der Technik. - 2019. - 3, 22 S., Mittweida, Hochschule Mittweida, Fakultät Angewandte Computer- und Biowissenschaften, Bachelorarbeit, 2019

Englischer Titel

Prototyping to cutover from Telnet Servers to a communication interface as state of art.

Kurzbeschreibung:

Die folgende Arbeit handelt von der Umstellung der Kommunikationssoftware eines Servers in der Logistikabteilung der KOMSA AG. Ziel ist es den Ist-Zustand zu analysieren und darauf basierend einen Prototyp zu erstellen oder einen Gerätevorschlag zu unterbreiten, welche die Fortführung der Prozesse im Logistikbereich ermöglichen. Um einen Prototyp zu entwickeln, ist eine Analyse der IT-Infrastruktur der betreffenden Logistikabteilung durchgeführt und darauf aufbauend Konzepte zum Ablösen der alten Telnet-Server entwickelt worden. Der Telnet-Server des Anbieters Pragmasys konnte den Anforderungen am besten gerecht werden, daher wurde der Prototyp mit diesem entworfen. Der KOMSA AG wird empfohlen vorerst den Prototypen zu nutzen und eine Ablösung der MDE-Geräte in Erwägung zu ziehen, da das Telnet-Protokoll zu unsicher ist.

Inhaltsverzeichnis

Inhaltsverzeichnis	I
Abbildungsverzeichnis.....	III
Abkürzungsverzeichnis	IV
1 Einleitung	1
1.1 <i>thematische Hinführung</i>	<i>1</i>
1.2 <i>Zielsetzung.....</i>	<i>1</i>
1.3 <i>Methodik.....</i>	<i>2</i>
2 Grundlagen und Definitionen	3
2.1 <i>Kommunikationsschnittstellen.....</i>	<i>3</i>
2.1.1 <i>Datenübertragung</i>	<i>3</i>
2.1.2 <i>Kommunikationsprotokolle.....</i>	<i>3</i>
2.2 <i>Einordnung im OSI-Modell.....</i>	<i>4</i>
2.3 <i>Werkzeuge und Umgebungen</i>	<i>4</i>
2.3.1 <i>MDE.....</i>	<i>5</i>
2.3.2 <i>Telnet.....</i>	<i>5</i>
2.3.3 <i>SSH.....</i>	<i>5</i>
2.3.4 <i>Open SSH.....</i>	<i>6</i>
2.3.5 <i>PuTTY</i>	<i>6</i>
3 Analyse der Kommunikationsschnittstelle am Fallbeispiel.....	8
3.1 <i>Firmenportrait.....</i>	<i>8</i>
3.2 <i>Erfassung des Ist-Zustandes.....</i>	<i>8</i>
3.3 <i>Zusammenfassung der Analyse</i>	<i>10</i>
4 Auswertung der Kommunikationsschnittstelle	11

5	Konzept	12
5.1	<i>SSH in MDE-Geräte implementieren.....</i>	<i>12</i>
5.1.1	OpenSSH auf Windows Server 2019 installieren	12
5.1.2	SSH auf MDE-Geräten ermöglichen.....	12
5.2	<i>Telnet Server von Drittanbietern.....</i>	<i>13</i>
5.2.1	Einrichtung der Server-Anwendungen.....	13
5.2.2	Nutzung des Telnet-Clients auf MDE-Geräten	14
5.3	<i>Austausch der MDE-Hardware.....</i>	<i>14</i>
5.3.1	Anforderungen an MDE-Geräte.....	14
5.3.2	Wechsel der MDE-Geräte	15
6	Methoden.....	16
6.1	<i>Einrichtung der Testumgebung</i>	<i>16</i>
6.2	<i>Test für die Nutzung der MDE-Geräte mit Telnet.....</i>	<i>16</i>
6.2.1	Kypm Telnet Server	16
6.2.2	Pragmasys Telnet Server	18
6.2.3	Weitere Telnet-Anbieter.....	19
6.3	<i>Aufsetzen des Prototyps mit Pragmasys.....</i>	<i>20</i>
7	Fazit und Ausblick	21
7.1	<i>Auswertung der Ergebnisse.....</i>	<i>21</i>
7.2	<i>Einschätzung der Lage</i>	<i>21</i>
7.3	<i>Empfehlung an die KOMSA AG.....</i>	<i>22</i>
	Literaturverzeichnis	V
	Selbstständigkeitserklärung.....	VI

Abbildungsverzeichnis

Abbildung 1: Darstellung des OSI-Modells	4
Abbildung 2: PuTTY-Client	7
Abbildung 3: MDE-Gerät BHT 805BW [URL-7].....	9
Abbildung 4 Kypm Telnet Server.....	17
Abbildung 5 Pragmasys Server Configuration	19
Abbildung 6 Quellcode von einem Skript zum Aktualisieren der Serverdaten.....	20

Abkürzungsverzeichnis

IP	–	Internet Protocol
IT	–	Informationstechnik
MDE	–	Mobile Datenerfassung
KTS	–	Kypm Telnet Server
LAN	–	Local Area Network
OS	–	Sperating System
SAC	–	Standard Ambiguous Code
SSH	–	Secure Shell
TCP	–	Transmission Control Protocol
UDP	–	User Datagram Protocol
VM	–	Virtuelle Maschine
WLAN	–	Wireless Local Area Network

1 Einleitung

Im Folgenden wird die thematische Hinführung erläutert. Weiterhin wird die Zielsetzung definiert, sowie die Methodik dieser Arbeit dargestellt.

1.1 thematische Hinführung

Die Kommunikationsschnittstelle dient zum Austausch von Daten zwischen zwei oder mehreren Parteien, welches nach festgelegten Regeln erfolgt. In Logistikbereichen können diese Schnittstellen genutzt werden, um die Kommunikation zwischen Scanner und Server zu realisieren. In der Vergangenheit wurde für die Kommunikation zwischen den Schnittstellen das Telnet-Protokoll genutzt. Hierfür hat Microsoft einen Telnet-Server bereitgestellt. Diese Server werden von einer Logistikabteilung der KOMSA AG genutzt. Die KOMSA AG zählt zu den 20 umsatzstärksten Unternehmen in Sachsen und beschäftigt über 1400 Mitarbeiter. Das Unternehmen ist in den Bereichen Großhandel für IT- und Kommunikationsprodukte, Reparaturdienst für Mobiltelefone und Dienstleistungen tätig [vgl. URL-1]. Die Logistikabteilung der KOMSA AG teilt sich in zwei unabhängige Logistikbereiche „Kkis“ und „MWCS“. In der folgenden Arbeit wird auf den Bereich „Kkis“ eingegangen.

1.2 Zielsetzung

Es soll mit Hilfe des Beispielunternehmens ein Serverprototyp entwickelt oder ein Gerätevorschlag erarbeitet werden, welcher den aktuellen Anforderungen, insbesondere der Kompatibilität zur IT-Infrastruktur des Logistikbereichs „Kkis“, erfüllt. Hierzu gehört eine Analyse des vorhandenen Serverkonstrukts für die mobilen Datenerfassungsgeräte (MDE-Gerät) und einen Vergleich mit den in der Literatur beschriebenen Konzepten. Abschließend wird aus den gewonnenen Resultaten eine Handlungsempfehlung abgegeben.

1.3 Methodik

Im ersten Kapitel werden die thematische Hinführung zu Kommunikationsschnittstellen ebenso wie die Zielsetzung für diese Arbeit formuliert. Die Methodik beschreibt den Aufbau und die Vorgehensweise.

Das Kapitel zwei bildet den theoretischen Teil dieser Ausarbeitung und beruht auf Angaben aus der Fachliteratur. Um grundlegende sowie für die Arbeit relevante Begriffe zu erörtern, befasst sich das zweite Kapitel mit Grundlagen und Definitionen.

Das Beispielunternehmen wird in Kapitel drei anhand eines kurzen Firmenportraits vorgestellt. Ebenso geht das Kapitel auf die Prozesslandschaft des Fallbeispiels ein, um das Prozessumfeld zu beschreiben. Den Kernpunkt dieses Kapitels bildet die Analyse der ansässigen IT-Infrastruktur, der Logistikabteilung „Kkis“. Um einen Überblick über die gesammelten Erkenntnisse zu gewährleisten, wird das Kapitel drei mit einer Zusammenfassung abgeschlossen. Die Analyse der Infrastruktur beinhaltet keine Wertung.

Im vierten Kapitel wird die Analyse aus Kapitel drei ausgewertet und Anforderungen an einen Server-Prototyp formuliert.

Das Kapitel fünf befasst sich mit der Entwicklung eines Konzepts basierend auf der Auswertung aus Kapitel vier. Hierbei werden die gewonnenen Erkenntnisse aus der Literatur herangezogen. Mithilfe des Kapitels fünf sollen Handlungsempfehlungen abgegeben werden.

Die Methoden zur Umsetzung der erarbeiteten Konzepte werden im Kapitel 6 beschrieben. Es wird dargestellt, wie ein ausgewähltes Konzept als Prototyp entworfen wird.

Empfehlungen für eine Kommunikationsschnittstelle mit der dazugehörigen Hardware werden im Kapitel sieben dargelegt.

2 Grundlagen und Definitionen

Das folgende Kapitel befasst sich mit Grundlagen der Kommunikationsschnittstellen und mit Definitionen für die Arbeit relevanter Begriffe.

2.1 Kommunikationsschnittstellen

Über die Kommunikationsschnittstelle werden Daten zwischen Programmen nach festgelegten Protokollen ausgetauscht. Die Schnittstellen sind die Berührungspunkte der Teilsysteme. So besitzt ein Gesamtsystem zwischen den Teilsystemen Kontaktpunkte, um es als Ganzes betrachten zu können.

2.1.1 Datenübertragung

Datenübertragung wird im Allgemeinen genutzt, um Informationen von einem Sender zu einem Empfänger zu übermitteln. In Bezug auf kabellose Datenübertragung wird in dieser Arbeit hauptsächlich auf für das MDE-Gerät möglicher Technik eingegangen. Das Wireless Local Area Network (WLAN) bezeichnet ein lokales Funknetz, welches auf den Eigenschaften des LAN basiert. Die Eigenschaften sind in den 802.X Standards beschrieben, wobei das WLAN den 802.11 Standards unterliegt. WLAN dient „hauptsächlich dazu, auf Schicht drei des ISO Modells IP-Pakete zu transportieren.“ [Sauter, 2018 S.337] Im Vergleich zu anderen drahtlosen Übertragungsmöglichkeiten, wie etwa Bluetooth, bietet WLAN eine gute Reichweite bei gleichzeitig hoher Datenübertragungsrage. [vgl. Sauter, 2011]

2.1.2 Kommunikationsprotokolle

Kommunikationsprotokolle sind Richtlinien nach denen zwei oder mehrere Prozesse Protokoll-Dateneinheiten über eine Schicht (siehe Abb. 1) miteinander austauschen können. Dabei ist es theoretisch egal auf welche Weise die Daten kommuniziert werden oder in welcher Zeitspanne sie gesendet werden. Auf der fünften Ebene des OSI-Modells (siehe Abb. 1) ist ein weit verbreitetes Protokoll das Transmission Control Protocol

(TCP). Es bietet eine stabile Datenübertragung, wobei eine Partei die Rolle des Servers und die andere als Client annimmt. Beide Seiten können dabei auch gleichzeitig Daten senden und der Server kann problemlos mit anderen Clients kommunizieren. Sobald eine Verbindung hergestellt werden soll, wird zunächst der „Three-way-Handshake“ durchgeführt. Dieser ermöglicht das zuverlässige Senden eines Datenpaketes. Sobald die Verbindung geschlossen ist, bleiben die Sockets noch 30 Sekunden offen bis sie wieder für eine neue Verbindung bereitstehen.

2.2 Einordnung im OSI-Modell

Bei dem „Open System Interconnection Model“ handelt es sich um ein Architekturmodell, das eine Netzwerkarchitektur in sieben verschiedene Schichten zerlegt und dabei jede einzelne feste Aufgabe auf diese Ebenen zuteilt. Weiterhin werden die Schnittstellen zu darauf aufbauenden Schichten klar definiert, sodass sich auf derselben Schicht befindliche Netzwerkprotokolle problemlos untereinander austauschen lassen und beliebig kombinierbar sind. [vgl. Microsoft Corporation, 2019]

7	Anwendungsschicht
6	Darstellungsschicht
5	Kommunikationsschicht
4	Transportschicht
3	Vermittlungsschicht
2	Sicherungsschicht
1	Physikalische Schicht

Abbildung 1: Darstellung des OSI-Modells

2.3 Werkzeuge und Umgebungen

Im folgenden Abschnitt werden die für diese wissenschaftliche Untersuchung benutzte Soft- und Hardware erläutert und anschließend die Gegebenheiten, sowie die Anforderungen analysiert.

2.3.1 MDE

MDE dient der lokalen Erfassung und zur Verfügungstellung von Daten, welche in Folgeprozessen genutzt werden. Die Datenerfassung kann online oder offline erfolgen. Bei der offline Variante werden die Daten im Terminal gespeichert und über eine Dockingstation auf das System übertragen. Im Gegensatz dazu übermittelt die online Variante die Daten mittels Datenfunk [vgl. Heiserich et al., 2011]. Die Daten werden aus vorhandenen Barcodes ausgelesen und werden auf MDE-Geräte erkannt und angezeigt. Der Vorteil solcher Methoden zur Erfassung von Waren, Paketen oder sonstigen Gegenständen ist die Vermeidung von Übertragungsfehlern und eine stets aktuelle Datenbank. Außerdem ermöglicht diese Technik eine tiefgehende Anwendungslogik für solche Logistikoperationen.

2.3.2 Telnet

Telnet ist sowohl ein Programm als auch ein Protokoll. Es wurde entworfen um die physischen Terminals, welche mit jedem Computer verbunden sind, zu ersetzen. Es erlaubt Benutzern von verschiedenen Computern Skripte auf entfernten Computern auszuführen. Eines der Sicherheitsrisiken von Telnet ist, dass alle Nachrichten in Klartext übertragen werden. Demzufolge könnten Programme, alle Informationen anzeigen, die über das Netzwerk übertragen werden, sobald sie sich im Netzwerk befinden. Der Telnet Server kann sowohl als Dateibrowser als auch als Terminalserver genutzt werden. Der Standardport, der für die Verbindung genutzt wird, ist der Port 23. Das TCP-Protokoll wird meistens als Kommunikationsprotokoll genutzt. [vgl. Comer, D. E., 2003]

2.3.3 SSH

Das Secure Shell Protokoll ist ein Netzwerkprotokoll, welches auf der fünften bis siebten Schicht nach dem OSI-Modell arbeitet. Im Gegensatz zu Telnet die Sitzungsdaten und Anmeldedaten nicht im Klartext verschickt. Bei einem Verbindungsaufbau werden die herausgehenden Daten verschlüsselt und auf dem Rechner des Empfängers wieder entschlüsselt. Der Port 22 wird üblicherweise hierfür genutzt [vgl. Riedel, 2004].

2.3.4 Open SSH

OpenSSH ist ein Programmpaket mit SSH-Werkzeugen zur Datenübertragung, welches Clients, Dienstprogramme und einen Server beinhaltet. Für die Übertragung zwischen Client und Server nutzt es Secure Shell (SSH) inklusive dem „SSH File Transfer Protocol“. Der Open-SSH Server „sshd“ ist der SSH Daemon des Softwarepakets [vgl. URL-3].

2.3.5 PuTTY

PuTTY bezeichnet eine Anwendung zur Verbindung zweier Systeme. Es steht kostenlos zur Verfügung und wird zur Terminalemulation genutzt. Es kann für verschiedene Verbindungsprotokolle genutzt werden, wie beispielsweise Telnet oder SSH. Es kann auf Windows, Linux und MacOS installiert werden. Es können diverse Konfigurationen vorgenommen und unter einem Profil gespeichert werden. Die Abbildung 4 zeigt eine vorgenommene Einstellung, welche eine IP-Adresse und einen Port beinhaltet, für eine Verbindung vom Programm zu einem Server. Ebenfalls zu beachten sind die Reiter Logging, welcher zur Auswertung von Testversuchen den Logordner konfiguriert, Translation, bei dem das remote character set auf das des Remote-Servers angepasst wird, und Telnet, bei dem die Terminalfunktionen bearbeitet werden. Ein remote character set ist der Zeichensatz, der für die Kommunikation verwendet wird.

Für die folgende Untersuchung dient der PuTTY-Client als MDE-Gerät-Emulator, womit grundlegende Funktionen der Kommunikation zum Server getestet werden können.

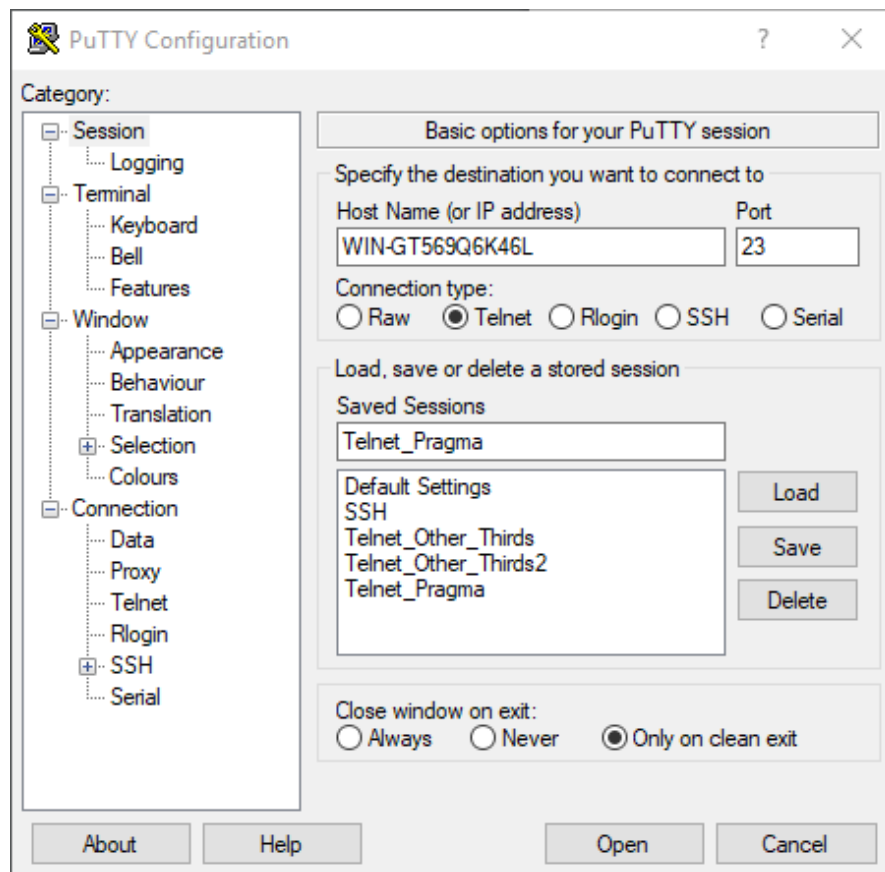


Abbildung 2: PuTTY-Client

3 Analyse der Kommunikationsschnittstelle am Fallbeispiel

Im folgenden Kapitel werden Informationen zu dem Beispielunternehmen erläutert. Die Infrastruktur der Logistikserverstruktur wird erfasst und ausgewertet.

3.1 Firmenportrait

Die KOMSA AG besteht seit 27 Jahren und hat mehr als 20000 Handelspartner in Deutschland und anderen europäischen Ländern. Das Sortiment der KOMSA AG umfasst über 25000 Produkte. Außerdem werden Dienstleistungen, wie zum Beispiel Reparatur von Mobilfunkgeräten, Schulung und Vermarktung angeboten [vgl. URL-1]. Die „KOMSA-Logistik versendet im Jahr rund 2,3 Millionen Pakete an Fachhändler, Systemhäuser, Onlinehändler, Flächenmärkte und Endverbraucher“[URL-1].

3.2 Erfassung des Ist-Zustandes

Die Logistikabteilung „Kkis“ arbeitet mit MDE-Geräten von Denso, den BHT 805BW. Es werden bis zu 10 solcher Geräte gleichzeitig für die Paketverwaltung im Bereich „Kkis“ genutzt. Durch Scannen des Barcodes registrieren die Geräte Pakete im System in einer Datenbank. Dies wird durch ein WLAN-Netzwerk über einen Server ermöglicht. Auf dem Server für die Logistikabteilung „Kkis“ ist das Betriebssystem Windows Server 2012 installiert. Es ist geplant die Server der KOMSA AG auf das Betriebssystem der Windows Server 2019 zu aktualisieren. Auf dem Kkis-Logistikserver ist das Feature „Telnet Server“ aktiviert, welches die Logistikoperationen der MDE-Geräte koordiniert. Die Logik für das Abspeichern von erfassten Barcodes auf einer Datenbank befindet sich in der Kkis-Logistikanwendung. Durch das Einloggen auf dem Telnet-Server als ein Nutzer der Gruppe „TelnetClients“ wird automatisch die Kkis-Logistikanwendung für denjenigen gestartet. Einmal pro Tag wird mittels einem PowerShell Skript der Telnet-Server gestoppt und aktualisiert. Danach erfolgt die Synchronisierung der Benutzerdaten und der Neustart des Telnet-Server.

Als MDE-Gerät wird das Gerät Denso BHT 805 (siehe Abb. 3) mit dem Betriebssystem BHT-OS verwendet. Auf diesem läuft die speziell für dieses System entwickelte Anwendung BHT-Term, welches das einzige geläufige Programm für das BHT-OS ist. Um Konfigurationen am Gerät vorzunehmen, muss die Anwendung mit den veränderten Einstellungen auf die Hardware übertragen werden. Änderungen können nicht innerhalb der bereits installierten Anwendung vorgenommen werden. Realisiert wird dies unter Anwendung des mitgeführten Windows Programm „BHT Term Config Tool“. Damit kann das Programm „BHT Term“ auf ein MDE-Gerät gespielt werden, welches in der Denso eigenen Programmiersprache BHT Basic 4.0 entwickelt wurde. Im BHT Term Config Tool können auf einer Benutzeroberfläche alle relevanten Konfigurationen vorgenommen werden [vgl. URL-6]. Die Verbindungseinstellungen verfügen zwar Anpassungsmöglichkeiten des Telnet-Protokolls, jedoch keine Möglichkeit das SSH Protokoll auszuwählen.



Abbildung 3: MDE-Gerät BHT 805BW [URL-7]

3.3 Zusammenfassung der Analyse

Auf dem Server des Logistikbereiches „Kkis“ der KOMSA AG ist das Betriebssystem „Windows Server 2012“ installiert. Auf diesem wird das Windows-Feature „Telnet-Server“ genutzt, um mit den MDE-Geräten über das Telnet-Protokoll zu kommunizieren. Als MDE-Gerät wird das „BHT 805BW“ verwendet, welches das Betriebssystem BHT-OS nutzt. Auf diesem läuft das Programm BHT-Term, welches ausschließlich das Telnet-Protokoll zur Netzwerkkommunikation nutzen kann. Die KOMSA AG plant deren Windows-Server auf die Betriebssystemversion „Windows Server 2019“ zu aktualisieren.

4 Auswertung der Kommunikationsschnittstelle

Auf Grund der geplanten Modernisierung der Windows-Server steht dem Beispielunternehmen die weitere Nutzung des Features „Telnet Server“ nicht mehr zur Verfügung, da dieses ab der Version „Windows Server 2016“ nicht mehr vorhanden ist [vgl. URL-2]. Dieses ist jedoch essenziell in der Prozesskette der Logistikabteilung, da die MDE-Geräte BHT 805BW nur über das Telnet-Protokoll kommunizieren können. Es ist ein Sicherheitsrisiko durch die Benutzung des Telnet-Protokolls vorhanden, aber durch das lokale Netzwerk wird grundlegender Schutz geboten.

Anhand dieser Auswertung ergeben sich folgende Anforderungen an einen Prototyp: Die Serveranwendung muss auf dem Betriebssystem Windows Server 2019 installiert und auf selbigem in Betrieb genommen werden können. Es ist notwendig, dass eine Verbindung zwischen dem BHT-Term des MDE-Geräts und der Schnittstelle des Servers zustande kommt. Hierbei ist irrelevant, ob dies über Telnet, SSH oder einen anderen Weg über WLAN geschieht. Die Sicherheit der Daten sollte zwar beachtet werden, aber nicht im Vordergrund stehen. Die Funktionen des Einlesens eines Barcodes sowie das Anmelden an dem Remote-Server und die daraus folgende Benutzung der Kkis-Logistikanwendung sollen weiterhin zur Verfügung stehen.

Unter den gegebenen Voraussetzungen werden potenzielle Ausweichmöglichkeiten für die eingesetzte Hardware betrachtet und ausgewertet. Die zukünftige Anwendbarkeit der Soft- bzw. Hardware werden auf ihre Zuverlässigkeit und die weitere Verfügbarkeit von benötigtem Support untersucht.

5 Konzept

In dem folgenden Abschnitt werden basierend auf der Auswertung der Analyse verschiedene Konzepte zu der Lösung des erläuterten Problems untersucht. An erster Stelle werden softwaretechnische Lösungen besprochen und im Gegensatz dazu wird im zweiten Teil ein Ansatz zur Lösung der Aufgabe durch Austausch der Hardware untersucht.

5.1 SSH in MDE-Geräte implementieren

Sowohl der Server als auch das MDE-Gerät müssen einen SSH-Client installiert haben, um das SSH-Protokoll nutzen zu können. Im Folgenden wird auf den SSH-Server von Microsoft, sowie dessen Installation und Konfiguration eingegangen. In Zusammenhang damit wird erklärt ob, und unter welchen Bedingungen ein SSH Client auf dem MDE-Gerät installiert werden kann.

5.1.1 OpenSSH auf Windows Server 2019 installieren

Die Anwendung OpenSSH befindet sich bereits bei den Windows Servern 2019 unter der Kategorie Features und kann an dieser Stelle aktiviert werden. Installiert werden hierbei der OpenSSH Authentication Agent und der OpenSSH SSH-Server. Die Standardterminalanwendung, die bei erfolgreicher Verbindung aufgerufen wird, muss auf die Kkis-Logistikanwendung angepasst werden. Dies kann mit folgendem Befehl erfolgen: *New-ItemProperty -Path "HKLM:\SOFTWARE\OpenSSH" -Name DefaultShell -Value "../KkisMde.exe"*. Mit der Anweisung *AllowGroups* kann der Gruppe *TelnetClients* die Verbindung erlaubt werden. Sobald der private key und der public key erstellt sind, kann der Server-Daemon Service *sshd* gestartet werden. [vgl. Stahnke, 2006]

5.1.2 SSH auf MDE-Geräten ermöglichen

Das Umstellen des Standardverbindungsprotokolls ist wie in 2.3.1 bereits angesprochen nicht möglich. Es besteht keine Möglichkeit in dem BHT-Term eine weitere Funktion

wie ein SSH-Protokoll hinzuzufügen. Andere Anbieter für solche Programme, beziehungsweise andere fertige Anwendungen für diesen Zweck, sind ebenfalls nicht vorhanden. Daraus folgt, dass ein neues Dienstprogramm entwickelt werden müsste, um die Nutzung eines SSH-Protokolls auf diesen MDE-Geräten zu ermöglichen. Das Betriebssystem BHT-OS ist veraltet und es werden keine weiteren Programme mehr dafür entwickelt. Daher befindet sich kein anderes Programm mit diesen Funktionen im Umlauf. Die Möglichkeit ein solches Programm zu entwickeln muss abgewogen werden. Der Umfang einer eigenentwickelten Lösung zur Anwendung von einem SSH-Protokoll überschreitet den Rahmen dieser Arbeit, weshalb jene Protokolle aus der praktischen Untersuchung ausgeschlossen werden.

5.2 Telnet Server von Drittanbietern

Es wird als Testumgebung vorerst ein Windows Server 2019 auf einer virtuellen Maschine genutzt. Die zu testende Software wird jeweils darauf installiert und konfiguriert. Sobald dies erfolgte, wird überprüft ob die Kommunikation störungsfrei verläuft. Dafür wird der PuTTY-Client genutzt, welcher auf die Serveradresse und das Telnet Protokoll konfiguriert wird. Anschließend wird die Konfiguration des MDE-Geräts und dessen Versuchsaufbau dargestellt.

5.2.1 Einrichtung der Server-Anwendungen

Die Server-Anwendung muss im gleichen Verzeichnis wie die Kkis-Logistikanwendung installiert werden. Die Standardterminalanwendung, welche bei einer erfolgreichen Verbindung aufgerufen wird, muss auf die „KkisMde.exe“ festgelegt werden. Es soll nur für eine bestimmte Gruppe möglich sein, sich mit dem Telnet-Server zu verbinden. Es entfällt bei dem Login die Abfrage der Domain und wird daher deaktiviert. Weitere wichtige Einstellungen sind das Aktivieren des automatischen Schließens der Kkis-Logistikanwendung nach Trennen der Verbindung und das Deaktivieren des Timeouts bei einer aktiven Verbindung.

5.2.2 Nutzung des Telnet-Clients auf MDE-Geräten

Damit das MDE-Gerät konfiguriert werden kann, überträgt das BHT-Term die Software mit den angepassten Parametern auf die Hardware. Es muss die IP Adresse und der Port des Telnet-Servers eingetragen werden. Um das Loginfeld im Terminal auszufüllen, ist ein Nummernfeld ungeeignet, da der verwendete SAC, die standardisierte Zifferntastenbelegung mit mehrfach belegten Tasten (siehe Abb. 3), nicht unterstützt wird. Für den Mitarbeiter muss die Möglichkeit bestehen seinen Nutzernamen und sein Passwordeinzugeben. Dies wird durch das Einlesen eines Barcodes mit dessen Nutzernamen oder Passwort garantiert. Hierfür muss zum Test ein Barcode für den Testbenutzer generiert werden.

5.3 Austausch der MDE-Hardware

Die Option die Hardware auszutauschen wird nur in Betracht gezogen, sollten alle anderen Möglichkeiten erhebliche Nachteile aufweisen. Es kann im Rahmen dieser Arbeit kein neues Gerät im System getestet werden. Die Analyse gilt ausschließlich der Anforderungsuntersuchung und der Beschreibung, wie ein Geräteaustausch ermöglicht werden könnte.

5.3.1 Anforderungen an MDE-Geräte

Das MDE-Gerät muss so beschaffen sein, dass es verbindungsicher mit einem Server über ein TCP beziehungsweise SSH Protokoll kommunizieren kann. Es sollte weitere Besonderheiten, wie eine ausreichende Akkulaufzeit, die mindestens eine Arbeitsschicht überdauern sollte, aufweisen. Es sollte ein robustes Betriebssystem gewählt werden, für welches regelmäßig Sicherheitsupdates bereitgestellt werden. Um die schnelle Einarbeitung in die Geräte zu gewährleisten, müssen neue Geräteden alten ähneln, mindestens in Bezug auf das Scannen und die Handhabung. Das zwei- und dreidimensionale Scannen muss verfügbar sein. Des Weiteren muss das Gerät möglichst sturzresistent sein, da diese Geräte in Höhen bis zu 3 Metern genutzt werden.

5.3.2 Wechsel der MDE-Geräte

Um den Wechsel der Geräte zu ermöglichen, reicht es nicht die Hardware auszutauschen. Im Falle eines Austauschs müssen folgende Maßnahmen ergriffen werden:

- Umstellung der Server auf Open-SSH
- Einweisung der Mitarbeiter in die neue Hardware
- Untersuchung der Kompatibilität des Gerätes mit der Kkis-Logistikanwendung
- Überprüfung der korrekten Datenübermittlung in die Datenbank

Falls der Austausch der Hardware feststeht, sollte der Verkauf der alten Geräte in Betracht gezogen werden, soweit diese nicht in anderen Bereichen des Beispielunternehmens eingesetzt werden können.

6 Methoden

Der folgende Abschnitt thematisiert die Umsetzung der Konzepte unter Verwendung des Betriebssystems „Windows Server 2019“ und dem MDE-Gerät BHT 805. Außerdem wird ein ausgewähltes Konzept als Prototyp entworfen und dessen Umsetzung dargestellt.

6.1 Einrichtung der Testumgebung

Die grundlegenden Tests von Telnet- und SSH-Servern werden vorerst mithilfe einer Virtuellen Maschine (VM) durchgeführt. Die Anwendung HyperV von Microsoft ermöglicht dies. Dabei werden verschiedene VMs für jede zu testender Software erstellt. Das Betriebssystem Windows Server 2019 wird auf jenen installiert und ein virtueller Switch zur Netzwerkvirtualisierung für diese erstellt. Da die VMs keinen Zugang auf die Netzwerk-Benutzerdatenbank haben, wird je Server ein Testbenutzer erstellt und der Gruppe *TelnetClients* zugewiesen. Wie unter 2.3.5 beschrieben werden für jeden Server Profile in PuTTY angelegt, wobei jedes dieser Profile die Serveradresse beinhaltet.

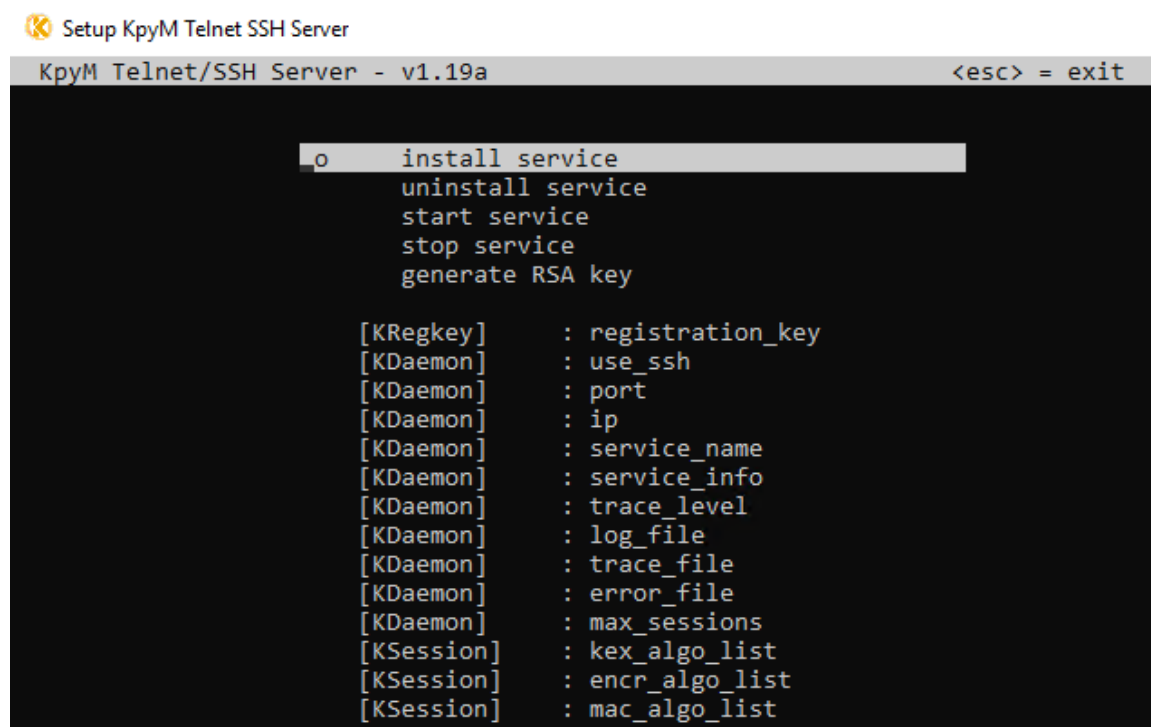
6.2 Test für die Nutzung der MDE-Geräte mit Telnet

Die Vorgehensweise der Telnet Server Tests der verschiedenen Anbieter für eine Verbindung zwischen MDE-Geräten und Server werden im Folgenden erläutert. Es wird ebenfalls auf die spezifischen Programme der jeweiligen Anbieter eingegangen.

6.2.1 Kypm Telnet Server

„Kypm Telnet Server (KTS) ist eine kostenlose Open Source Anwendung für Windows. Es unterstützt die Verbindung zwischen Computern über das Telnet oder SSH Protokoll.“ [URL-4]. Es fungiert als Terminalserver auf einem Microsoft-Server-Betriebssystem. Die Anwendung ist für Windows Server bis Windows Server 2012 zertifiziert. Da dieses Programm veraltet ist, steht der Registrationsschlüssel jedem frei zur Verfügung und kann daher kostenlos genutzt werden [vgl. URL-4].

Sobald der KTS installiert ist, können in der Anwendung diverse Einstellungen vorgenommen werden (siehe Abb. 4). Die für den Versuch notwendigen Konfigurationen sind: die Eingabe des Registrierungsschlüssels für die Freischaltung der Software, das Festsetzen des Ports auf den Standard von Telnet, die Festlegung des Log Ordners und die Einrichtung der Kkis-Logistikanwendung als Standardterminalanwendung. Damit der Server gestartet werden kann, muss der Service anschließend in Windows installiert werden. Dies geschieht unter Auswahl der in Abb.5 markierten Spalte, danach kann der Kypm Server gestartet werden. Der Port 23 ist somit offen und der Server kann mithilfe des Telnet-Protokolls über TCP angesteuert werden.



```
o  install service
   uninstall service
   start service
   stop service
   generate RSA key

[KRegkey]      : registration_key
[KDaemon]     : use_ssh
[KDaemon]     : port
[KDaemon]     : ip
[KDaemon]     : service_name
[KDaemon]     : service_info
[KDaemon]     : trace_level
[KDaemon]     : log_file
[KDaemon]     : trace_file
[KDaemon]     : error_file
[KDaemon]     : max_sessions
[KSession]    : kex_algo_list
[KSession]    : encr_algo_list
[KSession]    : mac_algo_list
```

Abbildung 4 Kypm Telnet Server

Der Verbindungstest wird durch das Aufrufen von PuTTY und der Auswahl des hierfür benötigten Verbindungsprofils initialisiert. Danach wird der Verbindungsversuch gestartet und damit auf die Login-Abfrage gewartet, welche die Verbindung zum Server bestätigt. Sobald der Login erfolgt ist, sollte sich das Kkis-Logistikprogramm in dem PuTTY-Terminal öffnen. Schlussendlich wird die Verbindung durch Schließen des Fensters getrennt und auf dem Server überprüft, ob sich die dazugehörigen Prozesse fachgerecht beendet haben.

Der Praxistest hat ergeben, dass diese Vorgehensweise grundsätzlich in dem Maß anwendbar ist, wie sie geplant war. Theoretisch konnten die Daten wie gewünscht übermittelt werden. Es wurden während der Anwendung teilweise unerwartete Probleme festgestellt, beispielsweise wurde die Kkis-Logistikanwendung nicht automatisch beendet, nachdem die Verbindung zwischen MDE-Gerät zum KTS durch Schließen des PuTTY-Terminals getrennt wurde.

6.2.2 Pragmasys Telnet Server

Für die Durchführung der Tests von Pragmasys-Telnet-Servern wird die Demoversion der Anwendung genutzt, welche für einen Monat aktiv ist. Die Anwendung ist ebenfalls nur für Windows Server bis Windows Server 2012 zertifiziert, Pragmasys erklärt jedoch ausdrücklich in der mitgelieferten Informationsdatei, dass Windows Server bis Windows Server 2019 unterstützt werden. [vgl. URL-5]

Nachdem die Installation erfolgt, werden folgende Einstellungen in dem Programm vorgenommen: das Einrichten der Debug-Logs und deren Speicherort unter dem Reiter Logging, das Festlegen der Standardterminalanwendung auf die Kkis-Logistikanwendung, sowie das Auswählen und Aktivieren des Telnet Servers unter dem Reiter InetD, beziehungsweise Telnet-Server (siehe Abbildung 5).

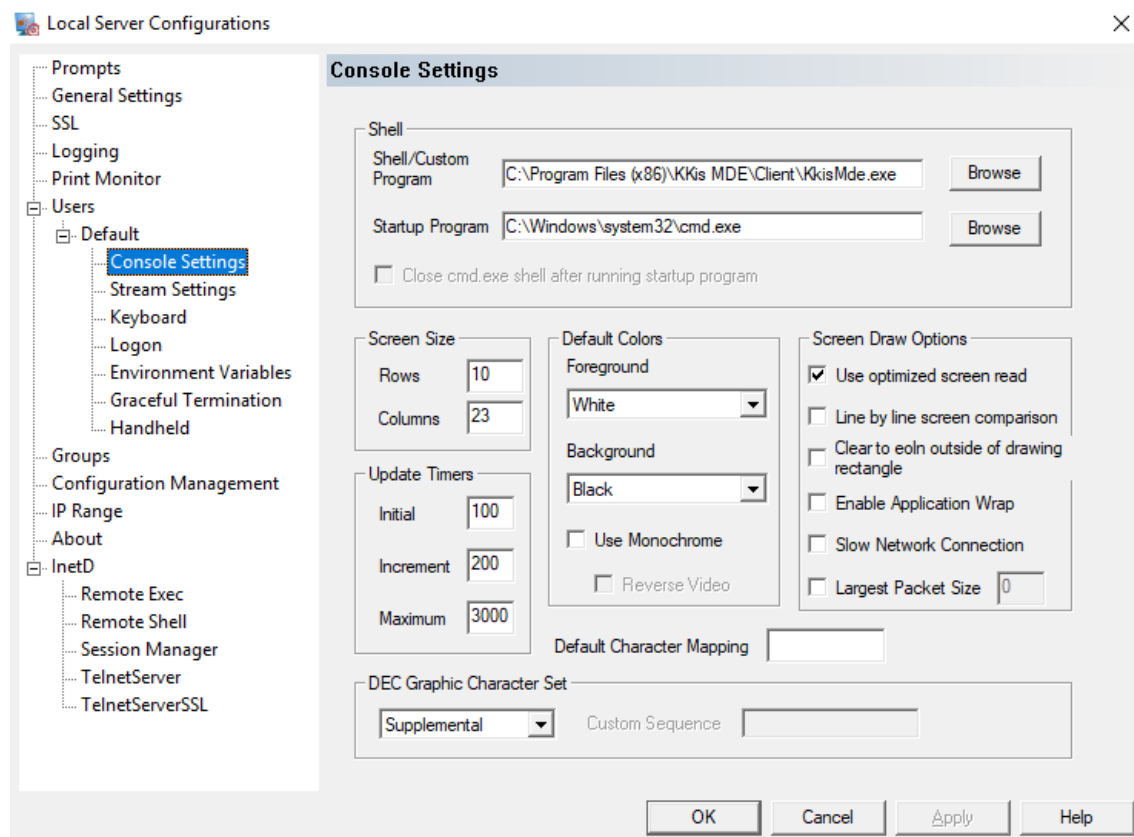


Abbildung 5 Pragmasys Server Configuration

Der Testverlauf gleicht dem von KTS in 4.2.1.

Der Verbindungsversuch zu dem virtuellen Server verläuft erfolgreich. Die Daten können ohne Probleme übertragen werden und nach Schließen des Terminals hat sich der Server sachgerecht verhalten, indem die offene Sitzung der Kkis-Logistikanwendung geschlossen wurde.

6.2.3 Weitere Telnet-Anbieter

Für die Server-Anwendungen der Anbieter Seattlelab und The Georgia SoftWorks wurden ebenfalls die Schritte zum Testen der Software erprobt. Diese scheiterten bereits bei der Installation auf Windows Server 2019. Eine Inkompatibilität war abzusehen, da die Anbieter keinerlei Informationen angeben, ob die Versionen mit Windows Server 2019 funktionieren. Die Kommunikation zwischen dem Windows Server und der Server-Anwendung war nicht möglich.

Eine andere Lösung des Problems könnte sein, den Telnet-Server über Linux auf einem Docker-Container laufen zu lassen. Dies birgt aber viele Schwierigkeiten, wie zum Beispiel die Funktionalität der Kkis-Logistikanwendung und das Synchronisieren der Benutzer mit dem Linux-Container.

6.3 Aufsetzen des Prototyps mit Pragmasys

Die erfolgreich getestete Serveranwendung wird auf einem Server installiert, welcher sich im gleichen Netzwerk wie das MDE-Gerät befindet. Darauf wird der Prototyp für ein funktionsfähiges System implementiert und getestet.

Das Programm Pragmasys-Telnet wird, wie in dem Versuch in 4.3.1 beschrieben, aufgesetzt und konfiguriert. Außerdem muss die Gruppe *TelnetClients* auf dem Windows Server und in der Pragmasys-Anwendung erstellt werden. Danach wird für diese Benutzergruppe die Standardterminalanwendung für den Terminalserver auf die Kkis-Logistikanwendung festgelegt.

Das in 2.4 erwähnte Skript, welches das tägliche Aktualisieren der Benutzer, sowie das Starten und Stoppen der Telnet-Prozesse beinhaltet, wird erstellt. Das Synchronisieren der Benutzer wird durch die Anwendung Groupsync.exe realisiert (siehe Abbildung 7 Z. 27).

Der Verbindungsaufbau zwischen Server und MDE-Gerät wird geprüft, gleichzeitig versucht das Gerät sich mit dem generierten Barcode (siehe 3.3.2) einzuloggen. Anschließend wird getestet, ob die Logistikanwendung ihre Funktionen ausführen kann und somit die volle Funktionsfähigkeit des entworfenen Prototyps besteht.

```
20
21 'Telnet-Dienst stoppen'
22 Import-Module "$PSScriptRoot\..\Services.psm1"
23 [string] $ComputerName = GetKkisMdeServerName 'TEST'
24 Stop-KkisMdeService $ComputerName
25
26 'GroupSync aufrufen'
27 &$PSScriptRoot\..\GroupSync.exe LDAP://CN=Nutzer_MDE,OU=Gruppen,OI
28 | Out-Null
29
30 'Telnet-Server neu starten'
31 Restart-Computer $ComputerName -Force
32
33 Stop-Transcript
```

Abbildung 6 Quellcode von einem Skript zum Aktualisieren der Serverdaten

7 Fazit und Ausblick

Die bisher durchgeführten Versuche werden im Folgenden ausgewertet und es wird analysiert, welche Möglichkeiten der KOMSA AG zur Verfügung stehen. Auf Basis dieser Erkenntnisse werden Empfehlungen formuliert.

7.1 Auswertung der Ergebnisse

Es wurde untersucht inwieweit vier verschiedene Telnet-Server auf Windows Servern 2019 funktionieren und ob es möglich ist über diese die Kkis-Logistikprozedur nachzustellen. In zwei Fällen ist dies gut gelungen. Da Pragmasys-Telnet die besseren Eigenschaften in Bezug auf deren Funktionalität aufweist, wurde der Prototyp damit entwickelt. Das endgültige Aufsetzen des Prototyps ist gelungen und es konnte eine stabile Verbindung zum MDE-Gerät hergestellt werden, welches die Anforderungen der Anwendung erfüllt.

7.2 Einschätzung der Lage

Es konnten nicht alle Konzepte untersucht beziehungsweise umgesetzt werden, da das SSH-Protokoll nicht geprüft werden konnte, ohne neue Geräte anzuschaffen. Durch erfolgreiche softwaretechnische Lösungen des Prototyps, ist ein Gerätevorschlag für die KOMSA AG nicht erforderlich.

Die Lösung den Telnet-Server von Pragmasys zu benutzen wird für die zukünftigen Jahre stabil funktionieren. Der optionale Support kann dabei auch in Zukunft Sicherheit bieten. Die Kosten für die Anwendung mit den erforderlichen Eigenschaften beläuft sich auf 699 USD und die Kosten für den Support betragen 119,8 USD pro Jahr. Darin sind die maximale Prozessoranzahl von 8 Prozessoren und Verbindungsanzahl von 20 aktiven Verbindungen abgesichert.

Im Gegensatz dazu würde die Anschaffung neuer Geräte in Zukunft sehr aufwändig, aber auch um einiges sicherer sein. Pragmasys zu benutzen, wird wesentlich weniger Aufwand

bedeuten. Dies bedeutet jedoch ein erhöhtes, aber nicht essenzielles, Sicherheitsrisiko. Außerdem ist die Weiterbenutzung der alten MDE-Geräte um einiges einfacher für die Mitarbeiter, da sie sich nicht in neue Geräte einarbeiten müssen. Die Vorteile, die MDE-Geräte abzulösen, wären die Erhöhung der Sicherheit durch Nutzung von dem SSH-Protokoll und der einfacheren Konfiguration. MDE-Geräte, welche die Anforderungen erfüllen wären alle Denso Geräte ab der Version BHT 900 mit dem „Windows Embedded CE“ oder „Android“ Betriebssystem. Die Kosten für die neue Hardware würden sich auf etwa 1000 € pro Gerät belaufen.

7.3 Empfehlung an die KOMSA AG

Vorerst sollte auf dem Server, wenn er auf die neue Version umgestellt wird, ein Telnet Server eines Drittanbieters installiert werden. Aufgrund deutlich herausgehobener Funktionalität empfiehlt sich die Anwendung von Pragmasys. Die Tatsache, dass Telnet weiterhin eine Sicherheitslücke darstellt, ist jedoch nicht außer Acht zu lassen. In Zukunft sollte über die Anschaffung neuer Geräte nachgedacht werden, welche die Anforderungen des Unternehmens erfüllen, wie in 3.3.1 beschrieben. Eine Umstellung der Geräte würde bevorzugt eine Erneuerung der Kommunikationsprotokolle auf SSH-Protokolle mit sich führen. Durch die folgende Ablösung des Telnet-Protokolls kann letztendlich die Sicherheit der Übertragung über die Kommunikationsschnittstelle um vielfaches angehoben werden.

Literaturverzeichnis

Comer, D. E. (2003) *TCP/IP, Protokolle und Architekturen*. 4. Auflage. Bonn: mitp-Verlag.

Heiserich, O.-E.; Helbig, K.; Ullmann, W. (2011) *Logistik Eine praxisorientierte Einführung*. 4. Auflage. Wiesbaden: Gabler Verlag | Springer Fachmedien Wiesbaden GmbH

Riedel, S (2004) *SSH kurz & gut*. 1. Auflage. Köln: O`Reilly Verlag.

Sauter, M. (2018) *Grundkurs Mobile Kommunikationssysteme*. 7. Auflage. Wiesbaden: Springer Vieweg

Stahnke, M. (2006) *Pro OpenSSH*. USA: Springer-Verlag New York, Inc.

URL-1 (05/10/2019) KOMSA AG: *Daten & Fakten zur KOMSA-Gruppe*. URL: <https://komsa.com/unternehmen>

URL-2 (05/10/2019) Gregory, K: *Telnet Server Removed From Windows Server 2016*. URL: <http://innovativeii.com/telnet-server-removed-windows-server-2016>

URL-3 (03/10/2019) Microsoft Corporation: *Installation von OpenSSH für Windows Server 2019 und Windows 10*. URL: https://docs.microsoft.com/de-de/windows-server/administration/openssh/openssh_install_firstuse

URL-4 (03/10/2019) Grigorov, K.: *KpyM Telnet/ SSH Server*. URL: <http://www.kpym.com/2/kpym/index.htm>

URL-5 (03/10/2019) Pragma Systems Inc: *Pragma Telnet Server*. URL: <https://www.pragmasys.com/telnet-server>

URL-6 (11/09/2019) Denso Wave: *BHT Term Emulator*. URL: <https://www.denso-wave.com/en/adcd/product/software/bht-term-emulator.html>

URL-7 (10/09/2019) Denso Wave: *Hand-held terminal BHT-800*. URL: https://www.denso-wave.com/en/adcd/product/handy_terminal/bht-800.html

Selbstständigkeitserklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Stellen, die wörtlich oder sinngemäß aus Quellen entnommen wurden, sind als solche kenntlich gemacht.

Diese Arbeit wurde in gleicher oder ähnlicher Form noch keiner anderen Prüfungsbehörde vorgelegt.

Mittweida, den

Richard Schubert